Reinforcement Learning Approaches for Autonomous Vehicle Navigation in Dynamic Environments

By Dr. Fumihiko Matsuno

Associate Professor of Mechanical Engineering, Nagoya Institute of Technology, Japan

1. Introduction

Traditional autonomy in vehicles can be divided into perception, prediction, and policy decision-making—the vehicle first perceives the surroundings through various sensors, such as cameras, LIDAR, and radar, tracks relevant objects based on sensor information, uses the prediction model to predict the future position and behaviors of moving obstacles, and then outputs policy decisions to control the vehicle movement. In recent years, thanks to the rapid development of deep learning, autonomous vehicle navigation has evolved from the original architecture into an end-to-end manner, where learning-based methods play the dominant role in autonomous navigation [1].

Non-learning-based approaches such as visual odometry (VO) and the artificial potential field (APF) are not adaptive to environmental changes and struggle with unknown dynamic environments, while end-to-end deep learning (DL) such as convolutional neural networks (CNNs) and recurrent neural networks (RNNs) require large amounts of labeled data, which is difficult to collect for navigation in unknown environments. Lately, they are not adaptive to environmental fluctuations and are highly dependent on initial learning. Inherently, Deep Reinforcement Learning (DRL) can learn behavior-specific navigation policies from the interactions with the environment, avoid needing large-scale labeled data, and make it possible to visualize policies that are crucial to solving the task, which can be expected to explore unseen policies and make breakthroughs in autonomous navigation, especially in dynamic environments.

1.1. Background and Motivation

Planning terrain navigation strategies are efficient than traditional methods, as they better generalize to various terrains. However, the terrain navigations lack the adaptability to unseen obstacles UOs, such as pedestrians, or approaching obstacles, AO, that hinder the traversability of the terrain during the motion, and could not adapt fast-enough to dynamic variations of the terrain. While perception-based approaches provide a safety layer to the vehicle's decision-making by providing the map of the environment, an especially challenging domain is routing, where the vehicle needs to plan its motion in the presence of unknown dynamic obstacles. Both model-based and learning-based techniques are employed for handling navigation in the presence of UOs and AOs. Many studies have addressed handling unknown obstacles using model-based methods; e.g. traditional deep-learning mapping approaches can rely on maps generated from depth image-based methods. Reinforcement learning (RL) can be used to plan paths while avoiding unknown dynamic obstacles [2].

Autonomous exploration with the aid of mobile robotics is an important research topic that has gained significant research attention in recent years. Robots can perform diverse tasks in unknown or complex environments such as hazard recognition and avoidance, door opening, object detection, and segmenting, understanding and searching semantic regions. To explore and map unknown or partially-known environments, traditional exploration mechanisms use a predefined navigation policy (i.e., learning-free algorithms) or a learning algorithm (i.e., learning-based algorithms). In conventional navigation strategies, artificial potential fields, graph search algorithms, evolutionary algorithms, population-based algorithms, optimal control, and randomized algorithms are used [3].

1.2. Scope and Objectives

It must also be noted that input data allow for the consistent use of a certain infrastructure-mainly an area of fixed light and road conditions at the right time of day. It is therefore a part of the tasks to adapt to altered lighting conditions, weather and object size in the further research and improvement of the detected object. Information presented in derived from this machine learning technique for pedestrian detection should be continuously exchanged with surveillance videos for autonomous driving. The driving decision taken will then effectively improve the robustness and accuracy of AVs when changing away from fixed light and road conditions [3,8].

A current fascination of society has been the field of Ambient Assisted Living (AAL) and the development of systems that provide pertinent and "intelligent support" while observing data protection regulations. This research topic explores new machine-learning technologies and investigates the choice of suitable embedded hardware. Previous work on the detection of road users has been studied, and an in-depth overview of different fusion strategies to combine the detection of radar or optically detectable objects with the intention of pedestrians has been provided. As a result, it was noted that many current applications in the pedestrian detection field select the strategy of sensor fusion between different sensors.2 This somewhat limits the impact of dynamic uncertainty during, for example, car-path hills.

