



International Journal of Artificial Intelligence and Machine Learning

Publisher's Home Page: <https://www.svedbergopen.com/>



Research Paper

Open Access

Reinforcement Learning-Driven Autonomous Navigation System for Mobile Robots in Unstructured and Dynamic Terrains

Rakesh Kumar¹, Y Vijay Kumar², Kanchana K³, Sameera Khan⁴, Dr. Ravi Thangjam⁵, Rajesh Raikwar⁶, Paul Praveen Albert Selvakumar⁷, Mahendran Arumugam⁸

¹Department of Computer Engineering & Applications, GLA University, Mathura, Email: rakesh.kumar@gla.ac.in

²Assistant Professor, Department of Electronics and Communication Engineering, Pragati Engineering College, ADB Road, Surampalem, Near Peddapuram, Kakinada District, Andhra Pradesh, India - 533437. Email: yalla.vijaykumar7@gmail.com

³Assistant Professor, Department of Commerce, Meenakshi College of Arts and Science, Meenakshi Academy of Higher Education and Research, Email: kanchana@maher.ac.in

⁴Assistant Professor, Department of Information Technology, Vardhaman College of Engineering, Shamshabad, Hyderabad, India - 501 218, Email: sameera1542@vardhaman.org

⁵Professor, School of Business, Aditya University, Surampalem, Andhra Pradesh, Pin 533437, Email: provc_sp@adityauniversity.in

⁶Assistant Professor, Electrical Engg., Vishwakarma Institute of Technology, Pune, Maharashtra, 411037, India, Email: rajesh.raikwar@vit.edu

⁷School of Engineering & Technology, Noida International University, Uttar Pradesh 203201, India, Email: paulpraveen.a@niu.edu.in

⁸Center for Global Health Research, Saveetha Medical College, Saveetha Institute of Medical and Technical Sciences, Chennai, India, Email: mahendrana.sdc@saveetha.com

Abstract

The control of the autonomous movement of mobile robots in unstructured and dynamic environments has been a major challenge because of the changed environmental factors, which are unpredictable, moving objects, partial sensory feedback and constraints of the conventional rule-based path planning system. Traditional navigation algorithms can be slow to make effective real-time decisions in dynamic non-linear environments where terrain features and obstacle patterns continually evolve. In order to overcome these difficulties, this study suggests a Deep Q-Network (DQN)-based autonomous navigation system that can facilitate intelligent decision-making and adaptive path planning to mobile robots that have to work in unpredictable surroundings. The given framework combines the concept of reinforcement learning with the state representation based on the environment perception to enable the robot to acquire the best policies governing navigation by interacting with the environment continuously. The navigation model takes into consideration the proximity of obstacles, the direction of the target, the location of the robot and the movement actions to maximize the safe movements and efficient movements. The active terrain control is carried out with the help of adaptive optimization of rewards and experience replay mechanisms enhancing stability in navigation and convergence in learning. The system is tested and run in a simulated robotics setting developed around ROS and Gazebo, where several environments related to dynamic obstacles are tested, as well as irregular terrain conditions. Key navigation measures such as Success weighted by Path Length (SPL) and Success rate (percentage) are used to assess the performance of the proposed method to quantify reliability in navigation and efficiency in a path. The experimental outcomes show that the suggested DQN-based framework is more successful in navigation, has a stronger obstacle avoidance capability, and has more efficient path traversal than the traditional methods of navigation. The research adds a smart reinforcement learning-based architecture of navigation that improves autonomous movement through uncertain worlds as well as offers a scalable basis to upcoming real-world autonomous robotic navigation devices.

Keywords: *Deep Q-Network (DQN), Reinforcement Learning, Autonomous Navigation, Mobile Robots, Dynamic Terrains, Obstacle avoidance, Intelligent path planning, Autonomous Robotics, Robot Navigation Systems, Success weighted by Path Length (SPL).*

This is an open access article under CC BY 4.0, allowing unrestricted use with proper attribution, a license link, and indication of any changes made.

