Scholar Publishing Group
International Journal of Neural Network
https://doi.org/10.38007/NN.2020.010301
ISSN 2790-2692 Vol. 1, Issue 3: 1-8

Deep Learning in Autonomous Driving

Qilu Zhang and Hu Shen*

Shandong Institute of Commerce and Technology, Jinan, China

**corresponding author*

Keywords: Deep Learning, Reinforcement Learning, Feature Network, Unmanned Driving

Abstract: Autonomous driverless technology, as an emerging technology of cross-integration of multiple fields, can control vehicles through autonomous intelligent control strategies, bring more safe and reliable driving environment and efficient driving scheme, which has very important research significance for improving the safety and efficiency of real road traffic. This paper mainly studies the application of deep learning in autonomous driving. This paper first analyzes the deep reinforcement learning method, and proposes an ODDPG autonomous driving decision method based on supervised training feature network for the real complex road autonomous driving decision problem. The feature extraction network is obtained in advance from imitation learning to improve the feature sample collection efficiency of reinforcement learning. ODDPG algorithm shows faster learning speed and better reward distribution of interactive data in the training process.

1. Introduction

In the course of history, as soon as the automobile, a machine, was invented, it quickly occupied the main territory of human way of travel, and played a revolutionary role in passenger transport, logistics, freight transportation and other aspects. In the era of modernization and information technology with the progress of science and technology, automobiles are gradually moving from simple mechanical component structure to electronic and information modern mode [1-2]. As sensors continue to be invented, improved, and deployed. Cars are getting smarter. But as time went on, people had higher expectations for cars, namely autonomous driving. The continuous development of hardware technology, including computing power, sensors such as cameras, radar and GPS, as well as the help of high-speed and high-frequency communication, makes the expectation of autonomous driving gradually come to reality [3]. In the current automatic driving solution, the common functional modules can be divided into decision making problem, planning problem and perception problem. Among them, the perception problem is the basis of the rest of the problems, and the excellent algorithm can be said to be completely based on the correct perception and judgment of the surrounding environment. In the perception problem, the commonly used sensors are millimeter wave radar, liDAR, camera and so on. Multiple groups of sensors together constitute a complete sensing system. Sensors can also be selected according to the actual situation under different circumstances. For example, Tesla prefers to use cameras as vision sensors, while Huawei prefers to use liDAR. In the perception task, common perception problems include vehicle detection, pedestrian detection, traffic sign and signal light detection, lane detection, semantic segmentation and other tasks. The source of sensor data can be basically divided into camera visual data, liDAR point cloud data and so on. As the application scenarios become more and more complex, the demand for automatic driving level becomes higher and higher, the required perceptual problems become more and more complex, and the complexity and functional requirements of the perceptual model increase gradually [4-5]. But with the increase in hardware computing power, this feature has gradually become a possibility. In perception problems, accuracy, rapidity, and robustness in the special environment of inclement weather have gradually become the main research content in academia and industry.

As early as the 1950s, some developed countries began to conduct research on driverless driving, and the research results are relatively mature. The world's first driverless car was developed by the American Barrett Electronics Company, marking the beginning of the transition to intelligence in the automobile industry [6]. The U.S. Department of Defense has held three DAPRA Urban Challenges, in which "Dust Storm" developed by Carnegie Mellon University, Stanley led by Stanford University, and Boss co-manufactured by Carnegie Mellon and General Motors won the first place three times, respectively. In these competitions, many unmanned teams have gradually developed [7]. Tesla launched the Autopilot semi-autonomous driving system and accumulated driving mileage in the Autopilot mode reached 1.88 billion miles in the past two years [8]. Uber conducted the trial operation of driverless taxis on the main roads of Pittsburgh in the United States, and began to pick up real passengers [9]. Audi A82018 came into the market, which was the first L3 class driverless car that really reached mass production [10].

