2024 International Conference on Advances in Data Engineering and Intelligent Computing Systems (ADICS)

Exploring and Analysing Deep Reinforcement Learning Based Algorithms for Object Detection and Autonomous Navigation

Joseph Rish Simenthy
Department of Electronics and Communication Engineering,
Hindustan Institute of Technology and Science, Chennai, Tamilnadu – 603103
rishsimenthy@gmail.com,

J.M Mathana
Professor, Department of Biomedical Engineering
Jayasakthi Engineering College, Tamilnadu
jm.mathana@gmail.com

2024 International Conference on Advances in Data Engineering and Intelligent Computing Systems (ADICS) | 979-8-3503-6482-8/24/$31.00 ©2024 IEEE | DOI: 10.1109/ADICS58448.2024.10533558

***Abstract*— Object recognition algorithms need to be very accurate and efficient in order for autonomous navigation systems to see and securely interact with their surroundings. Although conventional computer vision techniques have made significant advancements in this area, they sometimes have trouble with complicated situations, occlusions, and dynamic settings. This research study introduces and investigates deep reinforcement learning (DRL) methodologies to enhance object recognition in the context of autonomous navigation. A potent paradigm for addressing challenging decision-making issues is deep reinforcement learning. The purpose of this study is to show how agents may learn to identify and categorize objects in a variety of demanding situations by defining object detection as a reinforcement learning problem.**

***Keywords*— Object Detection, Machine Learning, Deep Reinforcement Learning, Autonomous Navigation**

I. INTRODUCTION

Autonomous navigation has emerged as a transformative technology, revolutionizing transportation and automation. At the heart of autonomous navigation lies the ability to perceive and understand the surrounding environment, enabling self-driving vehicles and robots to make informed decisions and navigate safely [1]. In this procedure, object detection is essential since it finds and recognises items in the surrounding environment. Conventional object detection approaches frequently had low performance and scalability because they relied on manually created features and shallow learning strategies.

For a variety of applications, such as self-driving cars, drones, and robots, object detection is essential to autonomous navigation. [2]. These methods enable the vehicle or robot to perceive and understand its surroundings, identify obstacles, and make informed decisions about how to navigate safely. Combination of more than one method have proved to be efficient in the object detection of Autonomous vehicles.

Deep learning (DL) has revolutionized the field of object detection, introducing algorithms that can learn complex patterns from large datasets, achieving remarkable accuracy and efficiency [3]. DL -based object detection algorithms have become indispensable for autonomous navigation, enabling self-driving vehicles to detect and classify objects like pedestrians, motor vehicles, traffic signs or signals, and lane markings [4]. In addition to vision-based methods, lidar sensors are often used for 3D object detection. Algorithms like PointNet, PointRCNN, and voxel-based approaches are used to process and detect objects in lidar point cloud data. Radar sensors are suitable for object detection in adverse weather conditions and low-light environments. They can be combined with other sensor data for robust object detection [5]. Many autonomous vehicles and robots use different sensor combinations, which includes cameras, lidars, radars, and ultrasonic sensors. Sensor fusion techniques, such as Kalman filters or particle filters, are employed to integrate information from multiple sensors and improve object detection accuracy. To achieve real-time performance, hardware accelerators like GPUs, TPUs, and custom ASICs are often used to speed up object detection computations .

A number of variables, including the kind of sensors at hand, the available processing power, the need for real-time processing, and the particular use case of autonomous navigation, influence the detection method selection. For strong and dependable object identification and tracking, many autonomous vehicles and robots integrate numerous techniques. To keep detected objects identities consistent over time, object tracking is necessary. Deep learning-based trackers, such as Deep SORT (Deep Simple Online and Realtime Tracking) and GOTURN (Generic Object Tracking Using Regression Networks), are used for this purpose [6],[7]. While not strictly object detection, semantic segmentation is used to classify each pixel in an image, which can be useful for understanding road scenes and obstacles [8]. Combining dl algorithms with techniques that have been proven to be effective in object detection is the aim of this effort