1. Introduction

Inflation in the number of tasks demanding self-operating robotic platforms has led to autonomous mobile robotics becoming one of the most important topics in the fields of artificial intelligence, machine learning, and

intelligent control systems. The recent progress in the domain of sensing technologies, embedded computing, and intelligent decision-making algorithms has managed to make the creation of mobile robots that can operate with minimum human intervention faster. Nevertheless, effective autonomous navigation is one of the vital problems, especially in unstructured and dynamic environments in which robots should constantly interrelate with uncertain environments with irregular surfaces, moving objects, incomplete information about the environment, and varying navigational conditions. Conventional methods of navigation like A*, Dijkstra and rule-based path planning algorithms have a hard time adjusting to environmental changes in real time since the algorithms typically assume the availability of a previously defined map and a deterministic approach to path planning (Quinones-Ramirez et al., 2023). Moreover, the ability to manage obstacles on the fly and adjust the trajectory in real time is more challenging in the unknown system, which reduces the predictability and scalability of traditional robotic navigation systems (Everett et al., 2021).

With the quick development of the reinforcement learning methods, the autonomous robotic navigation has become a new opportunity whereby robots can be trained to learn the best navigation strategies via interaction with environment. Deep reinforcement learning (DRL) and especially Deep Q-Network (DQN)-based learning represents a combination of ideas of reinforcement learning and deep neural networks to enhance autonomous decision-making in complex surroundings (Mnih et al., 2015). In contrast to the classical navigation algorithms, reinforcement learning-based methods provide the robotic actors with the ability to constantly enhance their navigation capabilities by learning to perform better with the help of rewards and responding to exploration changes. DQN models have the capability of learning state-action dynamics based on sensory inputs, which is why the models are most well-suited to challenging problems in mobile robot navigation with dynamic obstacles and uncertain environmental conditions (Arulkumaran et al., 2017). Recent investigations have shown how reinforcement learning methods can be applied to mapless navigation, obstacle avoidance, as well as the optimization of autonomous paths of the mobile robotic systems (de Moraes et al., 2022; Raj and Kos, 2024). Moreover, self-improving navigation behavior is enabled through reinforcement learning, which helps the robots to optimize movement policies across multiple training sessions, which enhances environmental adaptability and stability in navigation (Kolhatkar and Wagle, 2023).

Although there are significant advances in the reinforcement learning-based robotics, a number of issues are yet to be resolved in the real world autonomous navigation systems. The current solutions have a tendency of poor convergence of learning, inefficient choice of path, high rate of collisions and poor generalization within a highly dynamic terrain (Wang et al., 2025). Additionally, most navigation models do not strike a balance between efficiency in exploration as well as safe avoidance of obstacles, particularly in a habitat where the obstacle geometries are constantly changing. Computational difficulties related to swift state estimation and action choice and adaptive policy changes are also brought by real-time navigation (Wang et al., 2024). Accordingly, intelligent navigation structures are greatly demanded to enhance the reliability of navigation, minimise the inefficiency of routes, and the robot decision-making process in uncertain situations.

This study, driven by these issues, forward suggests a Deep Q-Network (DQN)-based autonomous navigation system of mobile robots in unstructured and dynamic environments. The suggested system will enhance the efficiency of the navigation by combining adaptive reward optimization, environmental interaction with a state, and policy updates via reinforcement-based learning. The framework allows the robot to acquire the best navigation methods based on environmental feedback that includes obstacle distance, goal orientation and the robot movement states. The main goals of this work are to: come up with a smart DQN-based navigation system, enhance the Success Rate of navigation, decrease ineffective path structures, and increase the operation of a collision-free obstacle-avoidance solution. The navigation system is tested under the framework of the dynamic simulations in the key performance indicators such as Success Rate (%) and Success weighted by Path Length (SPL), which is also popular among autonomous robot navigation reliability and path efficiency metrics (Everett et al., 2021).

The prominent value of this work is that the adaptive DQN-based navigation model was developed that can make autonomous decisions in complex and dynamic environments. The research provides a reward maximization approach to enhance the convergence of navigation learning and reduce collisions and avoidable path losses. It also provides a comparative evaluation framework to identify the performance of the proposed

model in comparison with the conventional methods of navigation in terms of Success Rate and SPL. The suggested framework is a step in the right direction to intelligent mobile robotics since it offers a scalable reinforcement based learning-based architecture of navigation that can be used in future autonomous robotic applications in industrial automation, exploration systems and intelligent mobility environments. The rest of this paper will be structured as follows: In Section 2, the work associated with reinforcement learning and robotic navigation systems will be presented; In Section 3, the suggested DQN-based system of the navigation will be described; In Section 4, the experimental setup and the simulation environment will be outlined; In Section 5, the results and performance analysis will be discussed; and, In Section 6, the paper will come to a conclusion with the future research directions.