2. Fundamentals of Reinforcement Learning

The choice of actions to perform may depend on the state of the environment and the actions need to be taken sequentially in order to achieve a goal. A computer program within is gradually introduced to the environment and can master a quadratical task like playing chess, go or video car racing through repeated trials. Although the history is long and filled with triumphs and disappointments, the last decade has witnessed a large push for the use of RL in real-world domains including healthcare, education, and transport.

[4] Reinforcement Learning (RL) can be regarded as the most powerful AI paradigm for teaching a machine how to act by interaction with the environment and learning from mistakes. It is a popular paradigm for robot or autonomous car control where the engine is often asked to play through trails in the graphics engine of the simulator and the learning process is completely unsupervised. The joint learning-based decomposition classes that involve scenario identification (actions that perform best), scenario modeling, control laws, and autonomous vehicle energy can be proposed. The prototype program concerns Valet and Car scenarios with target and target chasing tasks that can be achieved in them.[5] Reinforcement Learning (RL) is derived from psychology-based animal behavior psychology and can be considered as a change. However, it has recently been subject to a resurgence of attention by AI researchers and is becoming somewhat popular. It aims to enable a computer program to learn to carry out tasks by interacting through its environment and trial these are in the absence of other human guidance or supervision, with the Computer communicating the computer. This task of learning to perform through interacting with an environment is in many ways the setup shares many similarities with human and animal learning and decision-making, so in this way it is held a primary obtained Observed techniques for machine or artificial intelligence.

2.1. Definition and Key Concepts

Moreover, due to the fact that a traffic light does not appear in enough environments, the stationary agent cannot use the traffic light in some real scenarios, causing it unable to cross the road and indulge in an infinite time wait. Generally speaking, complicated autonomous vehicle navigation and control problems are sequential decision-making problems. Reinforcement learning is a natural and promising alternative for addressing these sequential decision-making problems and autonomous decision and control tasks. Therefore, we focus on reinforcement learning techniques for AVs in the scope of this paper [6].

Autonomous vehicles (AVs) are embodied with intelligent perception and adaptive control capabilities, and thus, learning algorithms including supervised or semi-supervised learning are primary choices for endowing AVs with intelligent decision and control capabilities. For example, convolutional neural network (CNN) based remote driving [7] enables vehicles to collect clockwise and counterclockwise steering angle and their corresponding input images in a driver-training sessions, and the CNN learned from the training sessions can help perform remote driving. However, remote driving is not a desirable approach for autonomous vehicle navigation in long term. In this paper, we are interested in autonomous driving systems for stationary agents in the environment, where a stationary agent refers to a sit-and-wait control mode that stays in the current location before moving. The major difference between the sit-and wait agent and moving agent is that the stationary agent needs waiting agents and is waiting for the minimum time that the waiting agent(s) pass(es) an appointed location because a moving agent does not need waiting agents.

2.2. Types of Reinforcement Learning Algorithms

[8] Reinforcement learning (RL) is a model-free algorithm for controlling systems. The reward function guides the agent to learn a system's optimal policy, according to which it finds the actions that maximize the cumulative reward.[1] RL can be classified into model-free and model-based algorithms. Model-free algorithms do not need prior information for the system states and dynamics but learn an action-value function or policy directly. In large state spaces, approximations such as deep reinforcement learning are used to work with large-scale problems. Model-based RL learns a model of the environment and may use the model to learn approximate value functions or policies. The paper by Torabi describes multiple model-based and model-free algorithms to demonstrate how the various types of RL algorithms can be adapted to autonomous driving navigation. It investigates deep reinforcement algorithms

such as DQN, DDPG, PPO, and A3C, and then reports results on the creation of a front-end feedforward neural network and social GAN to compare. The article shows that from the six different RL algorithms, PPO can efficiently work with the autonomous driving environment. [9] All authors of these papers used CARLA: Community Automotive Research Laboratory environment (Karaman and Frazzoli, 2012), with differences between the front-end models. The primary difference for all papers is the adoption of different front-end networks for the control tower. However, the backend side remains the same by using the Proximal Policy Optimization Algorithm. The study selected ten examples of research papers in the field of autonomous driving, as a guide to applying the CNN, GAN, and RNN models in the front-end side of the CARLA simulation environment.

