Training End-to-end Steering Models with Unfiltered Driving Data

Pesonen Julius[1]

[1] **Aalto University, Department of Mechanical Engineering**

ABSTRACT

A small study in implementing and testing improved deep learning training methods for end-to-end steering models with a focus on training with unfiltered data containing images and labels from unuseful scenarios such as crossroad situations. The model used for the testing is a simple convolutional neural network, PilotNet, and the training was tested with and without augmentations and with the usage of iterative trimmed loss minimization.
The results of the study were poor and further studies should be conducted to draw definitive conclusions. Best results were obtained by using small data augmentations without the usage of the iterative trimmed loss minimization.

Keywords: Computer Vision and Pattern Recognition, Artificial Intelligence, Machine Learning, Robotics

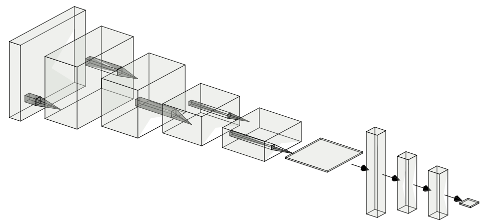
INTRODUCTION

End-to-end steering for autonomous driving has been studied a lot in the recent years with impressive results. However, most of the time these studies rely on heavily filtered and curated datasets only consisting of manually chosen images and corresponding steering angle labels. The goal of this study was to explore and experiment on the possibility of training end-to-end steering models with unfiltered data consisting of only automatically gathered steering angles and driving images, without removing any unnecessary or even harmful scenes such as intersection or lane change situations.

The possibility of training the models using such datasets would allow for a much lower bar of entry in the end-to-end steering model development as these datasets are readily available online. This would mean that researchers around the world could contribute to the research without an access to a data gathering platform and without the tedious task of manually filtering and curating the data for training.

MODEL

The original PilotNet model, by Bojarski et al. (2016), was implemented for this study, due to its fairly simple structure and high number of prior studies proving its functionality. The model is not the the best performing end-to-end steering model at the date of this study but it is assumed that improvements in the training scheme that affect the PilotNet model can be further translated to higher performance models as well.



Layer type	Input dimension	Kernel size
Normalization	$3 \times 66 \times 200$	
2d Convolution	$24 \times 31 \times 98$	5×5
2d Convolution	$36 \times 14 \times 47$	5×5
2d Convolution	$48 \times 5 \times 22$	3×3
2d Convolution	$64 \times 3 \times 20$	3×3
Flatten	$64 \times 1 \times 18$	
Fully-connected	1152	
Fully-connected	100	
Fully-connected	50	
Fully-connected	10	
Output	1	

Figure 1. The PilotNet architecture. Each layer except the output layer is paired with a ReLU activation function.

DATA

The A2D2 dataset by Audi, Geyer et al. (2020), was chosen for use in this study as it contains lots of unlabelled CAN-bus and image data from driving. The CAN-bus data includes the steering angles of each corresponding image timestamp. The dataset consists of multiple subsets and for the first part of this study the smallest of the three unlabelled subsets titled Gaimersheim was used. This subset contains 15688 RGB-images recorded at 30 frames per second, from which the first 80% were chosen for training and the latter 20% for validation. The data is ordered by time and the training-validation split was performed without shuffling the data. This results in less similarities between the training and validation data thus giving more meaningful results that represent the actual generalisation of the models.

It has to be noted that all data used in this study also contains images and labels from situations that the models are not expected to perform on, including the validation set. Thus the results are not directly comparable to other studies on the subject and require their own respective comparisons and evaluations to be used in the future.

Data Preprocessing

For usage with the PilotNet model, the images were cropped from bottom up to and rescaled match the aspect ratio of 3:1 and input size $200 \times 66 \times 3$ of the model. This resulted in smaller resolution images with most of the view above horizon removed from the image, as seen in figure 2, to enable the model to concentrate on the road. The image data was also reduced to 10 frames per second for both the training and validation set, to reduce similarities between images for more effective training as was done in the study by Bojarski et al. (2016).

The labels were converted from degrees, used in the A2D2 dataset, to radians used by the autonomous driving research vehicle of Aalto University, to allow for easier future development and real world testing.