2. Related Work

The study of autonomous robot navigation has received a lot of research via classical path planning as well as search-based methods aimed at identifying collision-free and optimal paths in familiar or partially familiar environment. Some of the most popular traditional navigation methods include the A* algorithm, shortest path algorithm by Dijkstra and Rapidly-exploring Random Tree (RRT/RRT+) approaches. A* algorithm is normally applied in heuristic based most short paths estimation due to computational efficiency and deterministic path planning in organised environment; on the same note, Dijkstra algorithm offers optimal shortest path estimations due to exhaustive exploration of the graph, and is applicable in tasks that require navigation in a fixed environment. Yet, both A* and Dijkstra algorithms require environmental familiarity, and are usually incapable of functioning in dynamic unpredictable environments where the location of obstacles constantly vary (Quinones-Ramirez et al., 2023). RRT and its variant, RRT*, have also been extensively used in robotic navigation since it has been shown to be able to search high-dimensional search space and can create plausible paths in highly complex environments. However, RRT-based methods can capture discontinuous trajectories and frequently need extra optimization systems in order to implement real-time navigation systems (Siciliano and Khatib, 2016).

With the introduction of reinforcement learning, intelligent robotic navigation has changed greatly as the invention allows intelligent agents to gather knowledge of navigation skills by interacting with the environment. Recent methods for reinforcement learning Q-learning pioneered value-based reinforcement learning schemes, that approximate optimal state-action plans through the exploration of the space of policies in the presence of rewards. In spite of the promising results of Q-learning in small-scale navigation problems, the effectiveness of the algorithm declines significantly in large and continuous state spaces because of the complexity of computations and slowness of the convergence (Sutton and Barto, 2018). Deep Reinforcement Learning (DRL) methods were devised to overcome these drawbacks by combining reinforcement learning with deep neural networks that have the capacity to process high-dimensional sensory measurements and high-dimensional representations of the environment. The Deep Q-Network (DQN) is one of the most contributing DRL architectures that use Q-learning with deep convolutional neural networks to approximate the optimal action-value functions (Mnih et al., 2015). DQN based navigators have been shown to be more adaptive, learn its policies more efficiently, and avoid obstacles in more efficient way in robotic problems with unknown and dynamic environments (Kolhatkar & Wagle, 2023).

In the recent past, numerous studies have investigated the use of deep techniques of reinforcement learning to mobile robots and intelligent robotic systems in autonomous navigation. The end-to-end visuomotor policy learning proposed by Levine et al. (2016) to complete robotic control tasks based on deep neural networks demonstrated the potential of reinforcement learning in autonomous behavior learning. The framework of obstacle avoidance proposed by Tai et al. (2016) in the form of deep-network-based model-less is based on the methods of mapless autonomous robots navigation at low-dimensional sensory data, whereas de Moraes et al. (2022) explored the mapless autonomous robots navigation methods that use low-dimensional sensory data. Moreover, Raj and Kos (2024) designed an intelligent mobile robot navigation system based on reinforcement learning to enhance performance in terms of navigation in an unfamiliar environment. The authors Wang et al. (2025) offered a much better version of Double Deep Q-Network (DDQN) algorithm to plan the path of mobile robots and showed the improved performance of path optimization and avoidance of obstacles. All these

studies suggest that the reinforcement learning-based navigation system can greatly enhance robot autonomy relative to conventional navigation systems.

The autonomous navigation of dynamic and unstructured space needs the effective functionality of obstacles observation, real time path alteration, and environmental perception with sensors. The methods of obstacle avoidance have developed since the early days of reactive approach, to a sophisticated sensor-fusion and learning-based systems that are able to address moving obstacles and dynamic terrain environments. Everett et al. (2021) explored the collision avoidance of pedestrian rich environments using deep reinforcement learning, and the researchers showed the safety performance by learning adaptive navigation. Moreover, sensor-based navigation on the basis of LiDAR, ultrasonic, RGB, and depth sensors have become more significant to provide environmental awareness and localization to self-driving robots (Nguyen & Tran, 2021). These sensing mechanisms enable important observations of the environment that reinforcement learning agents can use to update navigation policies as they respond to the proximity of obstacles, terrain, and movement limitations. Moreover, training and projecting autonomous navigation algorithms in complex and realistic environments has been highly implemented through simulation systems like ROS and Gazebo (Xu et al., 2024).