Based on the reinforcement learning theory, this paper proposes a set of autonomous driving behavior decision-making system based on deep reinforcement learning, and explores how to improve the exploration efficiency in the process of algorithm learning and how to make autonomous vehicles effectively learn in different scenes. This will be of great significance to improve the safety, comfort and stability of the unmanned driving system, so that the end-to-end system can really be realized.

2. Autonomous Driving Based on Deep Reinforcement Learning

2.1. Deep Reinforcement Learning Method

(1) Deep reinforcement learning method based on value function

In deep reinforcement learning, it is divided into value-based methods and policy-based methods according to the different ways in which deep neural network approximates nonlinear functions [11].

The most important DQN method of value-based DRL method is proposed by the team of Google. It uses the principle of traditional Q_learning algorithm, uses neural network to output the target Q, and sets two networks with the same structure but different update steps, one main network. A target network (Targer network) and the experience replay mechanism is used to train the algorithm, which improves the applicability of the reinforcement learning algorithm [12]. DQN uses three core technologies: objective function, Targer network, and experience playback mechanism. The algorithm optimization goal of deep network is to minimize the difference between the real Q and the estimated Q. By using a large number of input samples, the network parameters are fitted after training [13].

(2) Policy-based deep reinforcement learning method

In view of the continuous action space, using the method based on value function cannot solve, because in the process of strategy to improve, the method to solve the action value function, thus solving the optimal action, the need for each state requires a V value, calculate the certain state of the corresponding behavior is impractical, strategy needs to be parameterized directly, The strategy is represented by linear and nonlinear functions, approximated by neural network, and the optimal strategy is solved by policy gradient method [14-15].

Compared with the value function method, the strategy gradient method has better convergence, easier strategy exploration, and can learn the advantages of random strategies [16]. It is generally divided into stochastic strategy gradient method and deterministic strategy gradient method. In order to find the optimal policy, the stochastic policy is solved by parameters and states:

$$\pi_\theta(a \mid s) = P(a \mid s; \theta) \tag{1}$$

Distribution strategy is obtained by neural network training PI theta (a | s), in a known state s, training action a fitting can be obtained through a network of probability, theta is the network parameters. The stochastic policy gradient method can find the corresponding probability of each action, and its update formula is as follows:

$$\nabla_\theta J(\theta) = E_{s-\mu,a-\mu}[\nabla_\theta \log \pi_\theta(a \mid s,\theta) Q_\pi(s,a)] \tag{2}$$

Compared with the introduction of sample data required for the deterministic strategy, the algorithm is more efficient. This method reduces the sampling integral in the probability distribution space of the strategy, which is conducive to finding the optimal strategy in the continuous action space [17]. Because the policy gradient algorithm has the problem of low efficiency when it is updated, and it is easy to fall into local optimum in infinite action space.

Actor-critic (AC) method, which uses the approximate value function to guide the process of policy updating, greatly realizes the convergence of finding the optimal process. Critics approximate method is used to update action value function Q PI (s, a), action network using the method of random strategy under the guidance of commenting on the network, the stochastic gradient PI (a | s theta) update [18].

2.2. Autonomous Driving Decision Making Based on ODDPG

In order to solve the problem of low efficiency of reinforcement learning applied to complex task samples, the algorithm in this paper combines imitation learning and deep reinforcement learning to improve the decision-making ability of the model. The feature extraction network is obtained in advance from imitation learning to improve the efficiency of feature sample collection in reinforcement learning, so as to obtain a more general robust strategy. Firstly, imitation learning is used to train the feature network model to obtain the feature extraction part of the network model.

DDPG (Deep Deterministic PolicyGradients) autonomous driving algorithm based on supervised training feature network is mainly composed of two parts: The feature network is pre-trained by supervised training of imitation learning, and DDPG algorithm is trained by deep reinforcement learning on CARLA platform.