II. RELEVANCE OF USING DL ALGORITHMS FOR OBJECT DETECTION IN AUTONOMOUS NAVIGATION

DL-based object detection algorithms offer several advantages over traditional methods, making them the preferred choice for autonomous navigation applications. This effort aims to integrate deep learning algorithms with object detection techniques that have proven to be accurate [9]. This results in extremely high accuracy in object detection tasks, even in challenging conditions such as low lighting, poor weather, and occlusions. DL algorithms can also be tuned for real-time performance, which makes them appropriate for use in autonomous navigation systems that need to react quickly. [10]. This is particularly important for applications such as obstacle avoidance and collision prevention.

DL algorithms can also be scaled to different hardware platforms, from high-performance computing systems to mobile devices. This makes them suitable for a wide range of autonomous navigation applications, from self-driving cars to drones and robots. Furthermore, DL algorithms can be

adapted to different object detection tasks, such as pedestrian detection, vehicle detection, and lane detection [11]. This makes them versatile tools for autonomous navigation systems that need to operate in a variety of environments. As a result of these advantages, DL-based object detection algorithms have become the preferred choice for autonomous navigation applications. These algorithms are being used to detect and classify objects in real time, allowing autonomous vehicles and robots to navigate safely and effectively [12].

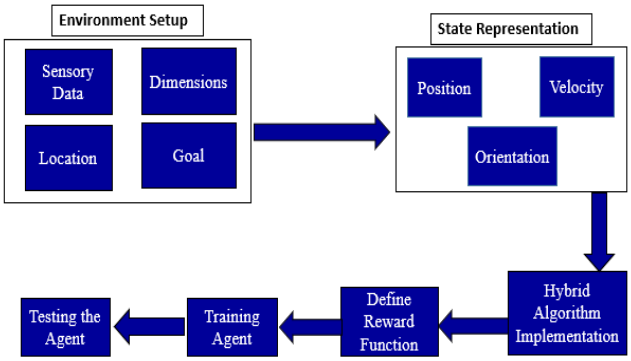

Figure 1: Block diagram of DRL based object detection system

The hybrid Deep Reinforcement Learning (DRL) methods for object detection and navigation in autonomous navigation are implemented as shown in the block diagram. Data from sensors fusion and other sensory devices, GPS, location-analysing devices, dimensions, the navigation device's purpose, and so on, are provided in the environment setup block. The state representation receives these data as input. After defining a reward function and applying the hybrid DRL algorithms to the input data from the state representation, the agent is trained and then put to the test for correctness. It is discovered that in the autonomous navigation scenarios, this configuration has attained the highest accuracy.

III. Matrices Used for the Object Detection and Navigation in Autonomous Vehicles

In evaluating DRL algorithms for object detection in autonomous navigation, crucial metrics ensure a comprehensive assessment of model performance. In order to quantify the accuracy of object localization, Intersection over Union (IoU) evaluates the overlap between predicted and ground truth bounding boxes. Precision and Recall gauge the algorithm's ability to make correct positive predictions and capture all relevant instances, respectively. Average Precision (AP) summarizes precision-recall curves, offering a unified performance metric. F1 Score harmonizes precision and recall, especially valuable for imbalanced datasets. Mean Average Precision (mAP) extends AP across multiple object classes, providing a holistic evaluation. False Positive Rate (FPR) is pivotal in scenarios where false positives carry substantial consequences. Together, these measures enable researchers to identify the advantages and disadvantages of various DRL algorithms, enabling them to make well-informed judgements that improve the accuracy of object identification for autonomous navigation systems.