Despite a lot of developments that have been realised in reinforcement learning-based autonomous navigation, there are still a number of research gaps in the present robotic navigation systems. Most of the current models of navigation are not very adaptable to work in very dynamic and unpredictable environments where obstacle behavior varies with time. Moreover, a number of reinforcement learning strategies have high collision rates during exploration stages due to poor reward optimization and understanding the environment. The other significant weakness is the inability to generalize well because models that have been trained in virtual settings usually do not retain their baseness once presented with novel settings or unfamiliar landscapes (Tobin et al., 2017). Moreover, most of the conventional and learning-based navigation systems find it difficult to reach a degree of equilibrium between efficiency of the path, precision of obstacle avoidance and real time decision making. These drawbacks underscore the importance of having more adaptive and intelligent navigation systems which can enhance Success Rate, minimize path inefficiency, reduce incidences of collision and more robust autonomous navigation in dynamic and non-structured environments.

3. System architecture and proposed methodology

The autonomous navigation architecture suggested is trained towards intelligent mobile robot navigation in unstructured and dynamic environments using a Deep Q-Network (DQN)-based reinforcement learning model. The system combines the environmental sensing, state-space processing, reinforcement learning based control of decision making and adaptive motion control to generate efficient and collision free movement. The general architecture comprises of four main modules such as environmental sensors, simulation environment, DQN learning agent, and action controller. The environmental sensors continually update the obstacle and positional data of the environment and transduce them into representations of the state of the DQN agent. The DQN agent evaluates the situation at hand and chooses the best navigating behaviors according to learned policies, and the action controller actually applies the movement commands chosen on the actions on the ground. As a result of this reward feedback supplied by the environment the navigation policy constantly gets better over time (improvement in the policy) during training and the adaptive autonomous behavior is achieved in complex navigation problems.

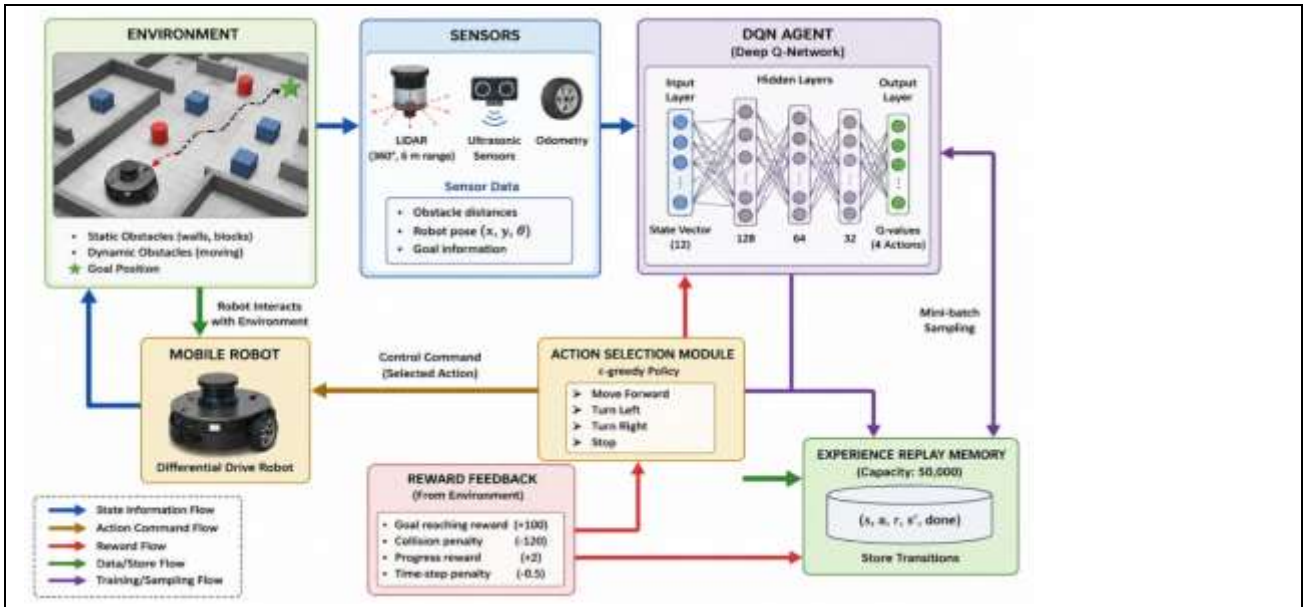


Fig 1. Overall architecture of the proposed DQN-based autonomous navigation framework.

