Imitation Learning of Path-Planned Driving Using Disparity-Depth Images

Sascha Hornauer(✉), Karl Zipser, and Stella Yu

International Computer Science Institute, University of California, Berkeley, USA
sascha.hornauer@icsi.berkeley.edu
{karlzipser,stellayu}@berkeley.edu

Keywords: End-to-End training · Autonomous driving
Path planning · Collision avoidance · Depth images · Transfer learning

1 Introduction

Sensor data representation in autonomous driving is a defining factor for the final performance and convergence of End-to-End trained driving systems. When theoretically a network, trained in a perfect way, should be able to abstract the most useful information from camera data depending on the task, practically this is a challenge. Therefore, many approaches explore leveraging human designed intermediate representations as segmented images. We continue work in the field of depth-image based steering angle prediction and compare networks trained purely on either RGB-stereo images or depth-from-stereo (disparity) images. Since no dedicated depth sensor is used, we consider this as a pixel grouping method where pixel are labeled by their stereo disparity instead of relying on human segment annotations. In order to reduce the human intervention further, we create training data from driving, guided by a path planner, instead of using human driving examples. That way we also achieve a constant quality of driving without having to limit data collection to exclude the beginning of a human learning curve. Furthermore, we have fine control over trajectories, i.e. we can set and control appropriate safety distances and drive the shortest feasible path.

With this methodology we approach the problem of training a network-based driver to find and traverse free space (free-roaming) in novel environments based on very little and easy to create training data. By using disparity images as perceptual organization of pixels in stereo images, we can create obstacle avoiding driving behavior in complex unseen environments. Disparity images reduce differences in appearance in between environments heavily and can be produced on our current embedded platform, the NVIDIA Jetson-TX1, in real-time.

Related Work: Network based autonomous driving can be distinguished in different directions: (1) Traditional approaches analyze the sensory input and develop a catalogue of path planning and driving policies under various scenarios [6,9,15]. Such approaches require a lot of engineering efforts and are often brittle in real applications. (2) Reinforcement learning approaches allow the model car

L. Leal-Taixé and S. Roth (Eds.): ECCV 2018 Workshops, LNCS 11133, pp. 542–548, 2019.
https://doi.org/10.1007/978-3-030-11021-5_33

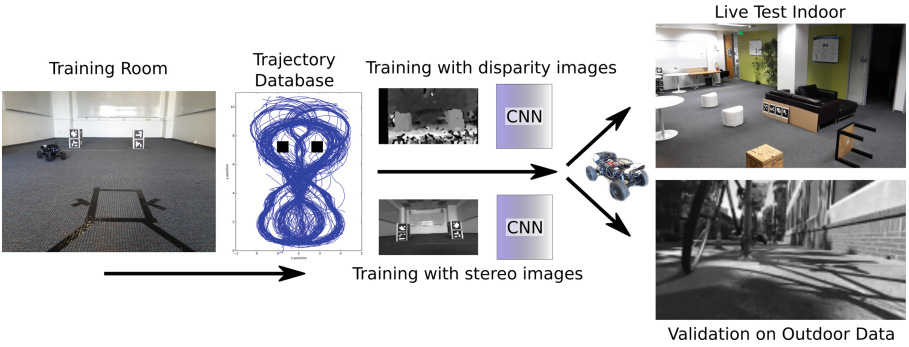


Fig. 1. Overview of the method: Data from training is used to train the same network in two different ways. Free roaming, based on disparity and stereo images, is evaluated in a novel cluttered room. Additionally, the performance with recorded images of outdoor driving is investigated.

to discover a driving policy online through trial and error [8,10,12]. However, such approaches are often sample-inefficient and because continuous crashing is part of the process, research is often supported by simulation. (3) Data-driven learning approaches that predict actions from visual input directly [1–3,5,11, 13,16]. With the availability of big data, computing power and deep learning techniques, deep-net based End-to-End driving systems become a major research direction. Comparable work on Behavioral Cloning of steering and motor signals for model cars shows that it is possible to train a convolutional neural networks (CNNs) to learn navigation in various terrains [2,3].