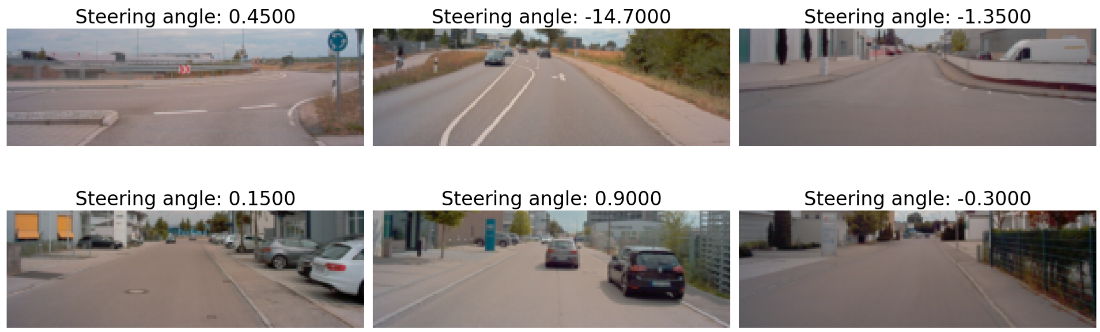


Figure 2. Random samples from the A2D2 dataset after preprocessing for PilotNet. The labels are in degrees as provided in the original dataset. Negative steering angles correspond to steering to the left and positive to the right.

Data Augmentations

For training purposes, some randomised data augmentations were applied to the training images to enable the model to improve results on the validation set. As seen in figure 3, the PilotNet model is unable to generalise when trained on the unfiltered data without the usage of any training augmentations, resulting in an overfitted model unable to perform any succesful predictions on the validation set.

The data was augmented by applying random horizontal flips, simulated brightness changes and small rotations, as they have been succesfully used to improve training by Yang et al. (2018). This adjustment already enables the model to perform predictions on the unaugmented validation set as seen in figure 3. The rotations were applied in the range of $-10°$ to $10°$ and the brightness was changed between -25% and 25%.

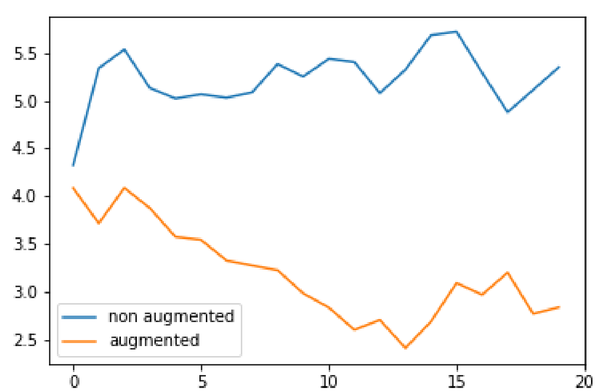


Figure 3. Validation errors during training with training done without augmentations and training with augmentations.

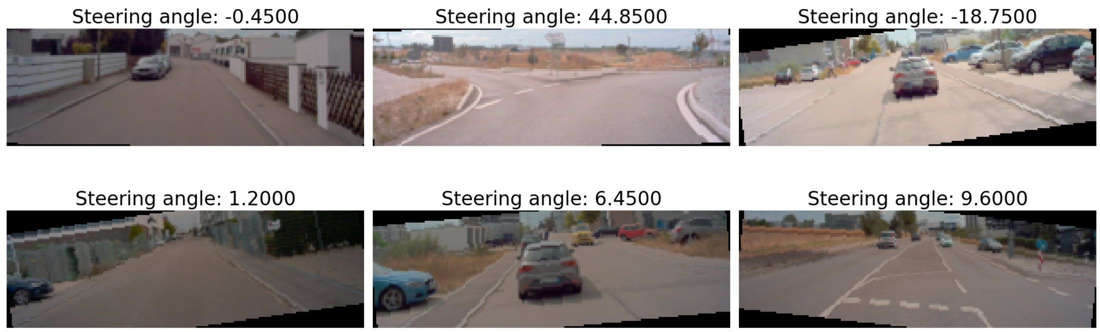


Figure 4. Samples from the dataset with the described random augmentations applied.

TRAINING

The model was first trained for 20 epochs with randomized minibatches of 32 samples using the Adam optimizer, Kingma and Ba (2014), with a learning rate of 1e-4 and betas (0.9, 0.999). The validation errors during training were tracked to analyse the learning of the model. Iterative trimmed loss minimization, which is described more in depth in its own subsection, was implemented in an attempt to improve the model's learning with the unfiltered dataset.