IV. Survey on DRL Algorithms Used for Autonomous Navigation

To tackle challenging sequential decision-making issues, deep reinforcement learning (DRL) is a potent machine learning topic that incorporates the best aspects of both deep learning and reinforcement learning (RL). Demonstrating remarkable performance and adaptability in intricate contexts, DRL algorithms have been utilised in multiple fields such as item identification, autonomous vehicle navigation, and robot control. Several DRL algorithms have been developed to address different challenges in learning. Three classes—value-based, policy-based, and model-based—can be used to broadly group these algorithms. Value-based approaches, like Q-learning, concentrate on projecting future returns for every state under a specified policy, whereas policy-based approaches, like DDPG and SARSA, learn the optimal policy directly. The Model based methods, like TRPO and PPO, use an environment model to plan and make decisions. DRL algorithms have shown great success in various applications, including mastering games like Go, Poker, and Quake III, as well as controlling robots and autonomous vehicles. These algorithms have the potential to revolutionize industries, such as transportation, manufacturing, and healthcare, by enabling the development as well as deployment of more efficient, safe, and adaptable systems.

A. Deep Q-Learning (DQN)

The DQN algorithm, or Deep Q-learning algorithm, is a RL algorithm that has been successfully applied in various fields. In order to help agents learn the best course of action in complicated environments, deep neural networks and Q-learning are combined. The algorithm uses a combination of experience replay and target networks to improve stability and convergence. The algorithm has shown improved performance in terms of convergence speed, decision-making time, energy consumption, utilization rate, exploration time, and learning capacities. According to this research, xiao et al. [13] developed an enhanced DQN method for a single AGV path planning problem that incorporates the D* technique and is used for the port AGV task. The paper describes the DQN algorithm as a DRL algorithm used for path planning in a port AGV control system.

A unique control algorithm based on a DQN network was given by Juan et al. [14]. It combined several open-source systems and outperformed other methods now in use. By adjusting parameters, such as employing the RMSprop optimizer and MSE loss function, the approach improved the neural network's functional structure and improved goal-finding and collision-avoidance performance. In an effort to reduce overall tardiness, Zhang et al. [15] presented a Deep-Q-Network (DQN) online scheduling technique for the hybrid flow shop scheduling problem with dynamic order arrival. The methods used seven state attributes to characterise the production state of the scheduling point and coupled traditional scheduling rules with DQN to choose jobs and assign them to feasible machines. The experiment's results demonstrated that DQN is superior to Q-learning and single scheduling rules in terms of benefits and versatility.

B. Double-Deep Q Learning (DDQL)

Double Deep Q-Learning (DDQL) is a technique that solves problems like object detection and autonomous vehicle navigation by combining the advantages of DL and Q-learning. To lessen the overestimation bias of Q-learning,

IEEE 979-8-3503-6482-8/24/$31.00 ©2024 IEEE

DDQL makes use of dual Q-value functions [16]. To lessen bias, the two target networks are combined using double Q-learning, both with and without delay. Numerous applications, including intrusion detection systems and autonomous car navigation, have made use of DDQL. Bin Issa et al. proposed improvement solutions for the Q-learning algorithm [17]. These included recombination Q-learning and merging Q-learning and deep learning. Future directions for study were indicated, and the state of the Q-learning algorithm's existing implementation in dynamic obstacle avoidance was emphasised. They discussed the limitations of traditional Q-learning algorithms in dynamic obstacle avoidance and the challenges they face in handling large state and behaviour spaces, designing suitable reward functions, and balancing exploration and exploitation.

C. Deep Deterministic Policy Gradient (DDPG)

An RL technique that learns deterministic policies for continuous action spaces is called Deep Deterministic Policy Gradient (DDPG). It is a hybrid policy class that incorporates ideas from value- and policy-based approaches. By eliminating action randomness, DDPG produces simpler and more predictable rules [18]. The Bellman equation and off-policy data are used to learn the Q-function, and the policy is then learnt via the Q-function. Applications for DDPG include mobile robots and autonomous vehicles.

