A Survey on Deep Learning Techniques for Autonomous Vehicles

Matt Strong[1], Greg Lund[1], Byungjin Kim[1], Brad Hayes[1]

***Abstract*— Creating controllers for autonomous vehicles is proving to be more and more challenging as the complexity of environments increases. To tackle this problem, deep learning methods are gaining more traction by enabling nonlinear learning and generalization for autonomy. In this paper, we present a brief survey on the state of the art (SOTA) deep learning techniques for autonomous control. While the field of autonomous vehicles encompasses a multitude of deep learning techniques, in this work, we solely address deep learning directly in the control setting. We draw upon detailed work in a recent survey that addresses the current state of and issues with deep learning for autonomous vehicle control [1]. We will address past, current, and proposed future ideas in the field, along with a section addressing how these techniques can be applied in a practical setting (the CSCI 4302 Final Challenge).**

I. Introduction

Every year, traffic accidents in the US result in thousands of deaths – 90 percent of which are estimated to have been human errors, while only 2 percent comes from vehicle-related failures [2]. Road traffic causes are the leading cause of death for people aged 1-54 in the United States. Furthermore, over 1.3 million people are killed on roadways a year. To put this into perspective, one person is killed every 25 seconds [3].

In order to combat this problem, autonomous vehicles have emerged as a powerful solution. Due to the nature of autonomy, safety is critical at all times and is the highest priority. A well-designed autonomous vehicle controller has the potential to not only greatly reduce traffic accidents, but also traffic flow.

II. Background

Before the deep learning explosion, there have been numerous attempts, with varying degrees of success, to address the problem of autonomous vehicle control.

Many attempts have been very reliant on sensor data, and they utilize rather expensive sensor systems (such as Lidar) along with fine-tuned controllers for specific situations. These rule-based controllers make generalization difficult, and tuning can be very finnicky, and must be handled quite differently in simulation and reality. Another difficulty faced in autonomous systems is its non-linearity, in which linear approximations and other related analytical solutions often do not work well.

Recent developments in deep learning, especially image recognition and speech recognition, have led to others finding others avenues to apply deep learning too.

[1]Robotics, Computer Science Department, University of Colorado, Boulder, CO 80309, USA `name.surname@colorado.edu`.

With deep learning research growing at a rapid rate, those involved in autonomy have been able to enjoy the benefits of such research – employing deep convolutional neural networks for object detection, instance segmentation, and semantic segmentation. However, this doesn't address the control problem. Some solutions have utilized CNNs directly from raw images to control outputs, eliminating the need for a perception module and instead learning end to end [4].

But, perhaps the largest benefit of deep learning in autonomous vehicle control is the powerful function approximation abilities and generalization in different environments. Given these abilities, deep learning has begun to flourish in autonomy, often outperforming classical control methods. We now go into a more specific overview of recent and current deep learning methods.

Broadly speaking, deep learning for autonomy can be broken up into supervised learning and reinforcement learning. For the purpose of this survey, we assume the reader is sufficiently familiar with the basics of each technique. From here, vehicle control can be broken up into lateral motion (steering), longitudinal motion (brake and gas on a vehicle), and a combination of both.

Fig. 1: ALVINN: One of the first examples of neural network controlled vehicles.

III. Lateral Control

Regarding lateral control systems, one of the first attempts to solve the problem was the Autonomous Land Vehicle in a Neural Network (ALVINN) by Pomerleau which utlized a standard feedforward neural network, predicting a discrete steering action [5]. The method used data augmentation and a buffering system to ensure that the data was not biased to the left or right. The method was trained on a 150m stretch of rate, with steering accuracies twice as accurate as humans. Later, Yu applied reinforcement learning to learn a controller via collected experiences [6]. Moriarty combined both supervised learning and reinforcement learning for lane selection [7].

These models were small, and after the computational power of GPUs significantly rose, Muller et. al proposed a system that learned a multi-layer CNN with two cameras through a human controlling a vehicle (and simultaneously avoiding obstacles) at 2m/s [8]. Later, NVIDIA built on top this approach; there was no need for perception of obstacles – the network learned them [4].

Q-Learning has made an appearance in lateral control; Wang et al. using RL to train a Q network [9]. Their network used a variety of information about the state of the car in order to output the desired yaw acceleration. While tested in simulation, the results were promising.