A similar approach uses a path planner in simulation to navigate to goals [14]. They apply their algorithm in the real world to navigate based on 2D-laser range findings. An external goal position and extracted features from a CNN are fused into their final fully connected layers to produce steering and motor commands towards that goal. A motion planner is used as expert to train the network though this is performed in a deterministic simulation. In contrast, we see advantages in real world training as the state progression of a driving model car is probabilistic, allowing for natural exploration of the state space, without having to add artificial noise as in simulation.

2 Method

For data collection, we let a Dubins-model based path planner with ground truth position information drive model cars on randomized trajectories to preset waypoints on a fixed map, as seen in Fig. 3 (left). Disparity images, created from incoming camera data, are used as representation of the input scene during training and testing (SGM stereo method [4], blocksize 5, number of disparities 16). We selected these parameters through hand-tuning, to achieve high details in the distance and less noise, while we tolerate remaining errors in

depth-reconstruction. In still images, seen in Fig. 2 the noise seems to be large though additional filtering did not lead to changes in driving performance though slowed down disparity image generation. Our hypothesis is that this representation generalizes well enough to learn collision avoidance with a very reduced training regime: We collect the path planner examples of driving away from walls and simple obstacles and train a network to predict the planner's steering commands using the recorded disparity images.

Fig. 2. Left to right: Stereo image from driving in a park. Reconstructed disparity image from the same scene. Image from the data collection room for training. Reconstructed disparity image with noise on the ground.

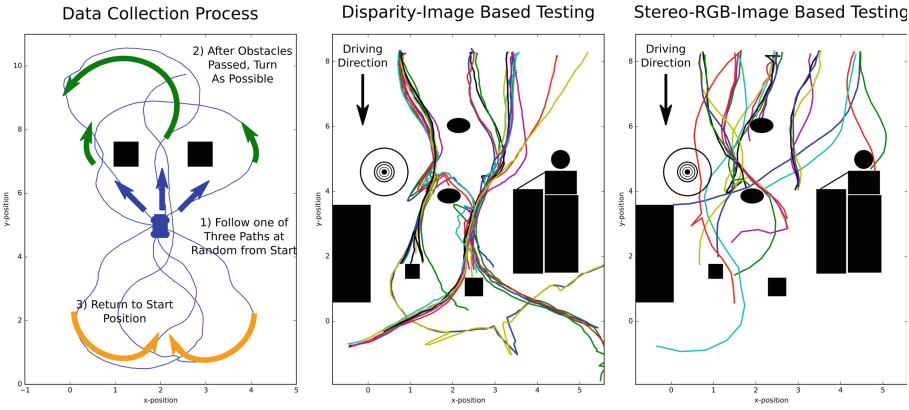


Fig. 3. Driving for data collection (left image) and comparison of trajectories in test-environment. Obstacles are shown as black forms. Some errors of the localization system are visible as short spikes. Driving direction in the evaluation (center and right image) is from the top to the bottom.

2.1 Network Training

With the collected data we train two networks for comparison, equal in design apart from the size of the input layer. Figure 1 shows an overview of the method and our cluttered test-room in the top right corner. Each network is based on SqueezeNet, as developed by [7], designed for image classification which performs well on our embedded platform. We removed the final Softmax layer and use the network for steering angle regression. In order to take temporal correlation over frames into account we concatenate frames over 10 time-steps. Single

RGB-camera images are 94×168 px and the input to the network therefore $3 \times 2 \times 10 \times 94 \times 168$ for stereo or $10 \times 94 \times 168$ for depth-images. The output is a vector of 10 steering commands over 10 time-steps, predicted by regression using a 2d convolution with 10 kernels. Only one steering command is used, though 10 are generated and compared against the ground truth path-planner steering using Mean Squared Error loss. Using 10 favours the prediction of entire trajectories over single points of control through the car and follows the motivation of leveraging side-tasks [2,17]. The speed is fixed and learning is performed in PyTorch with *Adam* optimization. Best generalization was achieved after training only one epoch on approximately 7 h of training data.