Chang et.al [19] reviewed recent research progress and achievements in pedestrian and object detection with dynamic obstacle avoidance, with a focus on analysing and improving the Q-learning algorithm. Paper focused on enhancing autonomous driving control using DRL. Since traditional control strategies are not very flexible in response to changing traffic conditions, autonomous driving control is achieved by combining CNN with DDPG and RDPG algorithms. Real-time road images from AirSim are used as training data, and a reward-generation method is developed to increase control performance and convergence speed. In addition to mentioning the usage of the DDPG algorithm for the continuous action space in the car-following problem, Dian et al. [20] presented the unique method of training and evaluating control policies using the CARLA simulator. The study noted earlier research that employed DRL algorithms for lane-keeping, lane-changing, overtaking, and multi-lane cruising[21], underscoring the promise of DRL in transportation settings[21].

D. Twin Delayed DDPG (TD3)

A model-free, online, off-policy reinforcement learning system using visual recognition algorithms is called Twin Delayed Deep Deterministic Policy Gradient (TD3). YOLOv5 was used for object recognition and to retrieve object position data in order to increase accuracy and speed of detection [22]. For each object, the proximal policy optimisation approach was used to establish the best grabbing strategy. Sitong et al.'s test results [23] showed how the recommended approach had a higher detection accuracy and a quicker rate of object recognition. With batch 16 and epoch 50, the YOLOv5 algorithm's minimal loss was 0.014, and its highest object identification confidence was 96%. The YOLOv5 algorithm reduced the training loss by 88.07% in comparison to the YOLOv4 technique [24].

Based on YOLOv5, mobile manipulators were identified 96% of the time, whereas YOLOv4 provided 94% of the recognition precision. The accuracy of YOLOv5 increased by 2.12% when compared to YOLOv4. When compared to previous ablation experiments, their method yields the highest precision, with a mean average precision (mAP)@0.5 value of 92.3%.

The study was primarily concerned with the suggested method for dynamic path planning for mobile robots. The topic of safe path planning for robots in situations where the layout was known but the precise obstacle distribution was unknown was the focus of the authors Wang et al.'s study [25]. They presented a unique technique that minimised the number of training epochs needed for stabilisation and optimised path length by combining the proximal policy optimisation (PPO) algorithm with the A* algorithm. In addition to presenting simulation tests to assess the effectiveness of the suggested algorithm, the study gave background information on conventional path planning methods and reinforcement learning algorithms.

E. Soft Actor-Critic (SAC)

A novel off-policy actor-critic DRL is called Soft Actor Critic (SAC), and it is based on the maximum entropy reinforcement learning framework. In order to allow the policy to explore broadly and capture numerous kinds of near-optimal behaviour, it tries to maximise both anticipated reward and entropy. SAC improves learning speed over state-of-the-art techniques by fusing a stable stochastic actor-critic formulation with off-policy updates. It is demonstrated to attain state-of-the-art performance on many continuous control benchmark tests and is intended to be applied to complicated, real-world domains. More effective path planning resulted from Zhao et al.'s [26] disclosure of an improved Soft Actor-Critic (SAC) method that supported a continuous action space and operated as an offline technique.

Maximum entropy was added to solve the local optimality issue and strengthen the system's resistance to interference. In order to speed up sample utilisation, the hindsight experience replay (HER) technique was also included.

The goal of the HER algorithm was to improve algorithm performance by efficient reuse of previous knowledge. It was possible to show that the improved algorithm performed better than the previous approach by using simulation studies. The potential of advanced machine learning algorithms, such as SAC, in enhancing the control and guidance capabilities of AUVs in challenging marine environments and provided insights into the practical implementation of such algorithms in real-world scenarios. The primary focus is on the sensitivity analysis of the state vector, which determines which sensors are required for the SAC algorithm to control the AUV. Additionally, the use of ROS middleware and real-time simulations facilitates the potential transferability of the SAC-based controller to actual AUVs.

V. COMPARISON OF DIFFERENT DRL ALGORITHMS

The table compares the six pioneer DRL algorithms used in the Autonomous Navigation systems for efficient object detection and accurate navigation. The study of these six Deep Reinforcement Learning algorithms gave insights to various performance parameters of the algorithms.