(1) Supervised training feature network

The feature extraction network is obtained in advance from imitation learning to improve the sample efficiency of reinforcement learning. Firstly, the official data set of Carla and expert data are used to train the end-to-end decision network model based on imitation learning, and the initial weight of the feature extraction network is taken. Depth using supervised learning the way of training a neural network, it is a sensor input is mapped to the driving instruction process, this article to mimic RGB images are acquired by monocular camera as network input, through neural network feature extraction, to full connection layer, in turn, classified as accelerator, such as the steering wheel steering instructions, by imitating the end-to-end training study.

Imitation learning is regarded as a neural network with discrete time steps interacting with the environment, and the training data is an array of expert data. Each walk length t has its corresponding observation value dt and action value at. The neural network is trained to imitate the command output of the expert.

The feature extraction network consists of 8 convolutional layers and 2 fully connected layers. Dropout and batch normalization are performed after the convolutional layer to prevent overfitting and improve the robustness of neural networks for data loss processing. The ReLu activation function is used to enhance the nonlinearity after feature extraction in each layer. The fully connected layer is composed of a standard fully connected layer, Dropout layer and ReLu layer.

(2) ODDPG algorithm

DDPG algorithm is a deep reinforcement learning model of continuous action domain. It uses convolutional neural network to construct policy function μ and Q function, and then forms policy network and Q network. DDPG network structure includes Critic network, Actor network and experience playback pool, Critic and Actor network include target network and current network respectively.

Actor Current network: Update and iterate the policy network parameter θ, select the current action AI according to the current real-time state s, and then interact with the environment to generate Si +1 and R.

Actor target network: According to the update frequency, the network parameter θμ is copied as θμ', and the next optimal action A 'is generated according to the next state Si +1 sampled by the empirical playback pool. When updating θμ', a smoothing coefficient of less than 1 is set in a smooth way to ensure the difference of updating.

Critic Current network: Update and iterate the value network parameter θw, calculate the Q value Q (s,a,θ) under the current state, and the current target value function is determined by the target Q' value Q (s ', a ',θ ').

Critic target network: copy the network parameter θw to θw' according to the update frequency, and calculate the target value Q '(s ', a ', θw'). The same smoothing coefficient is used for differentiated update as the Actor target network.

The experience playback pool stores the process data of exploration and uses the stored historical experience data to randomly select multiple groups to train the network.

In this paper, the ODDPG algorithm is based on CARLA as a simulation interactive environment for model training. According to the current action AI output by the Actor's current network, the interaction environment CARLA generates the next state Si +1 according to the state transition, and then generates the current reward value RI through the reward function calculation. Therefore, the design of reward function is very important.

3. Unmanned Driving Simulation Experiment

3.1. Experimental Scene Setting

In this paper, CARLA version 0.9.6 was selected as the simulation environment for the development and verification of autonomous vehicle agents. Based on the deep reinforcement learning algorithm IDDPG designed in this paper, the end-to-end decision algorithm of autonomous vehicle agents in the direction of reinforcement learning was studied and verified by experiments. In this paper, the design of unmanned vehicle agent driver's goal is to not leave the driveway and avoid collision cases, intelligent vehicle body along the lane center drive and finally reach the destination location, pay attention to during the driving without overtaking, take the initiative to change lanes, such as active driving behavior, and the path mainly considering other participant behavior under normal traffic conditions. Regarding the environment scene setting of vehicle agent training and testing, the map Town01 in CARLA is selected for training and the map Town02 for evaluation in this paper.

3.2. Experimental Process

In the training phase, this paper uses the same environmental interaction hyperparameters for both DDPG and ODDPG developed agents. Among them, ODDPG introduces human driving data on DDPG. The specific method is to collect the outer road of the training map Town01 by manual driving after the manual control mode is enabled in CARLA, and to mix the filtered driving data after the automatic cruise mode is enabled in CARLA.

4. Analysis of Experimental Results

4.1. Comparison of Driving Data

The reward distribution pairs of human driving data and environmental exploration data in the original DDPG are shown in Table 1.