3. Autonomous Vehicle Navigation

The majority of contemporary machine learning based planning and control models for autonomous vehicle navigation are mainly tailored for specific scenarios where they are trained and have not shown significant success under difficult scenarios. The existing models that are based on learning lack the diversity and adaptability which are required occasionally for situations that are rare in the training scenarios. Furthermore, the computational cost of sampling-based or optimization-based methods may be extremely high. For instance, widely-used deep neural networks (DNNs) require a considerable number of samples for training. And the model trained in one environment may not generalize well to another. A common strategy for real-time robot navigation is using a powerful analytical algorithm for trajectory planning and using machine learning models for steering and acceleration control. The machine learning models, in this case, are trained offline on human data to predict the states of nearby obstacles in the predicted trajectory space for the autonomously planned trajectory. Human feedback has shown potential for improving autonomous robot navigation in complex environments [10]. In this context, we like a to contribute to reinforcement learning-based approaches in autonomous vehicle navigation, providing a detailed summary of existing studies and research gaps, and suggest possible solutions and challenges for them.

Reinforcement learning (RL) has shown promising results in various control applications and has gained considerable attention in the field of autonomous vehicles navigation during recent years. RL-based navigation systems do not rely on explicit feature generation and therefore provide high generalization ability in a changing environment. In the context of autonomous driving, deep reinforcement learning (DRL) [9] has been utilized to learn a policy that optimizes vehicle states in order to maximize cumulative rewards associated with a safe and comfortable driving behavior. Navigation systems based on RL have the advantage of possessing learning ability, improving over time, and showing low reliance on sensor accuracy. A behavior-engineered navigation system has limitations to address real-life complexities that are not easily captured in a model. RL-based systems have the potential to be tested in a virtual environment, which reduces the risk and time required for the development of systems. Inverse reinforcement learning, a prominent branch of RL-based approaches, has also been investigated for developing autonomous navigation systems for vehicles. The expert feature extraction, reward function definition, and decision making are important attributes of IRL. The approach also maneuvers well in data scarcity because it can imitate the arbitrary expert rather than the expert whose dataset is used during training [11].

3.1. Challenges in Dynamic Environments

As a result, it is difficult to handle human-like robot navigation tasks such as reason intentionally and reason intelligently. Also, at present, there is hardly research work done for convoluted tactical-level autonomous navigation on an agile mission in unknown open environments under the DRL framework. Reinforcement learning algorithms are known to be data hungry, hence lack of enough training samples often leads to suboptimal policy. Although model-based planning has been used to generate trajectories in traditional optimal control algorithms. But in a complex and dynamic environment, actual scenes cannot be justified with the same kind of scene. So, the planning trajectory obtained in the simulation environment cannot be highly beneficial in the actual scenario [12].

- Sample efficiency largely depends on the complexity of state-action space, deep Q-network (DQN) is not sufficiently scaled against high-dimensional tasks, and model-based DRL method still has a relatively high consumption of computation. - Trajectory generated by the conventional rule of the geometric center of ground-based robots is simple and can hardly bypass dynamic obstacles. This is prone to missing exploration and improve the safety performance poorly.