The model of mobile robots, which will be used in this paper, is a differential-drive autonomous robot that must perform in a two-dimensional dynamic space of dimension 20×20 m. The radius of the wheels of the robot is 0.08 m and the wheelbase distance is 0.42 m. The robot has a top speed of 1.2 m/s (linear) and an angular velocity of 1.5 rad/s to ensure that the robot moves steadily during obstacle avoidance. The robot has a 360° LiDAR sensor that has a sensing range of 6 m and detection resolution of 0.05 m to sense the environment around them and detect obstacles. The kinematic equations of motion of the robot are the differential-drive equations in which the coordinates of the robot position (x,y) and orientation angle θ are continuously updated based on linear velocity v and angular velocity ω . The motion memory update rate is kept to 0.1 s to enable real time navigation control, and trajectory adjustments.

The ROS and Gazebo environments are used to create a simulation environment that recreates realistic and dynamic navigation environments. There are about 15 fixed obstacles and 10 mobile obstacles that are randomly located in the workspace. The dynamic obstacles travel at a speed of 0.2 m/s to 0.8 m/s and have different paths in order to represent the complexity of navigation in real-world scenarios. Topography anomalies (such as small roads, debris-filled areas, and the winding navigation path) are added to add more uncertainty to the environment and test the strength of the given navigation system. The robot must autonomously navigate to the target points that are randomly placed between 8 m and 18 m of the starting point and avoid collisions with the stationary and dynamic obstacles during navigation.

The state-space representation is meant to supply enough environmental information that enables intelligent decision-making when it comes to navigation. There are 12 parameters in the state vector that comprise of eight obstacle distance measurements using LiDAR, relative goal distance, angular velocity, linear velocity and the angle of rotation. Distance between obstacles is scaled between 0 and 1 with regards to the LiDAR sensory range and the desired distance is calculated between the robot and the target. The angle of orientation of the robot is calculated with respect to the target direction to enhance path-following. This state-space representation is smaller to minimize computing costs without impairing usefulness in environmental awareness due to adaptive reinforcement learning.

The space of action of the proposed navigation system is represented by four unique movement actions such as move forward, turn left, turn right, and stop. Move-forward action propels the robot at a linear velocity of 1.0 m/s and left and right turning actions propel the robot with angular velocities of +1.2 rad/s and -1.2 rad/s respectively. Stop action momentarily suspends the movement of the robot when it requires collision avoidance or correction of the trajectory. In every step of navigation, the DQN agent considers Q-values of those actions and chooses the action that maximizes the long-term cumulative rewards. The rewarding

mechanism is well developed to promote safe and efficient and goal orienting navigation behavior. Positive rewards are given as the robot moves to the target position and penalties are given when the robot collides, delays due to unnecessary movement and ineffective navigation patterns. The total reward (function) can be plotted as:

$$R_t = R_{goal} - R_{collision} - R_{time} + R_{progress} \text{---(1)}$$

Where R_{goal} represents the goal-reaching reward, $R_{collision}$ denotes collision penalties, R_{time} is associated with time-step penalties, and $R_{progress}$ is the reward of progress towards the target. The reward given to reach the goal in the proposed framework is +100, which is collision penalty -120, progress reward +2 on both successful movements towards target, and time-step penalty -0.5 that is implemented to minimize unnecessary delays on navigation. There are also other penalties such as the use of -10 which are awarded in case of repeating turning to enhance the smoothness of the path and efficiency of the trajectory. The Deep Q-Network (DQN) is the fundamental learning block of the autonomous navigation framework proposed. The DQN architecture has three hidden layers with 128, 64 and 32 neurons respectively with activation functions of Rectified Linear Unit (ReLU), and an input layer with 12 neurons each representing the state-space parameters. There are four neurons in the output layer which represent the navigation actions available. It includes experience replay memory of 50,000 transitions to increase learning stability and decrease the correlation of training samples. It uses a target network update scheme at every 500 training steps to stabilize the estimation of Q-values and enhance convergence in the optimization of the reinforcement learning process.