Table 1. Comparison of human driving data with DDPG raw exploration data

	500	1000	1500	2000	2500	3000
Human	2	6	4	1	3	7
DDPG	-9	1	-28	-59	-12	-15

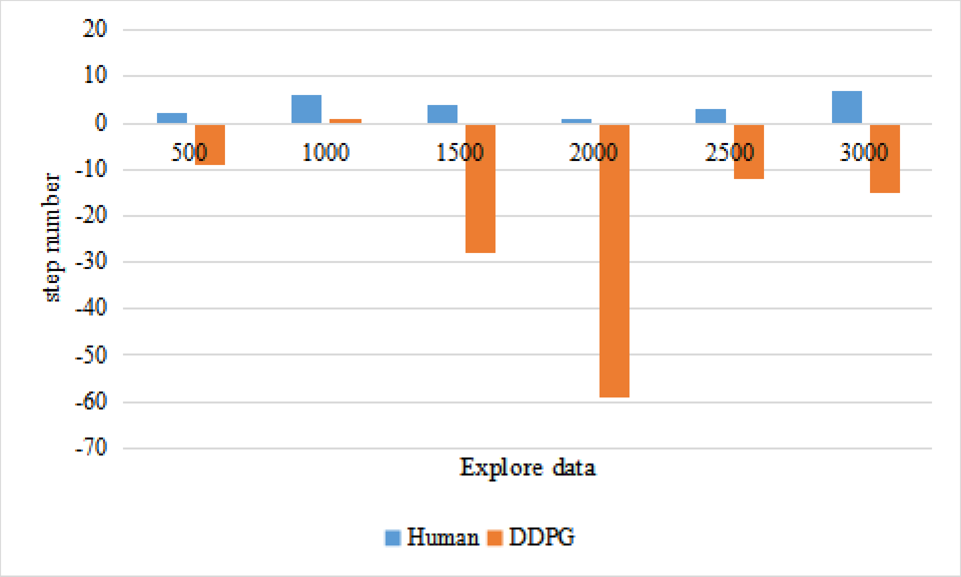

Figure 1. Comparison of reward distribution for original exploration data

As shown in Figure 1, the blue bars represent the human driving data and the orange bars represent the reward distribution of DDPG original exploration data. It can be seen that the overall reward value distribution of human driving data is higher than that of DDPG original exploration data. Secondly, the data distribution of the former is positive, while the latter is mostly negative, which indicates that the data of the former is closer to the driving behavior of the ideal driving strategy.

4.2. Training in Return

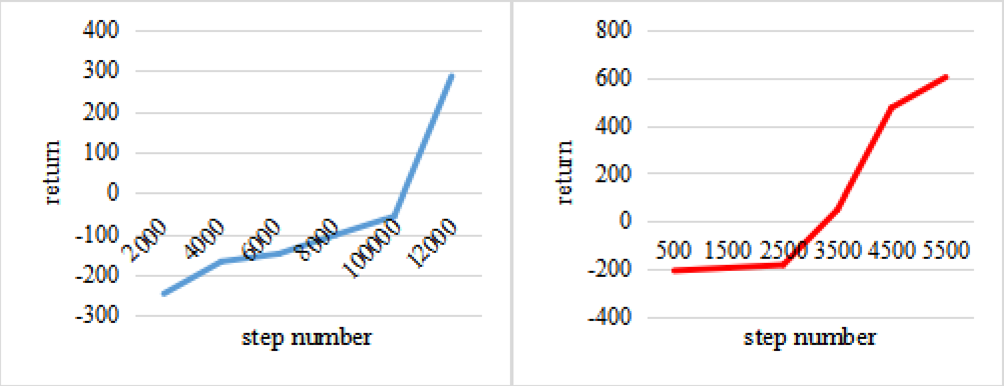

Figure 2. DDPG(left) and ODDPG(right) returns for each scene

As shown in FIG. 2, it compares the returns of DDPG and ODDPG vehicle agents in each act during training. The returns of both training acts show an upward trend, but ODDPG rises faster. Among them, DDPG training screen returns broke 0 after 120,000 steps, and ODDPG is about 3500 steps; "One step" represents an interaction between the agent and the environment. The greater the return value of one scene interaction, the more the driving strategy learned by the vehicle agent conforms to the ideal driving strategy in the current environment.