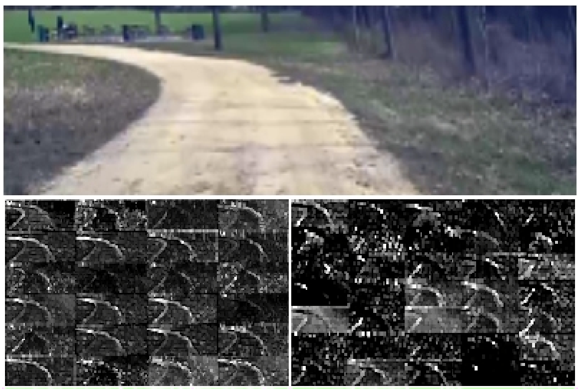

Fig. 2: Nvidia's convolutional neural network visualizations [4]. The network learns to distinguish the path from the vegetation.

IV. Longitudinal Control

Regarding *longitudinal* control, while ACC (adaptive cruise control) is a common choice, it has difficulty working in highly non-linear, high-dimensional situations. Deep reinforcement learning has emerged as a strong solution; Dai et. al was one of the first researchers to utilize RL; through using a Q estimator network (to estimate the best action value function) and a Fuzzy Inference System which gets the control output based on that Q estimator network [10]. They applied their method in a simulated setting for following a car – they varied the velocity of the leading vehicle, and provided reward based on the distance between the RL car and the leading car. Later, Huang et al. proposed a novel Parameterized Batch Actor-Critic (PBAC) reinforcement learning algorithm that utilized a multi-objective reward function to accommodate for safety [11]. This reward function was limited for adjacent vehicles, however. One reward function proposed by Chae et al. proposed a collision avoidance method using DQN [12]. The reward function focused on avoiding obstacles and getting out of high-risk situations. In addition, Chen et al. proposed an Adaptive Cruise Control method using human demonstration with Q-learning [13]. This reward function combined distance from the ahead vehicle, speed, and acceleration.

Due to the time intensive manner of reinforcement learning algorithms, supervised learning has been combined with RL. Wang et al. learned a hierarchical controller in an emergency breaking situation [14].

Finally, work has been done combining both latitude and longitudinal control. Zhang et al. proposed a method called SafeDAgger, using the DAgger imitation learning algorithm [15]. The policy learned to predict a continuous steering angle and a binary decision for breaking or not.

Inverse reinforcement learning, in which a reward function is learned, has also garnered interesting. Abbeel and Ng showed that when learning from expert actions, a learned reward function can achieve similar results as a potentially different reward function [16]. An experiment of 5 different actions for lane selection demonstrated this. However, the computational burden needed to learn the optimal reward function is heavy, and Hecker et al. completely turned previous ideas on their head through a 360 degree camera system using a DNN to predict both velocity and steering angle [17]. Finally, another work that broke the mold was ChaffeurNet, which pre-processed the inputs of the environment, then feeding them into a RNN for speed, heading, and waypoint, which was found into a standard controller [18]. This method was not as ”end to end” as previous methods, but eliminated some of the concerns of the transfer of simulation to reality; the pre-processing minimized the distance between simulated and real inputs, making transfer easier.

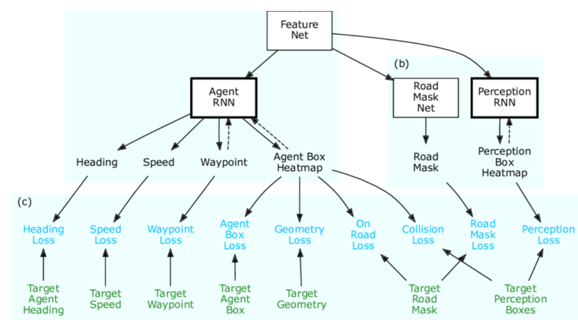

Fig. 3: The architecture of ChaffeurNet. It does not treat the control+perception problem as one; it utilizes different networks to achieve desirable results.