Different matrices are used to compare the specific characteristics and performance parameters which will give an insight to the application levels of these algorithms.

IEEE 979-8-3503-6482-8/24/$31.00 ©2024 IEEE

The Table 1 gives the hardware test beds used for the evaluation of those algorithms while Table 2 compares the accuracy, throughput and the efficiency of DRL algorithms.

TABLE I : COMPARISON OF DIFFERENT DRL ALGORITHMS AND THE HARDWARES USED FOR THE IMPLEMENTATION

Work	Algorithm	Hardware Used
Mnih et al., 2015, Hessel et al., 2017 Wang et al., 2016	DQN	GPU (GTX 580) [27] Google Brain's Tensor Processing Unit (TPU) [28] Multi-core CPU, NVIDIA - GeForce GTX 780 GPU [29]
Hasselt et al., 2016 Horgan et al., 2018	DDQN	GPU (NVIDIA Titan X) [30] Intel- Core i7-8700K CPU, NVIDIA- GeForce- RTX 2080 Ti GPU [31]
Hunt et al., 2015 Guez et al.,2016 Baram et al., 2017	DDPG	CPU (Intel Core i7) [32] (GPU: NVIDIA-Titan X) [33] NVIDIA-Titan X GPU, Intel Xeon E5-2630 v4 CPU [34]
Fujimoto et al., 2018 Marcin et al., 2017 Duan et al 2016	TD3	GPU (Nvidia- GTX 1080 Ti) [35] GPUs (NVIDIA-Tesla P100) GPUs (NVIDIA-Tesla K40) [36]
Schulman et al., 2017 Philipp et al., 2015	PPO	GPU (Nvidia-GTX 1080 Ti) [37] CPUs (Intel Core i7-5930K), GPUs (NVIDIA GeForce-GTX 980 Ti) [38]
Haarnoja et al., 2018	SAC	GPU (Nvidia-GTX 1080 Ti) [39]

TABLE II : COMPARISON OF DIFFERENT DRL ALGORITHMS ACCORDING TO THE ACCURACY, THROUGHPUT AND EFFICIENCY (EXTENSION OF TABLE I)

Algorithms	Accuracy	Throughput	Efficiency
DQN	Achieved performance similar to humans on Atari games	Limited by experience replay and off-policy updates	sample inefficient, overestimate Q-values
DDQN	Improved over DQN in Atari games, reduced overestimation bias	Similar to DQN	More stable learning, reduces overestimation bias
DDPG	Effective for continuous control tasks	Higher than DQN for continuous control tasks	Can be sensitive to hyper parameters, requires careful tuning
TD3	Improved over DDPG in continuous control tasks	Similar to DDPG, potentially higher due to stability	More robust to overfitting, addresses overestimation bias
PPO	Achieved outstanding results in various tasks	High, efficient use of data	Sample efficient, stable and outperforms other algorithms
SAC	Achieved futuristic results in continuous control tasks	High, efficient exploration	Sample efficient, stable learning, addresses off-policy issues effectively

SAC handles continuous action spaces, making it ideal for precise control in autonomous navigation. This enables smoother and more accurate movements, enhancing the overall navigation experience. It also balances exploration and exploitation, allowing the system to explore new paths while exploiting previous knowledge. This adaptability helps in handling diverse environments and coping with unseen obstacles effectively. Through trial and error, SAC uses reinforcement learning to determine the best navigation policies. Over time, it becomes more adept at object identification and navigation by taking in input from the surroundings.

PPO offers stable optimization by using a surrogate objective function. It prevents drastic policy updates, ensuring a smoother learning process and reducing the chances of catastrophic failures during navigation. PPO efficiently utilizes collected experience by reusing data multiple times, resulting in more effective learning from limited samples. This makes it suitable for scenarios where acquiring real-world experience is time-consuming or expensive. PPO is highly scalable, allowing it to handle complex environments and large action spaces. This flexibility makes it well-suited for object detection and navigation in diverse autonomous navigation scenarios.