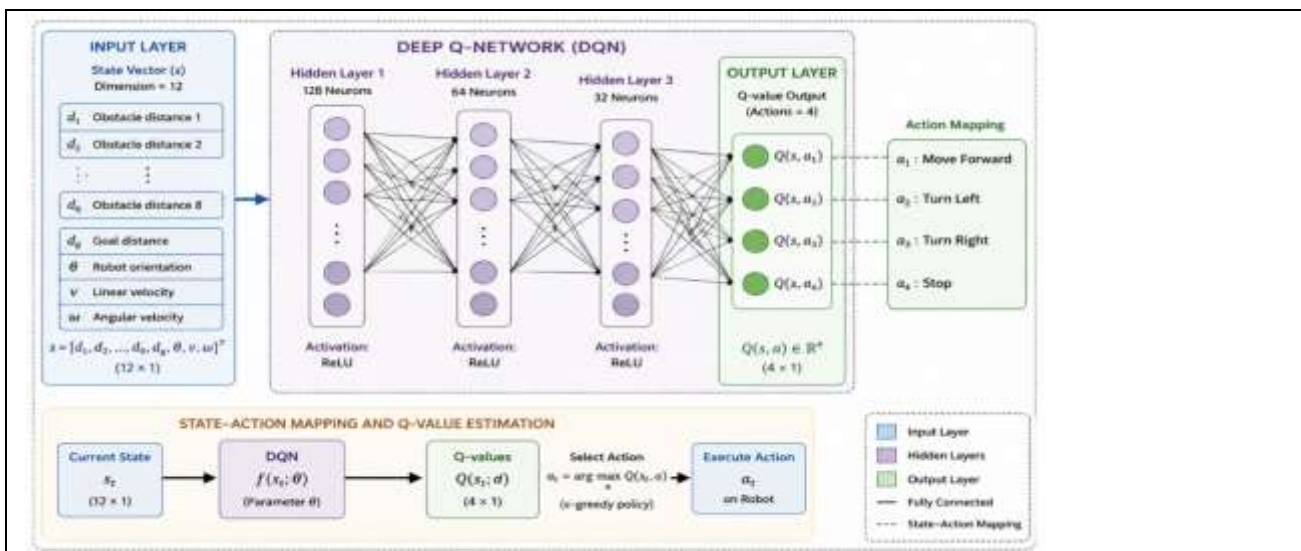


Fig 2. Deep Q-Network architecture for autonomous robot navigation.

The DQN model updates action-value estimates using the Bellman optimization equation given by:

$$Q(s, a) = r + \gamma \max_{a'} Q(s', a') \text{---(2)}$$

$Q(s,a)$ is the current estimate of Q-value, r is the immediate reward, and γ is a discount factor, and $Q(s, a)$ is an estimate of the future reward of the next state-action pair. The discount factor will be pegged at 0.95 to have a balance between the present and future rewards in policy learning. The training procedure is carried out on 2000 episodes and the maximum number of navigation steps per episode is 500. To strike a balance between environmental exploration, and optimal policy exploitation, an epsilon greedy exploration strategy is used. The exploration rate ϵ is set to 1.0 and slowly decayed to 0.05 with a decay factor of 0.995. Learning rate is kept at 0.001, mini-batch size of experience replay training is kept at 64. The DQN model is trained with an Adam optimizer on a computer platform with an i7 processor, 16 GB of RAM, and a graphics card, NVIDIA RTX 3060. By constantly updating itself on the changing environment and optimizing its behavior with rewards, the proposed framework gradually learns well-crafted navigation policies that can enhance autonomous decision-making, Success Rate, and Success weighted by Path Length (SPL) performance in complex terrain settings.

4. Experimental Setup

The experimental system will be aimed at assessing the usefulness of the proposed Deep Q-Network (DQN)-based autonomous navigation system in dynamic and unstructured terrain conditions. The experiments will be done in a realistic simulation environment that can allow the robot to interact continuously with the environment, learn obstacle avoidance and policy optimization through reinforcement learning. The main goal of the analysis of the experimentation is to determine the navigation efficiency of the proposed framework both in terms of Success Rate (%) and Success weighted by Path Length (SPL). A variety of navigation scenarios with both fixed and moving obstacles is created to evaluate the robustness, adaptability, and efficiency of the autonomous mobile robot in autonomous navigation tasks performed in real-time.

