Predicting Steering Angles in self - driving cars

Neha Yadav and Rishi Mody

Introduction

- Recent years have seen a tremendous push from leading automobile manufacturers to make cars more and more technologically equipped.
- A major development has been in making vehicles autonomous and self driven.
- A lot of artificial intelligence and deep learning technologies such as CNN, RNN, GANs etc. have been applied to make this futuristic concept a reality.
- The goal is to build a model that, given an image taken while driving, minimizes the RMSE (root mean squared error) between the predicted steering angle and the actual one produced by a human driver.

Dataset

- The dataset has been provided by Udacity as a part of its Self-Driving Car challenge. They have generated images using NVIDIAs DAVE-2 System which uses 3 cameras placed behind the windshield of the car. A time stamped video is captured along with the steering angle applied by the human driver. The video is captured in varying conditions of light and traffic.
- Training data set contains 101397 frames and corresponding labels including steering angle, torque and speed. There is also a test set which contains 5615 frames. The original resolution of the image is 640x480.

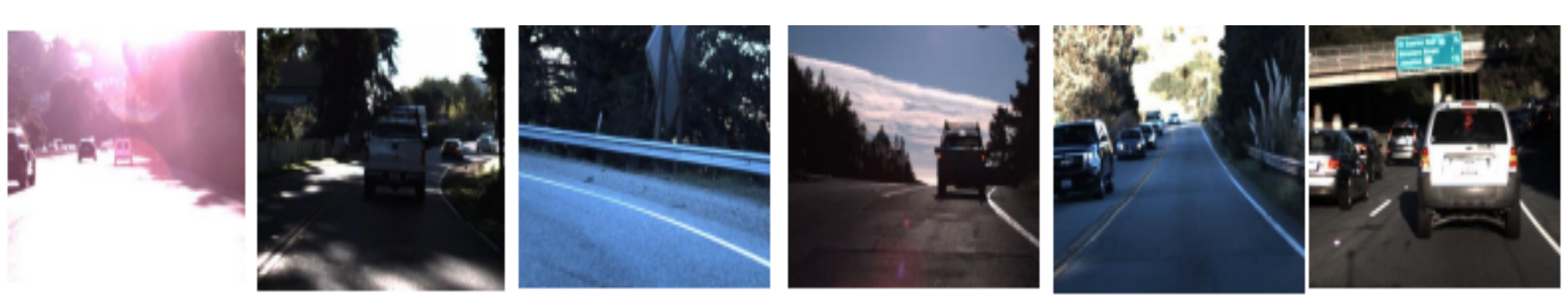


Fig. 1. Training data set. Typical images for different light, traffic and driving conditions. (a) Direct sunlight, (b) shadow, (c) sharp left turn, (d) uphill, (e) straight, (f) heavy traffic

Models

We developed two types of model structures. One is inspired from the architecture used in NVIDIA's end-to-end learning model for self-driving cars while the other is a hybrid model that uses 2D pre-trained convolutional model from transfer learning