EfficientDet is renowned for its exceptional balance between accuracy and efficiency. It achieves this through compound scaling, optimizing network depth, width, and resolution. This ensures high-performance object detection without excessive computational resource requirements. It excels in performing real-time object detection with high precision and recall. This is critical for autonomous navigation as it enables the system to detect and identify obstacles, pedestrians, traffic signs, and other relevant objects in its surroundings. Also, EfficientDet is designed to be versatile, capable of detecting objects of various sizes. It exhibits excellent performance for both small and large objects, providing comprehensive situational awareness for autonomous navigation systems.

VI. CONCLUSION

Quality and accuracy of object detection and Navigation can be improved by choosing a hybrid model of the algorithms SAC, PPO and EfficientDet. The combination of SAC, PPO, and EfficientDet will enhance the safety in autonomous navigation systems. Accurate object detection, smooth control, and stable optimization will contribute to better hazard perception and collision avoidance, ultimately reducing the risk of accidents. SAC's exploration-exploitation balance, PPO's scalability, and EfficientDet versatility will enable autonomous navigation systems to function effectively in diverse environments. They can handle variations in lighting, weather conditions, and different types of objects, ensuring reliable performance across various scenarios. Leveraging the SAC algorithm, PPO, and the EfficientDet model offers significant benefits for object detection and navigation in autonomous navigation systems. These techniques will enhance safety, real-time responsiveness, adaptability to different environments, and optimal resource utilization, ultimately improving the

IEEE 979-8-3503-6482-8/24/$31.00 ©2024 IEEE

effectiveness and reliability of autonomous navigation systems.

REFERENCES

[1] Schwarting, J.Alonso-Mora,"Planning and decisionmaking for autonomous vehicles," Annual Review of Control, Robotics, and Autonomous Systems, no. 0, 2018.

[2] J. Kocić, Jovičić, V. Drndarević, "Sensors and Sensor Fusion in Autonomous Vehicles," 26th Telecommunications Forum (TELFOR), Belgrade, Serbia, 2018, pp. 420-425, doi: 10.1109/TELFOR.2018.8612054.

[3] S. Shah, Dey and A. Kapoor, "Airsim:High-fidelity visual and physical simulation for autonomous vehicles," in Field and Service Robotics. Springer Journal, 2018, pp. 621–635

[4] T. Lesort, N. Diaz-Rodriguez, J.-F. Goudou, and D. Filliat, "State representation learning for control: An overview," Neural Networks, vol. 108, pp. 379 – 392, 2018.

[5] Balado J., Martínez-Sánchez J., Arias P., Novo , "Road environment semantic segmentation with deep learning from mls point cloud data," Sensors - 19 (16) (2019) 3466.

[6] Sankar K., Anima , Jhareswar, and Pabitra Mitra. "Deep learning in multi-object detection and tracking: state of the art," Applied Intelligence 51 (2021): 6400-6429.

[7] Zhao, Zhong-Qiu, Peng Zheng, Shou-tao, and Xindong. "Object detection with deep learning: A review," IEEE transactions on neural networks and learning systems 30, no. 11 (2019): 3212-3232.

[8] Gilan, Emad, M., & Alizadeh,(2019). "FPGA-based implementation of a real-time object recognition system using a convolutional neural network," IEEE Transactions on Circuits and Systems II, Express Briefs, 67(4), 755-759.

[9] Y. Tian, J. Gelernter, X. Wang, Chen, Gao, Zhang, X. Li (2018) "Lane marking detection via deep convolutional neural network Neurocomputing,"t-280, pp. 46-55

[10] A. Raffin, A. Hill, Traoré, T. Lesort, N. D. Rodríguez, and Filliat, "Decoupling feature extraction from policy learning: assessing benefits of state representation learning in goal based robotics," CoRR, vol. abs/1901.08651, 2019.

