

Machine Learning Engineer Nanodegree

Capstone Proposal

Jiaxi Yuan

August 7th, 2018

Domain Background

Due to the high speed development of machine learning, this technique has been used across our daily life. For example, autonomous driving, which is an unimaginable concept maybe 15 years ago, is applied to real life and mature products have been manufactured to the market. However, machine learning is not only something far away or expensive to get in touch, we can find it everywhere now, such as the intelligent recommendation of the music app and also what will be mainly discussed here -- traffic sign recognition.

Traffic sign is what we can see everyday, especially for those who need to drive. Traffic sign includes different types, such as stop sign, speed limit sign. Due to my personal interest in autonomous driving, I did a lot of research on what kind of techniques needed. The traffic sign recognition is one of the most basic and necessary part for the self-driving car. It needs the support of machine learning with enough data set.

Problem Statement

The problem to be investigated here is how to use machine learning to realize traffic sign recognition. German traffic sign dataset will be used as both training data and testing data. Some image processing needs to be done so that all the images can be better used as the input. Image augmentation might need to be done for the better recognition performance. For the model architecture, deep learning will be first considered.

Datasets and Inputs

What I will use is the preprocessed data set from another Udacity project called German Traffic Sign Recognition. The dataset is originally called GTSRB, known as German Traffic Sign Recognition Benchmark. GTSRB is a multi-class, single-image classification challenge held at the International Joint Conference on Neural Networks (IJCNN) 2011.

The images contain one traffic sign each. They contain a border of 10 % around the actual traffic sign (at least 5 pixels) to allow for edge-based approaches. The image size varies between 150 to 250 pixels. The training data set contains 39209 training images in 43 classes. The test data set contains 12630 test images. Each image has an annotation which includes: 1. Filename: Filename of corresponding image 2. Width: Width of the image 3. Height: Height of the image 4. ROI.x1: X-coordinate of top-left corner of traffic sign bounding box 5. ROI.y1: Y-coordinate of top-left corner of traffic sign bounding box 6. ROI.x2: X-coordinate of bottom-right corner of traffic sign bounding box 7. ROI.y2: Y-coordinate of bottom-right corner of traffic sign bounding box 8. ClassID: Assigned

class label(For training data only)

Solution Statement

First, the data set will be loaded and some image processing needs to be done for the images used as the input. Image augmentation is needed as some of the images are pretty vague which might result in bad result. As each input data has a label, supervised learning will be applied. For the training architecture, deep learning with different layer combinations will be tried to have the best performance. Finally, after the model trained, it'll be applied to the test data set and generate the accuracy to evaluate the performance of the model.

Benchmark Model

GTSRB can be used as the benchmark. It provides the top best results with different methods. The website provides the training and testing data set and also will update the data set accordingly. All the data set provided are labeled with the ground truth. They also provided the competition task. I think the website provides enough information along with the good results from different teams which is suitable to be used as the benchmark.

Evaluation Metrics

The metrics will be evaluated by the accuracy. The highest accuracy rate provided by GTSRB is over 99%. I'll use this as the benchmark and accuracy at least 95% is the expectation.

Project Design

1. Data preprocessing: For each image, it will be scaled and transformed to greyscale image. Then some image augmentation will be done to enhance the quality. Also, some image might need flipping or rotation.
2. Training: Plan to apply deep learning with different layer combinations, such as convolutional layer with some regularization.
3. Testing and optimizing: Apply the trained model to the testing data and get the final accuracy of the model. Find the best way to do traffic sign recognition. After choosing the best model, try different parameters to optimize the model.