Conventional planning techniques, such as A* or RRT, are not able to effectively deal with dynamic and complex environments [13]. In recent years, a number of DRL approaches have been proposed to tackle the challenges in autonomous navigation on a mission with a prior map in a simple and unchanging environment [3] , while few researches to be closely involved with the complex and dynamic tactical-level decision making in an open, missionspecific and unknown environments. For instance:

3.2. State-of-the-Art Techniques

In self-driving car control methods, many deep reinforcement learning (DRL) algorithms such as DQN, DDPG, PPO, A3C, TRPO, etc. are used in the literature in order to manage the environment perception and decision-making tasks. The main purpose of DRL used in autonomous vehicle navigation is to manage environment perception, control and decision-making processes. While perception and control layers are handled separately in traditional autonomous vehicle control methods, DRL enables end-to-end learning by being able to make decisions with high-level perception without needing individual control outputs. In [9], a driving control system is designed by training a deep deterministic policy gradient architecture in a virtual car racing environment. It has been proven that the driving control system trained in a virtual environment by reinforcement learning has been successful in the real world. In [14], double deep Q-learning and deep reinforcement learning-based steering modes are proposed for decision-making processes of autonomous vehicle navigation in outdoor environments. Specifically, the Faster R-CNN system was developed for outdoor environment perception, while Carla simulator was used for environment perception and decision-making purposes.

The capabilities of reinforcement learning (RL) techniques in outdoor environment perception and control could be seen in many studies in the literature. However, to adapt the RL environment perception and control out of the virtual world like in CARLA simulator, is a challenging area since the environment dynamics are so unpredictable. Due to this dynamics, the false decisions can cause the autonomous vehicle to collide with the environment. For this reason, it is very important to adapt the RL techniques to perform optimally in dynamic outdoor environments [15]. Although many studies could be seen in the literature which proposes RL based perception and control in real environments, CARLA simulator, which has a very realistic car and pedestrian control system, could be a big advantage during training RL's environment perception and decision-making models.

4. Integration of Reinforcement Learning in Autonomous Vehicles

Deep Reinforcement Learning (DRL), a subset of RL, leverages stacked layers of neural networks to estimate what actions an agent should take on the basis of its current state, a prior history of states, and the rewards it has experienced [14]. In autonomous vehicles, DRL serves for learning to drive in complex environments and better adapt the vehicle response with respect to common control functions, such as longitudinal and lateral control, obstacle avoidance, and speed adaptation. DRL training sample data can be collected with simple, scalable game simulations and cheap data augmentation in terms of different starting configurations, rival vehicle types, and various road settings, thus reducing the environmental complexity [13].

Reinforcement Learning (RL) is a computational paradigm in which a learning system called an agent learns how to decide the best actions to perform in different situations to maximize a notion of cumulative reward [7]. Adding RL to a controller requires a training phase in which a reward function is defined to make the physical implementation of the system learn expected behavior.

4.1. Sensors and Data Collection

The primary objective of our path planner is to track the waypoints (or the small sequence thereof) from the reference path such that the intelligent vehicle follows these waypoints (with a future point of reference). [16] Here tracking waypoints on-road helps to increase the adaptability of the vehicle to move around the variable types and densities of traffic, red-lights, and pedestrians. Multi-objective lane change trajectory optimization technique has been proposed, where the vehicle is driven in the centre lane when no obstacle is present on-road, and when an obstacle is present on-road, the vehicle is driver-driven in right or left lane to avoid static or dynamic obstacles, keeping consider driving in the centre lane safely (lane-deep learning agent). Once the obstacle is not in front, it can again move to the centre lane and continue driving. Moreover, the visual condition perception allows the intelligent vehicle to operate in all-day, nighttime, heavy rain, heavy snow, fog, and daytime weather con ditions, and cluttered, uncluttered, indoor, outdoor conditions and situations.

The problem addressed in this research is to develop generalizable deep reinforcement learning algorithms that allow autonomous vehicles to navigate safely through dynamic and complex urban environments such as those present in downtown areas and in residential zones with parking vehicles, traffic lights, and pedestrians. [17] These environments present on-road frequently moving obstacles, and the autonomous vehicle needs to adaptively react to moving obstacles in real time. This type of mixed autonomy, which operates on variable types of moving obstacles and receives only visual input from a camera, is not studied to a great extent in the current literature.