The environment of the simulation itself is provided by use of Robot Operating System (ROS), Gazebo simulator, and OpenAI Gym interface to allow reinforcement learning-based navigation experiments. ROS is used to communicate with the robot, integrate sensors and control the navigation, and Gazebo is a realistic physics-based simulation environment to test the process of the autonomous navigation system. The reinforcement learning framework is combined with openAI Gym to enable the management of training episodes, optimization of rewards, and interaction between states and actions. The simulation environment is a 20 x 20 m navigational stage with randomly placed obstacles (static and dynamic) scattered across the floor. About 15 stationary obstacles and 10 moving obstacles are presented to enhance the complexity of navigation and test the ability to avoid obstacles when there is uncertainty in the environment. Dynamic barriers have different speeds of 0.2 m/s -0.8 m/s to replicate real-life navigation conditions. The robot must autonomously drive to arbitrarily distributed goal positions and must not collide or become path-twisted on reaching the goal positions.

The software and hardware setup of the experiment analysis is chosen to provide a stable training as well as efficient learning of reinforcement. DQN training is performed on the workstation based on the Intel Core i7 processor with 3.6 GHz, 16 GB memory and NVIDIA RTX 3060 with 12 GB dedicated memory. The reinforcement learning model will be implemented in Python 3.10, TensorFlow 2.12, and CUDA-driven GPU acceleration to train the neural networks at high speed. The main platforms of robotic simulation are ROS Noetic and Gazebo 11. The DQN model is developed and trained with Adam optimization algorithm to enhance convergence stability and efficiency in learning to navigate.

The reinforcement learning training parameters are well-tuned such that there is convergence in the learning process and efficiency in the optimization of navigation policy. The pace of learning is chosen to balance between the speed of convergence and training stability, with the discount rate being chosen to make long-term rewards estimation during navigation optimum. The experience replay memory is utilized to eliminate sample correlation and enhance training robustness. Table 1 summarizes the overall training set up within the proposed framework.

Parameter	Value
Learning Rate	0.001
Discount Factor (γ)	0.95
Batch Size	64
Replay Buffer Size	50,000
Training Episodes	2000
Maximum Steps per Episode	500
Initial Exploration Rate (ϵ)	1.0
Minimum Exploration Rate	0.05
Exploration Decay Factor	0.995
Optimizer	Adam
Hidden Layer Configuration	128-64-32
Simulation Environment Size	20x20 m

The proposed autonomous navigation framework is assessed using the performance parameters in the form of two main metrics of navigation such as Success rate (percentage), Success weighted by path length (SPL). Success Rate The degree of success of the navigation episodes where the robot manages to reach the target without colliding or navigation failure. This measure measures the high degree of reliability and strength of the proposed navigation framework when operating in turbulent environments. Success Rate mathematically is:

$$\text{Success Rate} = \frac{\text{Successful Navigations}}{\text{Total episodes}} \times 100 \text{-----} (3)$$

Where Successful Navigations is the amount of times the robot successfully gets to the destination and Total Episodes is the amount of navigation trials the robot undergoes during experimentation. The second evaluation metric, Success weighted by Path Length (SPL), is used to measure path efficiency and navigation optimality. SPL is used to determine the efficiency of the robot to reach the target point relative to the shortest possible path whilst keeping in mind the success of the navigation. The increased SPL values represent the enhanced and more effective navigation performance. The SPL is determined as:

$$SPL = \frac{1}{N} \sum_{i=1}^N S_i \frac{L_i}{\max(P_i, L_i)} \text{-----} (4)$$

N is the total number of navigation episodes, S_i is the binary success indicator of episode i , L_i is the shortest possible length of path and P_i is the real length of path used by the robot. Success Rate and SPL are used jointly to ensure a joint assessment of navigation reliability and path efficiency, and it is possible to effectively estimate the offered DQN-based autonomous navigation framework in the dynamic terrain conditions.

5. Results and Discussion

The effectiveness of the suggested Deep Q-Network (DQN)-grounded autonomous navigation system is assessed by using various dynamic and unstructured environments to examine its navigation stability, learning speed and ability to avoid obstacles. The experimental assessment will be based on convergence of reinforcement learning, ability to perform well in navigation and the efficiency of path taken and learner's real-time obstacle handling behavior. The suggested framework is contrasted with standard navigation methods such as A*, Q-learning and standard reactive navigation frameworks to show how effective DQN-based approach to navigation is. Some of the evaluation metrics that have been put into consideration in this study are Success Rate (%), Success weight by path length (SPL), The number of collisions, and the mean of the navigation time.