4.3. Interactive steps

Table 2. Comparison of interactive steps in each scene

	2000	4000	6000	8000	10000	12000
Human	17	23	21	27	42	129
DDPG	16	113	166	138	191	151

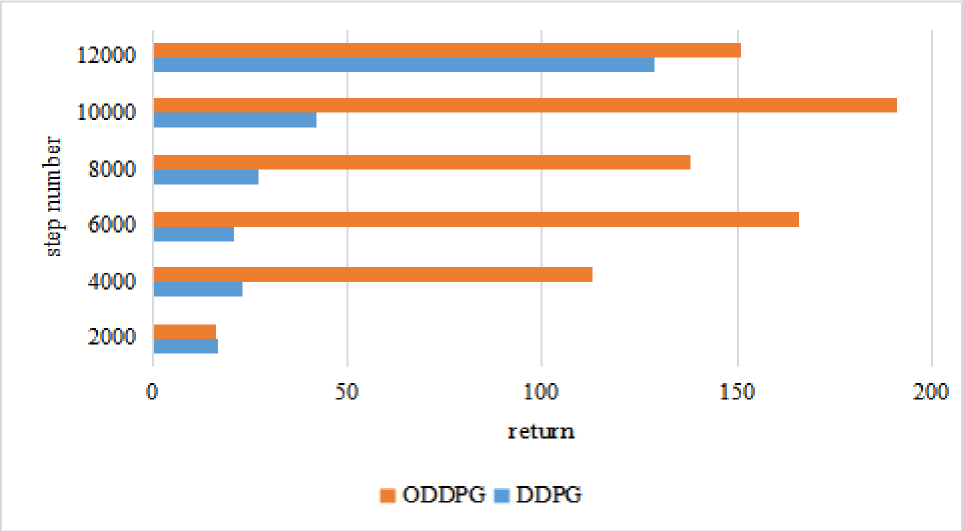

Figure 3. Comparison of the number of interactive steps in each act between DDPG and ODDPG during training

With the rapid development of hardware equipment and artificial intelligence technology, autonomous autonomous driving, as an emerging technology, is constantly making breakthroughs. Traditional automobile manufacturers and Internet technology companies have entered the field of autonomous driving and tested their autonomous autonomous driving systems. Traditional autonomous unmanned driving technology is usually controlled based on rules. However, in the face of highly uncertain traffic driving environment, the formulation of rules is often too complicated or even impossible to solve. This paper proposes an autonomous driving strategy method based on deep reinforcement learning, which iteratively learns the correct driving strategy through the reward of environmental feedback, applies it to the multi-task driving scenario of autonomous unmanned system, and controls the vehicle to complete the test task under the virtual driving environment CARLA.

Funding

This article is not supported by any foundation.

Data Availability

Data sharing is not applicable to this article as no new data were created or analysed in this study.

Conflict of Interest

The author states that this article has no conflict of interest.

References

[1] Ucar A, Demir Y, Guzelis C. Object recognition and detection with deep learning for autonomous driving applications. Simulation: Transactions of The Society for Modeling and Simulation International, 2017, 93(9):003754971770993. https://doi.org/10.1177/0037549717709932

[2] Kiran B R, Sobh I, Talpaert V, et al. Deep Reinforcement Learning for Autonomous Driving: A Survey. IEEE Transactions on Intelligent Transportation Systems, 2020, PP(99):1-18.