4.2. Decision-Making Process

In order to handle the real-time decision-making process in the autonomous navigation of an autonomous vehicle in a dynamic environment, a decision support system was introduced for the vehicle so that it predicts and adjusts its movement. The movement prediction mechanism in this study can be variable. The use of reinforcement learning makes it possible to predict movement with the minimum information received from the environment and thus contributed to controlling the movement by predicting the sudden changes in the environment. Reinforcement learning algorithms have been used as an alternative to modeling algorithms to solve the decision-making problem of using complicated dynamic models in the proposed solution in the context of both modeling of movement and handling the uncertainties developing in the environment. [4] as well.

Deep reinforcement learning architectures designed for autonomous navigation in dynamic environments are also being developed. One study presented a deep reinforcement learning-based architecture for the decision-making process of an autonomous vehicle in dynamic environments [8]. This proposed architecture of the Decision Transformer Network Model (DTNM) consists of three main sections to identify incentives, predict future occurrences, and determine the appropriate driving behavior. The DTNM was evaluated in simulation and visual data, but the authors stated that hardware validation would be the next step.

5. Case Studies and Applications

To date, considerable work has been done to extend Deep Q-Networks (DQNs), such as Q-learning with target networks, double DQN, and dueling network architectures. Efficient use of DQNs is important, as the poor practical performance can be attributed to the Q-learning process, in which the effect of overall reward signals can be over/underestimated; errors can be propagated between nearby time-steps, and average susceptibility to oscillation and divergence of the update process can be observed. Separately, from the environment-generated feedback signal, deep neural networks can learn supplementary expert policies from observing human demonstration data not generated from the domain to be learned. The generality of these techniques is striking, as the learner only requires access to some behavioral demonstrations of the target task.

The Double Deep Q-Network (DDQN) algorithm has played a significant role in many existing reinforcement learning approaches [14]. RL-based navigation systems require rich sensory input, and autonomous vehicle navigation environments must meet certain safety constraints. The performance of the autonomous vehicle navigation system is known to be affected by how well it relates the sensory information to the corresponding state space representation [18]. However, the current methods of transforming the state space are often very rough, and there is still no explicit unified model paradigm in this regard. The early development of machine learning referred to various classic problems in a one-to-one correspondence that should be specified explicitly before learning.

5.1. Urban Traffic Scenarios

Algorithms for localization of autonomous vehicles for example; ICP or LOAM developed as a stand-alone module. Instead, both algorithms proposed here manages to contribute in both localization and object tracking modules. Urban traffic scenarios are very challenging with high density and different velocities. On the other hand, many unstructured objects are observed from the LiDAR's sensor point of view [15]. There is also a challenge how to join road network. As a natural result, high-quality and rich datasets are difficult to gain. Please refer to the Section "Introduction" for further details. All these make algorithms designed and tested in other datasets overfit and hard to be transferred to other datasets. All of these challenges pushes the community to work on same datasets which is named nuScenes dataset. On the contrary, the parameters might be different. Combining the enhancements on the analytical algorithms for instance, by clustering of li-dar points during the process of registration, we are able to plan the next navigation decision in a changable setting, which is mostly strucured. Besides, present analytical smartphone scope architecture workings only under assumption of a static environment and it crashes in the lack of such. With this in mind, we need to get rid of the correctness uncertainty concerning the presented input data, so instead of a selection from the collected surface elevation data, we have to increase robustness by representing the object tracking and prediction in the navigation decision in a unified way. On the other hand, the above algorithms can learn from different trajectories collected from eg. other vehicles, but they cannot able to express new collision-free trajectory if their point clouds are not rotated to some other view perspective. Steering new considering this compatibility constraint is a future research interest.