Iterative Trimmed Loss Minimization

As the data is known to contain samples which are in practice impossible or at least undesirable for the model to learn and only applying data augmentations will not completely remove the problem but only make it more manageable. Thus, in theory, it would be preferable to use a training scheme that estimates samples which should be left out of the training set. To test this hypothesis, a training scheme called iterative trimmed loss minimization proposed by Shen and Sanghavi (2018) was implemented and tested for the model. This scheme trims the training dataset by subsampling only the samples with the lowest loss, leaving the worst case scenarios out of the training set, which in this case are known to contain non useful information.

The iterative trimmed loss minimization algorithm is is tuned with the parameter α which determines the portion of samples used in the later stages of the training. In this study the values of 10% and 20% for α were tested which correspond to 90% and 80% of the training samples being used for the training respectively.

As can be seen in figure 5, the use of the iterative trimmed loss minimization resulted in overfitting of the model with the tested parameters. This setup was not tested further due to time limitations and it could be further tested with more training epochs and other values of α, to determine its usefulness more precisely.

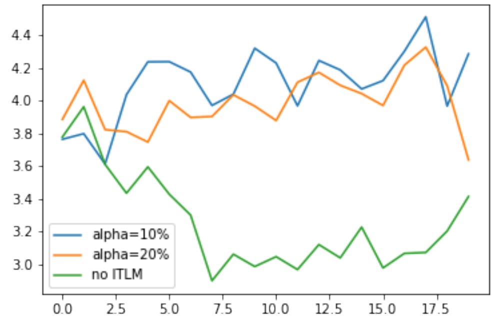


Figure 5. Validation mean squared errors per each training epoch with and without the ITLM training configurations.

RESULTS

As described in the beginning of this report, the model was trained using 80% of the smallest unlabelled A2D2 dataset and tested with the rest 20%. The resulting errors for each tested training setup are shown in the table 6. The results show that the model only learnt any useful features from the data when presented with the data augmentations without the usage of the iterative trimmed loss minimization.

Training Setup	Error (Radians)	Error (Degrees)
No augmentations, no ITLM	1.1717	67.13
Augmentations, no ITLM	**0.8685**	**49.76**
Augmentations, ITLM, $\alpha = 10\%$	1.0549	60.44
Augmentations, ITLM, $\alpha = 20\%$	1.0149	58.15

Figure 6. Mean absolute errors over the validation set. Best configuration in bold.

The Audi Q7 e-tron which is used for gathering the data has a steering ratio of 15.8. Thus, the average absolute steering error of 49.76 degrees obtained by our best training setup corresponds to around 3.15 degree wheel angle error. In very low speeds seen in crossroad areas in urban driving, this error is fairly small but for highway driving errors of this magnitude would be huge and very dangerous. Thus, any of the resulting models of this study could not be directly used in real road testing and further development would have to be made.

DISCUSSION

Even though the results of this study alone were fairly poor and showed no success in training a well performing model with the given dataset alone, this topic could be studied further and none of the results alone mean that the problem would be impossible only very difficult with multiple possible approaches.

Possibilities for further studies include some of the following topics: other training algorithms with similar goals as the iterative trimmed loss minimization, further tuning of this particular setup, further testing of the data augmentation methods, which were also succesfully used in this study with fairly small testing, training of the model with a combination of multiple different datasets, finding ways to filter the dataset autonomously before the actual training of the model and so forth. Thus, the only conclusion that can be made from this study alone is that the original research question remains open and further studies would be required.

REFERENCES

Bojarski, M., Testa, D. D., Dworakowski, D., Firner, B., Flepp, B., Goyal, P., Jackel, L. D., Monfort, M., Muller, U., Zhang, J., Zhang, X., Zhao, J., and Zieba, K. (2016). End to end learning for self-driving cars. *CoRR*, abs/1604.07316.

Geyer, J., Kassahun, Y., Mahmudi, M., Ricou, X., Durgesh, R., Chung, A. S., Hauswald, L., Pham, V. H., Mühlegg, M., Dorn, S., Fernandez, T., Jänicke, M., Mirashi, S., Savani, C., Sturm, M., Vorobiov, O., Oelker, M., Garreis, S., and Schuberth, P. (2020). A2D2: Audi Autonomous Driving Dataset.

Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization.

Shen, Y. and Sanghavi, S. (2018). Learning with bad training data via iterative trimmed loss minimization.

Yang, Z., Zhang, Y., Yu, J., Cai, J., and Luo, J. (2018). End-to-end multi-modal multi-task vehicle control for self-driving cars with visual perception.