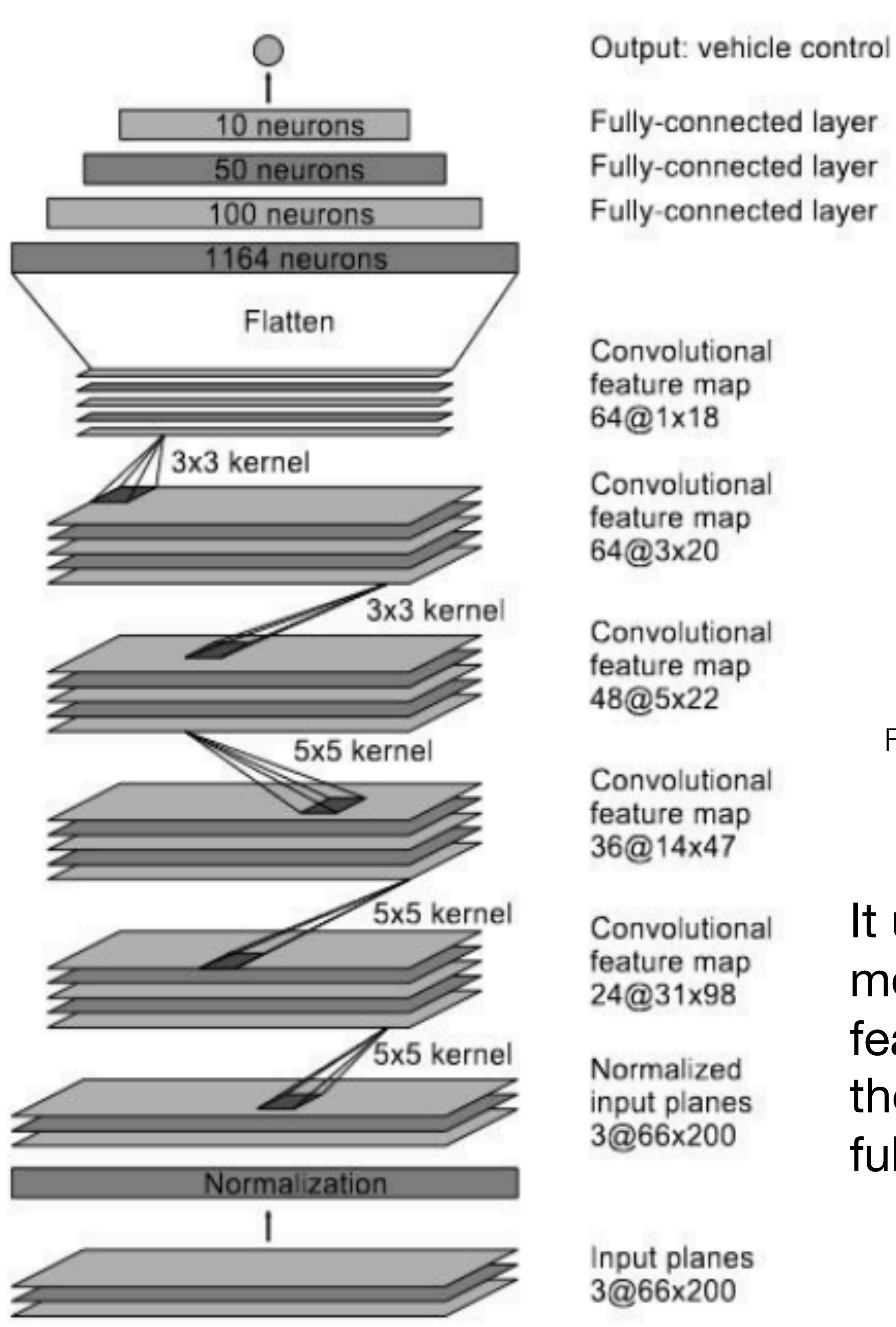


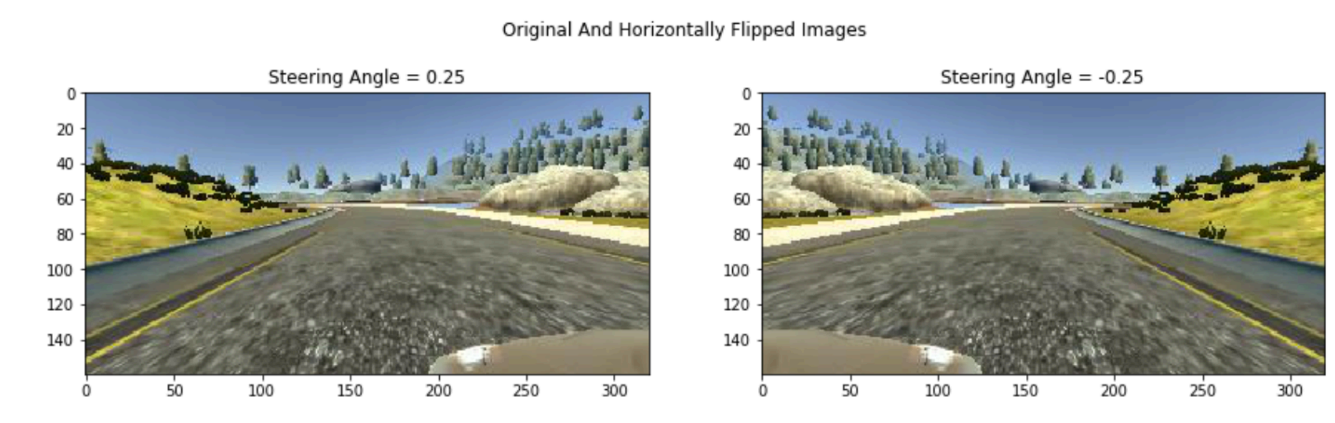
Fig. 2. NVIDIA architecture

Fig. 3. 2D pre-trained convolutional model from transfer learning

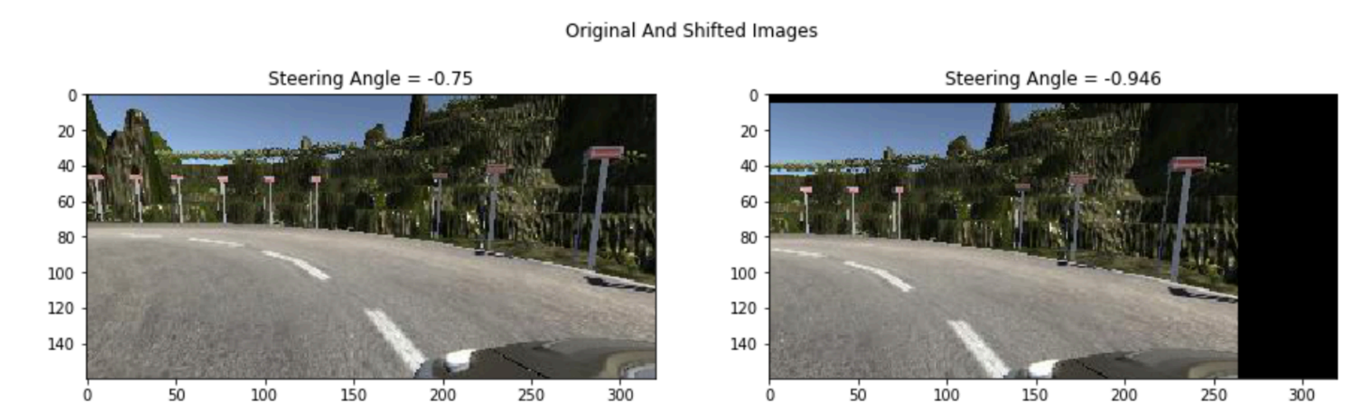
It uses a pre-trained VGG16 model to extract visual features and processes those features with stacked fully connected layers.

Experiments

Data Augmentation: We generated new and augmented images on the fly as we trained the model. This was done in order to generalize our model. For each image we randomly shifted vertically between -20 and +20%. We also randomly apply a darkened area to each image. To neutralize steering bias from the imbalanced training data we performed horizontal flip on images with the corresponding inversion of steering angle.



Original vs flipped image

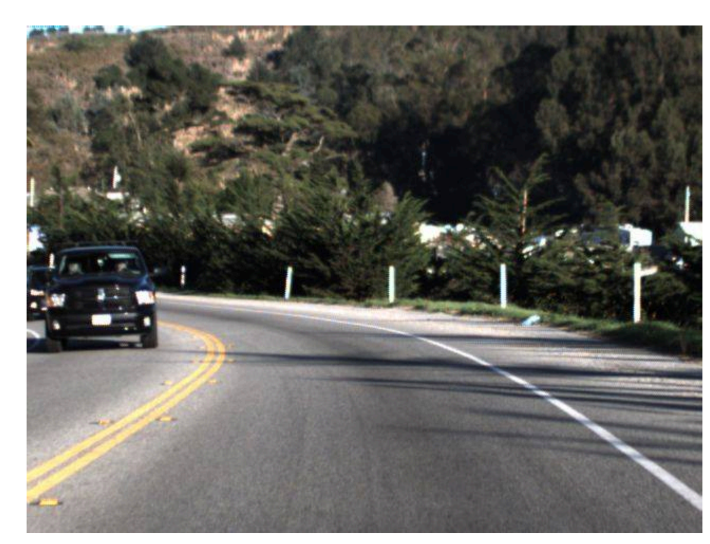


Original vs shifted image

Model: Apart from our two main models we trained another Conv net model. This model consists of 5 convolutional layers with Max pooling between them followed by a couple of dense layers and a linear activation to output continuous steering angles. We use 'elu' activation throughout with adam as an optimizer.

Results

Predicted : 0.00143
True: 0.00174



Predicted : 0.00757
True: 0.00776

Model	Transfer Learning	Conv Net	Experimental Conv Net
RMSE	0.187	0.2036	0.2631

Conclusions

- Transfer learning with a model trained on ImageNet gave the best result on this dataset. This model outperforms NVIDIA's model.
- For the amount of epochs utilized, only minimal data augmentation proved to be of any major use for these models.

Future work

- In order to reduce loss even further we can apply a 3D conv network
- Because of the nature of the input which is a series on consequence images using RNN with a memory unit can be useful.
- GANs could be used to generate more scene with sharp angles and to transform the dataset.
- Use a smoothing function after the output, training longer, better augmentation, etc.
- Speed and throttle can also be considered as features and their values can also be predicted moving the model closer to a fully functional autonomous car.