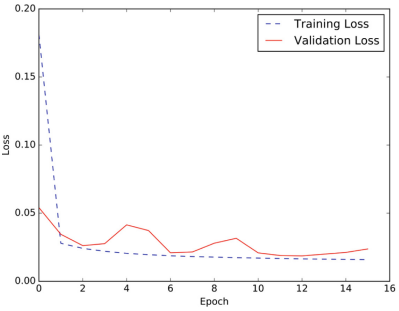
3 Experimental Evaluation

During test time the model car traverses a novel cluttered room in several trials from different starting positions. Each trial ends once the car reaches the other side of the room or collides. The average length of trajectories is compared for depth- and stereo-image based driving and they are shown in Fig. 3 (middle and right image). Even though stereo-image based driving shows successful avoidance manoeuvres, the number of collisions is higher. Table 1 shows longer average (54% more) and individual (60% more) trajectory driven by the depth-image trained model. In addition to the cluttered office space we tested the indoor-trained models on previously recorded outdoor data on sidewalks to test the generalization properties further. While in the office space no label for the test data exists, outdoor steering and throttle labels from human drivers were recorded. Validation loss of these experiments is shown in Fig. 4b.

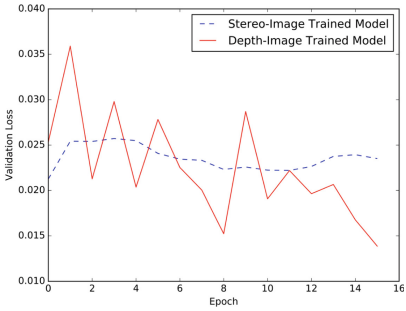
Table 1. Results of depth-image against stereo image-based driving, compared by the achieved trajectory length with given standard deviation σ in different environments. Stage refers to the *Stage* simulator with simple, complex and real being different maps

Images type used/Test environment	# Trajectories driven	Avg. length	σ Length	Longest trajectory
Stereo/Cluttered room	24	5.32 m	2.22 m	11.23 m
Depth/Cluttered room	28	**9.78** m	3.09 m	**18.80** m
Depth/Stage simple	20	7.44 m	4.03 m	10.97 m
Depth/Stage complex	20	5.73 m	4.32 m	11.67 m
Depth/Stage real	20	3.63 m	3.35 m	11.03 m

Additionally, we tried to compare our result to [14] even though in their use case they provide a goal during training and test time. In order to compare roaming across their maps, we measured the farthest distance traveled from start positions at edges of the maps, seen in Fig. 5. Our model drove without any re-training with the depth-image provided by the simulation.



(a) Training results when training with disparity images



(b) Outdoor video-validation, comparing steering prediction using each visual representation against human steering decisions.

Fig. 4. Shown are MSE training- and validation-loss. Indoor training (a) shows good convergence and robustness against over- and under-fitting. Evaluation of the additional test of the trained model on outdoor video-data is shown in (b). In later epochs the depth-image based method outperforms stereo-based steering angle prediction.

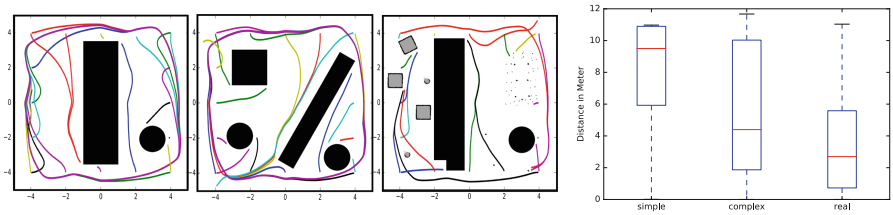


Fig. 5. *Stage*-Simulator experiments. Start positions of trajectories are along the edges in maps called **Simple**, **Complex** and **Real** (FLTR) with the driving-distance compared on the right. On each map the other side is reached though less often with increasing complexity.

4 Conclusion

With approximately 7 h of driving examples in a simple room, our model car demonstrates good driving performance in an unseen cluttered office environment, avoiding collisions with novel obstacles. We showed the information gained from disparity images, inferred from RGB-stereo images, are not only sufficient to navigate the model car but generalize better when predicting steering angles. This enables to leverage a path planner, driving in a room with very sparse visual features, to create enough expert examples so human intervention can be limited to get the car unstuck approximately once every hour. As future work we created training data in the Carla simulator and will compare the two image-based driving methods to quantify our results.