Recent advances in the domain of autonomous navigation present Reinforcement Learning (RL) as a governing paradigm. Urban traffic scenarios are highly dynamic and centralized traffic control systems have difficulties in managing them [19]. When a traffic control system is absent, neural networks and RL have been employed in various traffic flow optimization settings. Reinforcement learning, considered as a universal method, has been successfully applied to un-signalized urban traffic regulation. In a microsimulation environment, an DQN (Deep Q Network) agent, trained using the retimed traffic light phase durations considering the dynamic aspects of the environment, successfully reduced the average passenger waiting time of the vehicles. Therefore, the effectiveness of DQN was shown in handling dynamic aspects of traffic flow in a rule-based urban intersection. Real-world scenarios have not been investigated by existing studies with (well performing) highly non-linear DNNs that differ from rule-based simulations with respect of their complexity and other aspects. Real-world intersections and traffic systems consist lots of structured and unstructured data, which, together with the lack of true, static imitate-able models (e.g. neighbors behaviors) and inner limitations of our learning method, can lead into sub-optimal behavior of the trained agent in the real world. A real-world-friendly approach was therefore needed to be created using RL for un-signalized control of our target, the urban intersection. An alternative urban closed loop control strategy combining the ability to capture complex and dynamic traffic flow regulation dynamics with an interpretation of the search for the global optimization in real time has a great potential to be developed.

5.2. Highway Driving

Overall, the encouraging empirical results in the paper thus indicate that the PASP-DQN approach may better able to use off-policy data and learned models to maximize the non-global rewards, and represents a promising avenue for future research. Furthermore, the emergence of PASP-DQN also suggests alternative explanation for the successful results of using hint rewards together with off-policy reinforcement learning.

This chapter investigates a variant of the reinforcement learning (RL) formulation, called action suffix prediction, that aims to learn near optimal policies without needing to have access to a global or consistent reward signal [10]. Leveraging off-policy learning and learned rewards, Prioritized Action Suffix Prediction DQN (PASP-DQN) is able to achieve the optimal policy on term decaying tasks of classical control, Atari and EODM domain. In particular, the default PASP-DQN network architecture consistently outperforms the Duelling DQN [20] on all four investigations [15].

6. Evaluation Metrics and Performance Analysis

The most crucial properties required from intelligent agents in any learning environments where a real-world robotic application can find its use, such as autonomous driving, are safety, adaptation, efficiency, and ability to learn from raw sensory data. Safety is always the primary objective, and failure to follow the safety measure in autonomous vehicles can be fatal. Thus, a massive effort has been made in recent years to build an intelligent autonomous driving system, efficient enough to understand the nominal surroundings and their least changing configurations. The system must be capable of adapting to new changes in the surroundings, such as new construction, new kinds of guards, diversions, and parking lots. Moreover, it should be capable of learning trajectory data from the entropy of its immediate past, as well as being able to optimize it with different conditions. In the MaRLn approach, the agent-based navigation environment is capable of investigating the given prompt across different metric sensors, comparing the measured values, and using a pre-trained deep reinforcement learning module as a navigation policy to pick an optimal action that is adapted to the environment [7].

Learning-based approaches have recently gained significant interest in robotics research and applications [ref: 38e70e63-9b04-4abe-a279-739dff71ed06, 4e619ff2-4eeb-4258-8239-2267b62478b7]. Deep learning-based approaches, such as deep reinforcement learning, can learn features directly from raw sensory data to enable navigation in highly complex environments. Deep reinforcement learning (DRL) algorithms adapt to changes due to learning from their own experiences and can be employed in both simulation and real-world scenarios. Hence, they are effective in providing an agile and adaptable navigation solution in the dynamic environment [21].