[3] Fujiyoshi H, Hirakawa T, Yamashita T. Deep learning-based image recognition for autonomous driving. IATSS Research, 2019, 43(4):244-252. https://doi.org/10.1016/j.iatssr.2019.11.008

[4] Makantasis K, Kontorinaki M, Nikolos I. Deep reinforcement-learning-based driving policy for autonomous road vehicles. IET Intelligent Transport Systems, 2020, 14(1):13-24. https://doi.org/10.1049/iet-its.2019.0249

[5] Hoel C J, Driggs-Campbell K, Wolff K , et al. Combining Planning and Deep Reinforcement Learning in Tactical Decision Making for Autonomous Driving. IEEE Transactions on Intelligent Vehicles, 2019, PP(99):1-1.

[6] Yi H, Park E, Kim S. Multi-agent Deep Reinforcement Learning for Autonomous Driving. KIISE Transactions on Computing Practices, 2018, 24(12):670-674. https://doi.org/10.5626/KTCP.2018.24.12.670

[7] Kim S M, Kim T H, Dong H K. Autonomous Driving Through Non-uniform Steering Angles Nodes Determination by Deep Learning. Journal of Institute of Control, 2019, 25(8):677-683. https://doi.org/10.5302/J.ICROS.2019.19.0101

[8] Sallab A , Abdou M , Perot E , et al. Deep Reinforcement Learning framework for Autonomous

Driving. Electronic Imaging, 2017, 2017(19):70-76. https://doi.org/10.2352/ISSN.2470-1173.2017.19.AVM-023

[9] Schneider L, Hafner M, Franke U. The Stixel world – A comprehensive representation of traffic scenes for autonomous driving. At - Automatisierungstechnik, 2018, 66(9):745-751. https://doi.org/10.1515/auto-2018-0029

[10] Alberti E, Tavera A, Masone C, et al. IDDA: a large-scale multi-domain dataset for autonomous driving. IEEE Robotics and Automation Letters, 2020, PP(99):1-1.

[11] Greer R, Deo N, Trivedi M. Trajectory Prediction in Autonomous Driving with a Lane Heading Auxiliary Loss. IEEE Robotics and Automation Letters, 2020, PP(99):1-1.

[12] Seo E, Lee S, Shin G, et al. Hybrid Tracker Based Optimal Path Tracking System of Autonomous Driving for Complex Road Environments. IEEE Access, 2020, PP(99):1-1. https://doi.org/10.1109/ACCESS.2020.3078849

[13] Vitas D, Tomic M, Burul M. Traffic Light Detection in Autonomous Driving Systems. IEEE Consumer Electronics Magazine, 2020, 9(4):90-96. https://doi.org/10.1109/MCE.2020.2969156

[14] Devi T K, Srivatsava A, Mudgal K K, et al. Behaviour Cloning for Autonomous Driving. Webology, 2020, 17(2):694-705. https://doi.org/10.14704/WEB/V17I2/WEB17061

[15] Weon I S, Lee S G, Ryu J K. Object recognition based interpolation with 3D LIDAR and vision for autonomous driving of an intelligent vehicle. IEEE Access, 2020, PP(99):1-1. https://doi.org/10.1109/ACCESS.2020.2982681

[16] Muhammad K, Ullah A, Lloret J, et al. Deep Learning for Safe Autonomous Driving: Current Challenges and Future Directions. IEEE Transactions on Intelligent Transportation Systems, 2020, PP(99):1-21.

[17] Mehra A, Mandal M, Narang P, et al. ReViewNet: A Fast and Resource Optimized Network for Enabling Safe Autonomous Driving in Hazy Weather Conditions. IEEE Transactions on Intelligent Transportation Systems, 2020, PP(99):1-11.

[18] Mohseni F, Voronov S, Frisk E. Deep Learning Model Predictive Control for Autonomous Driving in Unknown Environments - ScienceDirect. IFAC-PapersOnLine, 2018, 51(22):447-452. https://doi.org/10.1016/j.ifacol.2018.11.593