[11] Kumar A, Saini, Pandey PB, Agarwal A, Agrawal, Agarwal (2022) "Vision-based outdoor navigation of self-driving car using lane detection. International Journal of Information Technology," 14(1):215–227

[12] Zaidi, S. S., Ansari, Aslam, A., Kanwal, N., Asghar & Lee, B. (2021). "A Survey of Modern Deep Learning based Object Detection Models,". ArXiv. /abs/2104.11892

[13] Xiao Leibing, Xinchao, Yuelei et al. "An Improved DQN Algorithm for Automated Guided Vehicle Pathfinding Problem in Port Environment," 2023, [https://doi.org/10.21203/rs.3.rs-2911598/v1]

[14] Juan, Escobar-Naranjo., Gustavo,Paulina, X., Ayala., Carlos, A., Garcia., Marcelo, Garcia. (2023). "Autonomous Navigation of Robots: Optimization with DQN," Applied Sciences, 13(12):7202-7202. doi: 10.3390/app13127202

[15] X. Wang, T. Li, B. Wang and Zhang, "DQN-based online scheduling algorithm for hybrid flow shop to minimize the total tardiness," 15th International Symposium on Computational Intelligence and Design (ISCID), Hangzhou, China, 2022, pp. 66-69, doi: 10.1109/ISCID56505.2022.00022.

[16] S. Liu, G. Tian, Y. Zhang, M. Zhang and S. Liu, "Active Object Detection Based on a Novel Deep Q-Learning Network and Long-Term Learning Strategy for the Service Robot," IEEE Transactions on Industrial Electronics, vol. 69, pp. 5984-5993, 2022, doi: 10.1109/TIE.2021.3090707.

[17] Bin Issa, R..Das, M.. Rahman,Barua, Rhaman, M.K.; Ripon, K.S.N., Alam, M.G.R. "Double Deep Q-Learning and Faster R-CNN-Based Autonomous Vehicle Navigation and Obstacle Avoidance in Dynamic Environment," Sensors 2021, 21, 1468. https://doi.org/10.3390/s21041468.

[18] Silver, David & Lever, Guy & Heess, Nicolas & Degris, Thomas & Wierstra, Daan & Riedmiller, 2014 "Deterministic Policy Gradient Algorithms," 31st International Conference on Machine Learning, ICML 2014. 1.

[19] Chang, Che-Cheng, Jichiang , Jun-Han , and Yee-Ming. 2021. "Autonomous Driving Control Using the DDPG and RDPG Algorithms," Applied Sciences 11, no. 22: 10659. https://doi.org/10.3390/app112210659

[20] Li, Dian-Tao and Ostap . "Modified Ddpg Car-Following Model with a Real-World Human Driving Experience with Carla Simulator," SSRN Electronic Journal 2021

[21] Sitong Zhang, Yibing , Qianhui , "Autonomous navigation of UAV in multi-obstacle environments based on a Deep Reinforcement Learning approach," Applied Soft Computing, Volume -115, 2022, 108194, https://doi.org/10.1016/j.asoc.2021.108194.

[22] Jeng, Shyrlong & Chiang, Chienhsun. 2023. "End-to-End Autonomous Navigation Based on Deep Reinforcement Learning with a Survival Penalty Function," Sensors. 23. 8651. 10.3390/s23208651.

[23] N. Abo Mosali, S.Shamsudin, Alfandi, R. Omar and N. Al-Fadhali, "Twin Delayed Deep Deterministic Policy Gradient-Based Target Tracking for Unmanned Aerial Vehicle with Achievement Rewarding and Multistage Training,"2022 IEEE Access, vol. 10, pp. 23545-23559, doi: 10.1109/ACCESS.2022.3154388.

