****Build a Traffic Sign Recognition Project****

The goals / steps of this project are the following:

* Load the data set (see below for links to the project data set)
* Explore, summarize and visualize the data set
* Design, train and test a model architecture
* Use the model to make predictions on new images
* Analyze the softmax probabilities of the new images
* Summarize the results with a written report

1.Summarize the dataset

* The size of training set is 37499
* The size of the validation set is 4410
* The size of test set is 12630
* The shape of a traffic sign image is 32x32
* The number of unique classes/labels in the data set is 43

2. Include an exploratory visualization of the dataset.

Here is an exploratory visualization of the data set.

Then I convert the images to grayscale and normalize them, so that the image data has mean zero and equal zero.For image data,(pixel -128)/128 is a quick way to do that

Here is an exploratory visualization of the data set after grayscale and normalization

![alt text][image2]

As a last step, I normalized the image data because ...

I decided to generate additional data because ...

To add more data to the the data set, I used the following techniques because ...

Here is an example of an original image and an augmented image:

![alt text][image3]

The difference between the original data set and the augmented data set is the following ...

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

Layer	Description
Input	32x32x1 image
Convolution 5x5	1x1 stride, valid padding, outputs 28x28x32
RELU	
Max pooling	2x2 stride, outputs 14x14x32
Convolution 5x5	1x1 stride, valid padding, outputs 10x10x64
RELU	
Max pooling	2x2 stride, outputs 5x5x64
Convolution2X2	1X1 stride, same padding outputs 5x5x128
RELU	
Max pooling	2x2 stride, outputs 4x4x128
Flatten	
Fully Connected	input 2048 output 512
Fully Connected	input 512 output 256
Fully Connected	input 256 output 43

To train the model, I used an AdamOptimizer, 256 of the batch size, 50 epochs and 0.001 of the learning rate

My final model results were:
* training set accuracy of 0.990
* validation set accuracy of ?0.938
* test set accuracy of 0.929

First I just use the original LeNet,and it gave an accuracy of 55,I then normalized my data and converted them to grayscale and saw that the model gives an accuracy of 72 for 10 epochs. I then increase the number of epochs to 50 and I achieved the accuracy of 85,after that the accuracy didn’t change much even I increase the epochs; Then I increase one convolutional layer in lenet and add dropout, finally I got the accuracy to 0. 990for training set.

Test a Model on New Images

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

About the five pictures, the third one might be hard to classify because the word might be misclassified to something else like 60;

The model was able to correctly guess 4 of the 5 traffic signs, which gives an accuracy of 80%.

Here are the results of the prediction:

Image	Prediction
Priority road	Priority road
Road work	Road work
Stop	Speed limit (60km/h)
Children crossing	Children crossing
Bicycle crossing	Bicycle crossing

The accuracy of the model when tested on the new images is a little small than the accuracy in the original test set, I think it's because the number of new images is too small.

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.

The code for making predictions on my final model is located in the 11th cell of the Ipython notebook.

The top five softmax probilities of the predictions on the captured image are outputted
Below are my softmax results
The first one and the fourth one is the model certain of, the second one and the fifth one is a little uncertain , but the third one is clearly uncertain,the correct prediction didn't appear in the top five softmax probabilities.

Original traffic sign was: 12 (Priority road)
Predicted with a probability of 100.000% to be 12 (Priority road)
Predicted with a probability of 0.000% to be 40 (Roundabout mandatory)
Predicted with a probability of 0.000% to be 7 (Speed limit (100km/h))
Predicted with a probability of 0.000% to be 9 (No passing)
Predicted with a probability of 0.000% to be 2 (Speed limit (50km/h))

Original traffic sign was: 25 (Road work)
Predicted with a probability of 99.979% to be 25 (Road work)
Predicted with a probability of 0.011% to be 24 (Road narrows on the right)
Predicted with a probability of 0.003% to be 30 (Beware of ice/snow)
Predicted with a probability of 0.003% to be 28 (Children crossing)
Predicted with a probability of 0.003% to be 22 (Bumpy road)

Original traffic sign was: 14 (Stop)
Predicted with a probability of 41.972% to be 3 (Speed limit (60km/h))
Predicted with a probability of 8.706% to be 12 (Priority road)
Predicted with a probability of 8.583% to be 40 (Roundabout mandatory)
Predicted with a probability of 6.340% to be 11 (Right-of-way at the next intersection)
Predicted with a probability of 5.464% to be 17 (No entry)

Original traffic sign was: 28 (Children crossing)
Predicted with a probability of 100.000% to be 28 (Children crossing)
Predicted with a probability of 0.000% to be 30 (Beware of ice/snow)
Predicted with a probability of 0.000% to be 29 (Bicycles crossing)
Predicted with a probability of 0.000% to be 24 (Road narrows on the right)
Predicted with a probability of 0.000% to be 23 (Slippery road)

Original traffic sign was: 29 (Bicycles crossing)
Predicted with a probability of 93.379% to be 29 (Bicycles crossing)
Predicted with a probability of 5.817% to be 28 (Children crossing)
Predicted with a probability of 0.531% to be 24 (Road narrows on the right)
Predicted with a probability of 0.131% to be 23 (Slippery road)
Predicted with a probability of 0.124% to be 30 (Beware of ice/snow)