7. Future Directions and Emerging Trends

In addition to the summary of case studies, emerging trends, and open challenges have been identified including closed-loop navigation, uncertainty-aware navigation, hierarchical learning, end-to-end agents, few-shot learning, multi-agent systems, multi-modal learning, imitation learning, one-shot imitation from humans, better exploration methods, transferring good driving habits, over-the-air updates, robustness of learned policies, long-term degradations in traffic dense environments, reusable and modular component-based learning, training and deployment in the presence of sensor noise and dropped packet loss, consider multi-scale environment and routing updates, and also addressing trade-offs between driver comfort and behavior versus efficiency and safety [3]. An overview of the state-of-the-art Reinforcement Learning-based tools and methodologies for developing autonomous navigation for wide variety of scenarios and scenarios that have emerged in the autonomous matter research community is also presented.

Developing robust and dependable autonomous vehicle architectures for safe, reliable operation in dynamically-evolving real-world environments remains an open challenge for the research community to address. Although recent advancements in computational infrastructure and machine learning algorithms have driven significant improvements in robustness and generalization capabilities in simulated training environments, further work is critical to combine these advancements with perception, real-world considerations, and in-action decision-making to translate promising results from the simulator to the real world [22]. In this technical report, we have provided an overview of several reinforcement learning paradigms, architectures, methodologies, and scenarios for autonomous navigation from the study of different existing methodologies with emphasis on recent trends, and open challenges [6].

8. Conclusion

[14]Autonomous vehicles require an efficient autonomous decision-making algorithm to operate in dynamic environments. In this paper, we discuss and evaluate various reinforcement learning (RL) algorithms for this purpose, on a toy data set with traffic and pedestrians to present a decision-making problem. The RL algorithms were evaluated on single-agent and multi-agent strategies using traffic light settings in different configurations representing various intersection types. The inclusion of pedestrians and different orientations was set to challenge the model with limited observations in learning to avoid

them in the intersections. The evaluation feedback was based on bad behaviors when driving, such as driving through red lights, pedestrian collisions, and exceeding speed limits. These parameters were used to define the performance of the algorithms. The results indicated that faster R-CNN in conjunction with Double Deep Q-network (DDQN) in the pathfinder environment yields both robust and efficient performance, getting the best average reward and time to complete the navigation.[6]This system can learn from raw pixel data and develop a map of the tyre-force environment without explicitly specifying the dynamics. We also explore queries for which the system needs to reconstruct the static environment and those for which it can exploit online exploration to generalize to new scenarios. A real-world vehicle application is presented for reconfiguration manoeuvre planning on the REINVENT test vehicle, where the velocity vector of the vehicle is optimized for a LuGre tyre model. We demonstrate experiments on changes of the peak coefficient of friction, which shows that a dynamics-aware, data-efficient policy is learned. Motion planning based on offline dynamics data is able to achieve a top speed of 2.95 m/s while the system achieves 3.05 m/s with an adaptive online training strategy, indicating that dynamics-adaptive feedback policies are advantageous when the ROI changes during train time. Software for submission is released on [Link], and supplementary videos are available at [Link]

References:

1. [1] Óscar Gil, A. Garrell, and A. Sanfeliu, "Social Robot Navigation Tasks: Combining Machine Learning Techniques and Social Force Model," 2021. ncbi.nlm.nih.gov
2. [2] L. Kästner, X. Zhao, Z. Shen, and J. Lambrecht, "Obstacle-aware Waypoint Generation for Long-range Guidance of Deep-Reinforcement-Learning-based Navigation Approaches," 2021. [PDF]
3. [3] Y. Tang, C. Zhao, J. Wang, C. Zhang et al., "Perception and Navigation in Autonomous Systems in the Era of Learning: A Survey," 2020. [PDF]
4. [4] B. Udugama, "Review of Deep Reinforcement Learning for Autonomous Driving," 2023.
5. Tatineni, Sumanth. "Beyond Accuracy: Understanding Model Performance on SQuAD 2.0 Challenges." *International Journal of Advanced Research in Engineering and Technology (IJARET)* 10.1 (2019): 566-581.
6. Vemoori, Vamsi. "Comparative Assessment of Technological Advancements in Autonomous Vehicles, Electric Vehicles, and Hybrid Vehicles vis-à-vis Manual

Vehicles: A Multi-Criteria Analysis Considering Environmental Sustainability, Economic Feasibility, and Regulatory Frameworks." *Journal of Artificial Intelligence Research* 1.1 (2021): 66-98.