[24] Q. Zheng, Z. Peng, Zhu,Zhao, Zhai and W. Ma, "An Object Recognition Grasping Approach Using Proximal Policy Optimization with YOLOv5," 2023 IEEE Access, vol. 11, pp. 87330-87343, doi: 10.1109/ACCESS.2023.3305339.

[25] Jin, & Wang, "Proximal policy optimization based dynamic path planning algorithm for mobile robots," 2022, Electronics Letters, 58(1), 13-15. 2022 https://doi.org/10.1049/ell2.12342

[26] Zhao, Wang., Zhao, Q., Zheng, & Gao, H. (2023). "A Path-Planning Method Based on Improved Soft Actor-Critic Algorithm for Mobile Robots,"Biomimetics,8(6),481.https://doi.org/10.3390/biomimetics8060481

[27] Mnih, V., Kavukcuoglu, K., Silver, D., Rusu,, Veness, Bellemare, M. G., Graves, A., Riedmiller, M., Fidjeland, Ostrovski, G., Petersen, S., Beattie, Wierstra, D., Legg, & Hassabis, D. 2015. "Human-level control through deep reinforcement learning," Nature, 518(7540), 529-533. https://doi.org/10.1038/nature14236

[28] Hessel, Matteo, Modayil, Joseph, Van Hasselt, HadoSchaul, Tom & Ostrovski, Dan & Piot, Bilal,Azar, Mohammad,Silver, David,2017. "Rainbow: Combining Improvements in Deep Reinforcement Learning," Proceedings of the AAAI Conference on Artificial Intelligence. 32. 10.1609/aaai.v32i1.11796.

[29] Wang., Sun, Xie, Y, Bin, & Xiao, J. 2023. "Deep reinforcement learning-aided autonomous navigation with landmark generators," Frontiers in Neurorobotics,17. https://doi.org/10.3389/fnbot.2023.1200214

[30] Van Hasselt, Hado,Guez, Arthur, Silver, David. 2015. "Deep Reinforcement Learning with Double Q-Learning," Proceedings of the AAAI Conference on Artificial Intelligence. 30. 10.1609/aaai.v30i1.10295.

[31] Horgan, D., Quan, Budden, D., Hessel, Van Hasselt, Silver, 2018. "Distributed Prioritized Experience Replay,". ArXiv. /abs/1803.00933

[32] Lillicra., Hunt, Pritzel, A., Heess, Erez, Tassa, Y., Silver, D., & Wierstra,.2015. "Continuous control with deep reinforcement learning," ArXiv. /abs/1509.02971

[33] Hasselt, Guez., & Silver 2016. "Deep reinforcement learning with double q-learning," In Proceedings of the AAAI conference on artificial intelligence (Vol. 30, No. 1).

[34] Anschel, O., Baram & Shimkin 2017. "Averaged-dqn: Variance reduction and stabilization for deep reinforcement learning," In Proceedings of the 34th International Conference on Machine Learning-Volume 70 (pp. 176-185). JMLR. org.

[35] Fujimoto, S., Van Hoof, & Meger 2018"Addressing Function Approximation Error in Actor-Critic Methods,". ArXiv. /abs/1802.09477

[36] Duan, Y., Chen, Houthooft, R., Schulman, J., & Abbeel,2016. "Benchmarking Deep Reinforcement Learning for Continuous Control," ArXiv. /abs/1604.06778

[37] Schulman, J., Wolski, Dhariwal, Radford, A., & Klimov,2017. "Proximal Policy Optimization Algorithms," ArXiv. /abs/1707.06347

[38] Schulman, John & Moritz, Philipp & Levine, Sergey & Jordan, Michael & Abbeel 2015 "High-Dimensional Continuous Control Using Generalized Advantage Estimation,"

[39] Haarnoja, Zho., Abbeel,& Levine2018 "Soft Actor-Critic: Off-Policy Maximum Entropy Deep Reinforcement Learning with a Stochastic Actor,". ArXiv. /abs/1801.01290

IEEE 979-8-3503-6482-8/24/$31.00 ©2024 IEEE