项目

Use Deep Learning to Clone Driving Behavior

此部分属于 Self Driving Car Engineer Nanodegree Program

项目审阅 | 代码审阅 2 | 注释

Meets Specifications

SHARE YOUR ACCOMPLISHMENT

Excellent work, you nailed it!
👏👏👏

I can see you put a lot of effort in your project and advanced a lot, you should be really proud!

You have shown a firm grasp of the concepts presented here and are good to go.

I must say, this is one of the best submissions for this project I've seen so far 👍

Keep going and good luck!
Paul

PS. If you have further questions remember you can find me on Slack as `@viadanna`

Required Files

The submission includes a model.py file, drive.py, model.h5 a writeup report and video.mp4.

All required files are present.

Quality of Code

The model provided can be used to successfully operate the simulation.

The provided models can be used to successfully drive the autonomous car.

The code in `model.py` uses a Python generator, if needed, to generate data for training rather than storing the training data in memory. The `model.py` code is clearly organized and comments are included where needed.

Nice generator and model implementation.

Awesome

Excellent work commenting and organizing the code.

Model Architecture and Training Strategy

给这次审阅打分

The neural network uses convolution layers with appropria… to introduce nonlinearity into the model. The data is normalized in the model.

Excellent choice of architecture.

Train/validation/test splits have been used, and the model uses dropout layers or other methods to reduce overfitting.

Excellent work using pooling, dropout and augmentation to reduce variance.

Suggestion

You can check the this article for further information on using dropout.

Learning rate parameters are chosen with explanation, or an Adam optimizer is used.

Good idea using the AdamOptimizer that actually takes care of updating the learning rate.

Suggestion

If you want to know more about optimizers, check this article for a nice description and comparison of different algorithms.

Instead of using a fixed number of epochs one alternative is using Keras' EarlyStopping callback which stops training the model when it stops improving.

Training data has been chosen to induce the desired behavior in the simulation (i.e. keeping the car on the track).

Excellent work here, your model developed nice driving skills based on the training data.

Architecture and Training Documentation

The README thoroughly discusses the approach taken for deriving and designing a model architecture fit for solving the given problem.

Awesome

Excellent work experimenting with the network architecture.

The README provides sufficient details of the characteristics and qualities of the architecture, such as the type of model used, the number of layers, the size of each layer. Visualizations emphasizing particular qualities of the architecture are encouraged.

Nice description and visualization of the architecture.

Suggestion

If you desire to deep-dive into the architecure using TensorBoard, Keras includes a callback to simplify saving the logs.

The README describes how the model was trained and what the characteristics of the dataset are. Information such as how the dataset was generated and examples of images from the dataset must be included.

Excellent work augmenting the training dataset with left/right images.

Simulation

No tire may leave the drivable portion of the track surface. The car may not pop up onto ledges or roll over any surfaces that would otherwise be considered unsafe (if humans were in the vehicle).

The model correctly navigated both courses, congratulations!

下载项目

2 代码审阅评注 ›

返回 PATH

学员 FAQ Reviewer Agreement