References

1. Bojarski, M., et al.: End to End Learning for Self-Driving Cars. arXiv preprint arXiv:1604.07316, pp. 1–9 (2016)
2. Chowdhuri, S., Pankaj, T., Zipser, K.: Multi-Modal Multi-Task Deep Learning for Autonomous Driving. arXiv preprint arXiv:1709.05581 (2017)
3. Codevilla, F., Müller, M., Dosovitskiy, A., López, A., Koltun, V.: End-to-end Driving via Conditional Imitation Learning. arXiv preprint arXiv:1710.02410 (2017). To be published in proceedings - IEEE International Conference on Robotics and Automation (2018)
4. Hirschmuller, H.: Accurate and efficient stereo processing by semi-global matching and mutual information. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 2, pp. 807–814 (2005). https://doi.org/10.1109/CVPR.2005.56
5. Hou, E., Hornauer, S., Zipser, K.: Fast Recurrent Fully Convolutional Networks for Direct Perception in Autonomous Driving. arXiv preprint arXiv:1711.06459 (2017)
6. Hubmann, C., Becker, M., Althoff, D., Lenz, D., Stiller, C.: Decision making for autonomous driving considering interaction and uncertain prediction of surrounding vehicles. In: Proceedings of the IEEE Intelligent Vehicles Symposium (IV), pp. 1671–1678 (2017). https://doi.org/10.1109/IVS.2017.7995949
7. Iandola, F.N., Han, S., Moskewicz, M.W., Ashraf, K., Dally, W.J., Keutzer, K.: SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <0.5 MB model size. In: Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) vol. 9351, pp. 324–331 (2016). https://doi.org/10.1007/978-3-319-24574-4_39
8. Kahn, G., Villaflor, A., Ding, B., Abbeel, P., Levine, S.: Self-supervised deep reinforcement learning with generalized computation graphs for robot navigation. In: Proceedings of the IEEE International Conference on Robotics and Automation (2018)
9. Kong, J., Pfeiffer, M., Schildbach, G., Borrelli, F.: Kinematic and dynamic vehicle models for autonomous driving control design. In: 2015 IEEE Intelligent Vehicles Symposium (IV), pp. 1094–1099. IEEE (2015). https://doi.org/10.1109/IVS.2015.7225830
10. Kuderer, M., Gulati, S., Burgard, W.: Learning driving styles for autonomous vehicles from demonstration. In: Proceedings of the IEEE International Conference on Robotics and Automation, pp. 2641–2646 (2015). https://doi.org/10.1109/ICRA.2015.7139555
11. LeCun, Y., Muller, U., Ben, J., Cosatto, E., Flepp, B.: Off-road obstacle avoidance through end-to-end learning. In: Advances in Neural Information Processing Systems, vol. 18, p. 739 (2006)
12. Mirowski, P., et al.: Learning to Navigate in Complex Environments. Accepted for poster presentation ICRL 2017 (2016)
13. Pan, Y., et al.: Agile off-road autonomous driving using end-to-end deep imitation learning. In: Robotics: Science and Systems 2018 (2018). https://doi.org/10.15607/RSS.2018.XIV.056, http://arxiv.org/abs/1709.07174
14. Pfeiffer, M., Schaeuble, M., Nieto, J., Siegwart, R., Cadena, C.: From perception to decision : a data-driven approach to end-to-end motion planning for autonomous ground robots. In: 2017 IEEE International Conference on Robotics and Automation (ICRA), pp. 1527–1533. IEEE, Singapore (2017). https://doi.org/10.1109/ICRA.2017.7989182

15. Plessen, M.G., Bernardini, D., Esen, H., Bemporad, A.: Spatial-based predictive control and geometric corridor planning for adaptive cruise control coupled with obstacle avoidance. IEEE Trans. Control Syst. Technol. **26**(1), 38–50 (2018). https://doi.org/10.1109/TCST.2017.2664722
16. Pomerleau, D.a.: Alvinn: an autonomous land vehicle in a neural network. In: Advances in Neural Information Processing Systems, vol. 1, pp. 305–313 (1989)
17. Xu, H., Gao, Y., Yu, F., Darrell, T.: End-to-end learning of driving models from large-scale video datasets. In: 2017 Proceedings of Conference on Computer Vision and Pattern Recognition, pp. 2174–2182 (2017). https://doi.org/10.1109/CVPR.2017.376