V. How We Can Use This

As for how these methods can be utilized into the project, the works that utilize a CNN for end to end control are of most interest to us (we will elaborate why later in this section). The research involving distance to other cars is not relevant for the solo car challenge. While the distance between a leading vehicle is important in more dangerous environments, our task does not involve the presence of other vechiles, so we have no need for those kinds of techniques. Additionally, we could look at the ChaffeurNet, which used preprocessing of the environment before feeding into a RNN. However, we are not as concerned with the transfer of simulation to reality, as we are exclusively working in simulation. Additionally, the extra work for preprocessing may not be fully worth it in the more simple environment that we are working in. Inverse reinforcement learning is also a possible option; however, the time it takes to find a near-optimal reward function may be too great compared to a standard supervised CNN. This brings us to the best option that we might be able to apply – end to end learning of visuomotor policies. Data collection may take some time (which is the main drawback), but previous work has shown that such kind of work is able to learn efffective policies over long distances, even learning to avoid obstacles.

REFERENCES

[1] S. Kuutti, R. Bowden, Y. Jin, P. Barber, and S. Fallah, "A survey of deep learning applications to autonomous vehicle control," *IEEE Transactions on Intelligent Transportation Systems*, 2020.

[2] "Fatality facts 2018: State by state." [Online]. Available: https://www.iihs.org/topics/fatality-statistics/detail/state-by-state

[3] "Road traffic injuries." [Online]. Available: https://www.who.int/news-room/fact-sheets/detail/road-traffic-injuries

[4] M. Bojarski, M. Bojarski, B. Firner, B. Flepp, L. Jackel, U. Muller, K. Zieba, and D. D. Testa, "End-to-end deep learning for self-driving cars," Aug 2020. [Online]. Available: https://developer.nvidia.com/blog/deep-learning-self-driving-cars/

[5] D. A. Pomerleau, "Alvinn: An autonomous land vehicle in a neural network," in *Advances in neural information processing systems*, 1989, pp. 305–313.

[6] G. Yu and I. K. Sethi, "Road-following with continuous learning," in *Proceedings of the Intelligent Vehicles' 95. Symposium*. IEEE, 1995, pp. 412–417.

[7] D. E. Moriarty, S. Handley, and P. Langley, "Learning distributed strategies for traffic control," in *From Animals to Animats: Proceedings of The International Conference on Simulation of Adaptive Behavior (SAB)*, 1998, pp. 437–446.

[8] U. Muller, J. Ben, E. Cosatto, B. Flepp, and Y. Cun, "Off-road obstacle avoidance through end-to-end learning," *Advances in neural information processing systems*, vol. 18, pp. 739–746, 2005.

[9] P. Wang, C. Chan, and A. de La Fortelle, "A reinforcement learning based approach for automated lane change maneuvers," *CoRR*, vol. abs/1804.07871, 2018. [Online]. Available: http://arxiv.org/abs/1804.07871

[10] X. Dai, C.-K. Li, and A. B. Rad, "An approach to tune fuzzy controllers based on reinforcement learning for autonomous vehicle control," *IEEE Transactions on Intelligent Transportation Systems*, vol. 6, no. 3, pp. 285–293, 2005.

[11] Z. Huang, X. Xu, H. He, J. Tan, and Z. Sun, "Parameterized batch reinforcement learning for longitudinal control of autonomous land vehicles," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 49, no. 4, pp. 730–741, 2017.

[12] H. Chae, C. M. Kang, B. Kim, J. Kim, C. C. Chung, and J. W. Choi, "Autonomous braking system via deep reinforcement learning," in *2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2017, pp. 1–6.

[13] X. Chen, Y. Zhai, C. Lu, J. Gong, and G. Wang, "A learning model for personalized adaptive cruise control," in *2017 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2017, pp. 379–384.

[14] B. Wang, D. Zhao, C. Li, and Y. Dai, "Design and implementation of an adaptive cruise control system based on supervised actor-critic learning," in *2015 5th International Conference on Information Science and Technology (ICIST)*. IEEE, 2015, pp. 243–248.

[15] J. Zhang and K. Cho, "Query-efficient imitation learning for end-to-end autonomous driving," *arXiv preprint arXiv:1605.06450*, 2016.

[16] P. Abbeel and A. Y. Ng, "Apprenticeship learning via inverse reinforcement learning," in *Proceedings of the twenty-first international conference on Machine learning*, 2004, p. 1.

[17] S. Hecker, D. Dai, and L. Van Gool, "End-to-end learning of driving models with surround-view cameras and route planners," in *Proceedings of the european conference on computer vision (eccv)*, 2018, pp. 435–453.

[18] M. Bansal, A. Krizhevsky, and A. Ogale, "Chauffeurnet: Learning to drive by imitating the best and synthesizing the worst," *arXiv preprint arXiv:1812.03079*, 2018.