7. Shaik, Mahammad, Srinivasan Venkataramanan, and Ashok Kumar Reddy Sadhu. "Fortifying the Expanding Internet of Things Landscape: A Zero Trust Network Architecture Approach for Enhanced Security and Mitigating Resource Constraints." *Journal of Science & Technology* 1.1 (2020): 170-192.
8. Vemori, Vamsi. "Human-in-the-Loop Moral Decision-Making Frameworks for Situationally Aware Multi-Modal Autonomous Vehicle Networks: An Accessibility-Focused Approach." *Journal of Computational Intelligence and Robotics* 2.1 (2022): 54-87.
9. [9] J. Hossain, "Autonomous Driving with Deep Reinforcement Learning in CARLA Simulation," 2023. [PDF]
10. [10] R. Trauth, A. Hobmeier, and J. Betz, "A Reinforcement Learning-Boosted Motion Planning Framework: Comprehensive Generalization Performance in Autonomous Driving," 2024. [PDF]
11. [11] A. Kendall, J. Hawke, D. Janz, P. Mazur et al., "Learning to Drive in a Day," 2018. [PDF]
12. [12] T. Ganegedara, L. Ott, and F. Ramos, "Learning to Navigate by Growing Deep Networks," 2017. [PDF]
13. [13] W. Zhang, K. Zhao, P. Li, X. Zhu et al., "A Closed-Loop Perception, Decision-Making and Reasoning Mechanism for Human-Like Navigation," 2022. [PDF]
14. [14] R. Bin Issa, M. Das, M. Saferi Rahman, M. Barua et al., "Double Deep Q-Learning and Faster R-CNN-Based Autonomous Vehicle Navigation and Obstacle Avoidance in Dynamic Environment," 2021. ncbi.nlm.nih.gov
15. [15] K. Cheng, X. Long, K. Yang, Y. Yao et al., "GaussianPro: 3D Gaussian Splatting with Progressive Propagation," 2024. [PDF]
16. [16] D. Paz, N. E. Ranganatha, S. K. Srinivas, Y. Yao et al., "Occlusion-Aware 2D and 3D Centerline Detection for Urban Driving via Automatic Label Generation," 2023. [PDF]
17. [17] P. Cai, S. Wang, Y. Sun, and M. Liu, "Probabilistic End-to-End Vehicle Navigation in Complex Dynamic Environments with Multimodal Sensor Fusion," 2020. [PDF]

18. [18] Z. Li, S. Yuan, X. Yin, X. Li et al., "Research into Autonomous Vehicles Following and Obstacle Avoidance Based on Deep Reinforcement Learning Method under Map Constraints," 2023. ncbi.nlm.nih.gov
19. [19] Y. Shi, Y. Liu, Y. Qi, and Q. Han, "A Control Method with Reinforcement Learning for Urban Un-Signalized Intersection in Hybrid Traffic Environment," 2022. ncbi.nlm.nih.gov
20. [20] T. Liu, X. Mu, X. Tang, B. Huang et al., "Dueling Deep Q Network for Highway Decision Making in Autonomous Vehicles: A Case Study," 2020. [PDF]
21. [21] F. Carton, D. Filliat, J. Rabarisoa, and Q. Cuong Pham, "Evaluating Robustness over High Level Driving Instruction for Autonomous Driving," 2021. [PDF]
22. [22] Y. Chen, C. Ji, Y. Cai, T. Yan et al., "Deep Reinforcement Learning in Autonomous Car Path Planning and Control: A Survey," 2024. [PDF]