UDACITY

‹ Return to Classroom

Traffic Light Classifier

REVIEW

CODE REVIEW

HISTORY

Meets Specifications

Awesome job with your submission! Your project is very impressive, you have a brilliant future in the self driving car industry 🏆 Hand engineer features is difficult, in the future you will see more advanced computer techniques like convnets and semantic segmentation that are going to make your life easier.

Notebook Questions

In the project notebook, all questions are answered. (There are two questions total.)

✔ Nice description of the features you used and the thought process behind choosing them!

Pre-processing

All input images (before they are classified) should be processed so that they are the same size.

✔ Great work! You've standardized all the image data into the same size with the shape of 32x32!

All labels should be a one-hot encoded vector of length 3. Ex. 'yellow' becomes: [0, 1, 0].

✓ Very good ! Your function performs one-hot encoding perfectly.

Create a brightness feature

Using HSV colorspace, extract a feature from a traffic light image that represents the level(s) of brightness in an image. This feature can help classify any traffic light image. A feature can be a list, array, or a single value.

✓ Great work! You extracted the colour feature using the HSV color space from the traffic light images.

Model Evaluation

The model must have greater than 90% accuracy on the given test set.

In the given test set, red traffic lights can never be mistakenly labeled as green.

Classification Model

Using any created features, write a classification function that takes in a standardized RGB image and outputs whether a traffic light is red, yellow, or green as a one-hot encoded label.

✓ Dear student you went above and beyond! 97.97% accuracy with only 6 misclassified images is outstanding 😄

DOWNLOAD PROJECT

RETURN TO PATH

Rate this review

START