**Traffic Sign Classification Project**

**Data Augmentation:**

* Augment the existing data by adding some additional samples

**Preprocessing:**

* Convert to Gray Scale
* Flip left/right
* Rotate
* Translate
* CLAHE equalization

**Model Architecture:**

I used LeNet architecture with 2 layers of convolution, Leaky RELU activations, Max Pooling and Dropout followed by 3 ‘Fully Connected’ layers.

*Layer 1:*

* Convolution
  + Filter Size: 5x5x32
  + Strides: 2x2
  + Padding: VALID
* Leaky RELU
  + Alpha = 0.5
* Max Pooling
  + ksize: [1, 2, 2, 1]
  + strides: [1, 2, 2, 1]
* Dropout
  + Probability = 0.5

*Layer 2:*

* Convolution
  + Filter Size: 5x5x64
  + Strides: 2x2
  + Padding: VALID
* Leaky RELU
  + Alpha = 0.5
* Max Pooling
  + ksize: [1, 2, 2, 1]
  + strides: [1, 2, 2, 1]
* Dropout
  + Probability = 0.5

*Fully Connected Layer 1:*

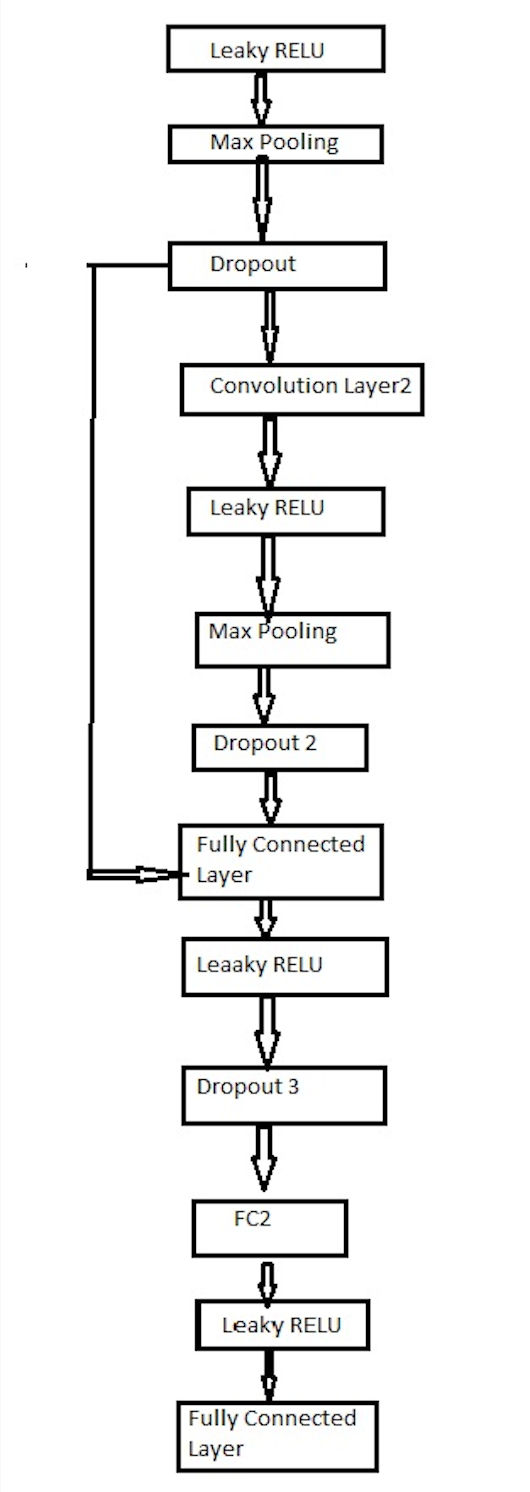
* Added flattened outputs from layers 1 & 2
* Matrix Multiplication
* Leaky RELU
  + Alpha = 0.5
* Dropout
  + Probability = 0.5

*Fully Connected Layer 2:*

* Matrix Multiplication
* Leaky RELU
  + Alpha = 0.5

*Fully Connected Layer 3:*

* Matrix Multiplication



**Model Training:**

learning\_rate = 0.0005

epochs = 100

batch\_size = 128

L2 lambda = 0.0001

* Used softmax as well as L2 regularization to determine the loss
* Used AdamOptimizer for minimizing the loss

**References:**

1. <http://www.pyimagesearch.com/2017/01/02/rotate-images-correctly-with-opencv-and-python/>
2. <https://opencv-python-tutroals.readthedocs.io/en/latest/py_tutorials/py_imgproc/py_histograms/py_histogram_equalization/py_histogram_equalization.html>
3. <http://stackoverflow.com/questions/24341114/simple-illumination-correction-in-images-opencv-c>