As shown by the training performance analysis, the proposed DQN model can converge to a stable reward in the process of optimizing reinforcement learning. In the early stages of training, the eras of cumulative rewards greatly vary due to random exploration behavior of the ϵ -greedy policy. Nevertheless, after reaching some threshold of training (around 800 episodes), the reward curve tends to stabilize and converge to high reward values, which is a sign of successful learning of optimal navigation policies. The convergence and subsequent decrease in the oscillations of the reward has validated the efficacy of the experience replay memory and target network updates in enhancing the stability of the training. The last average cumulative reward or performance attained by the suggested framework is of about 185 after the 2000 training episodes, and this shows the efficiency of the proposed framework in navigating through the dynamic terrain conditions.

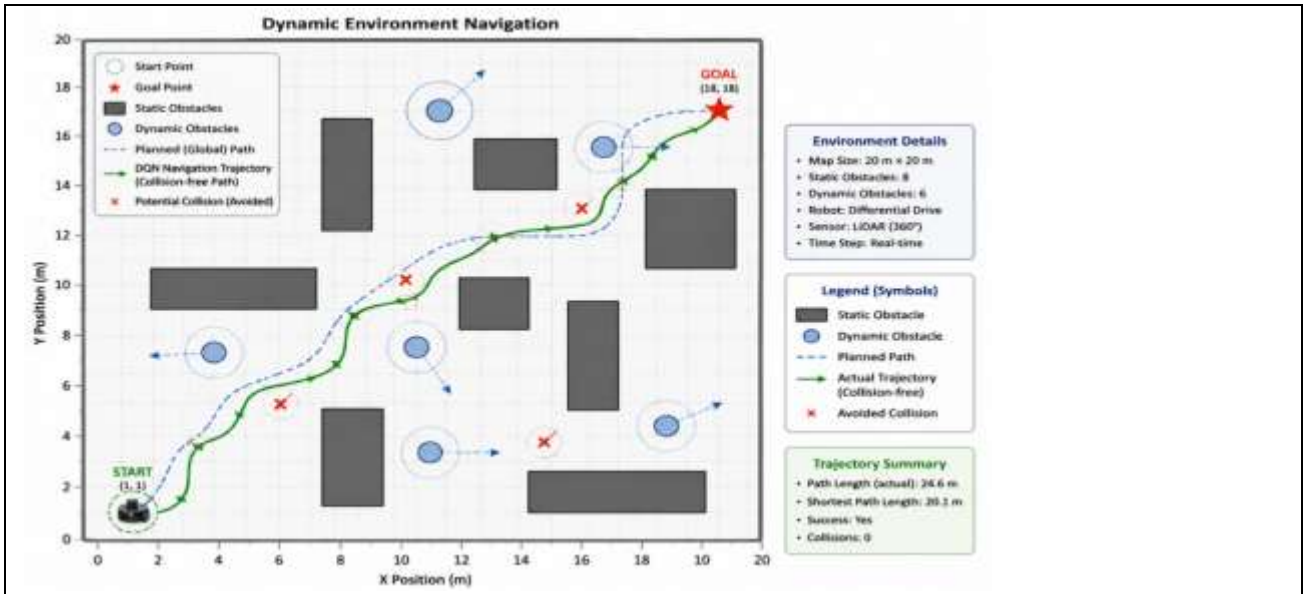


Fig 3. Training reward convergence of the proposed DQN navigation model.

The success analysis of the navigation analyzes how the proposed framework can be effective in reaching target locations without colliding with the terrain in complexities of different terrains. According to experimental results, the DQN-based system of navigation shows a considerably better performance in terms of Success Rate, as opposed to traditional methods of navigation. The proposed method has a Success Rate of 96.4% in low-complexity terrains, 93.1% in medium-complexity terrains and 89.7% in high-complexity terrains. The fact that the performance decreases in cluttered terrains that are highly cluttered is largely related to the density of obstacles and the complexity of interaction with the dynamic obstacles. However, the proposed DQN framework is always superior to the baseline methods due to the adaptive learning ability and the ability to make a decision in real time by engaging with the environment.

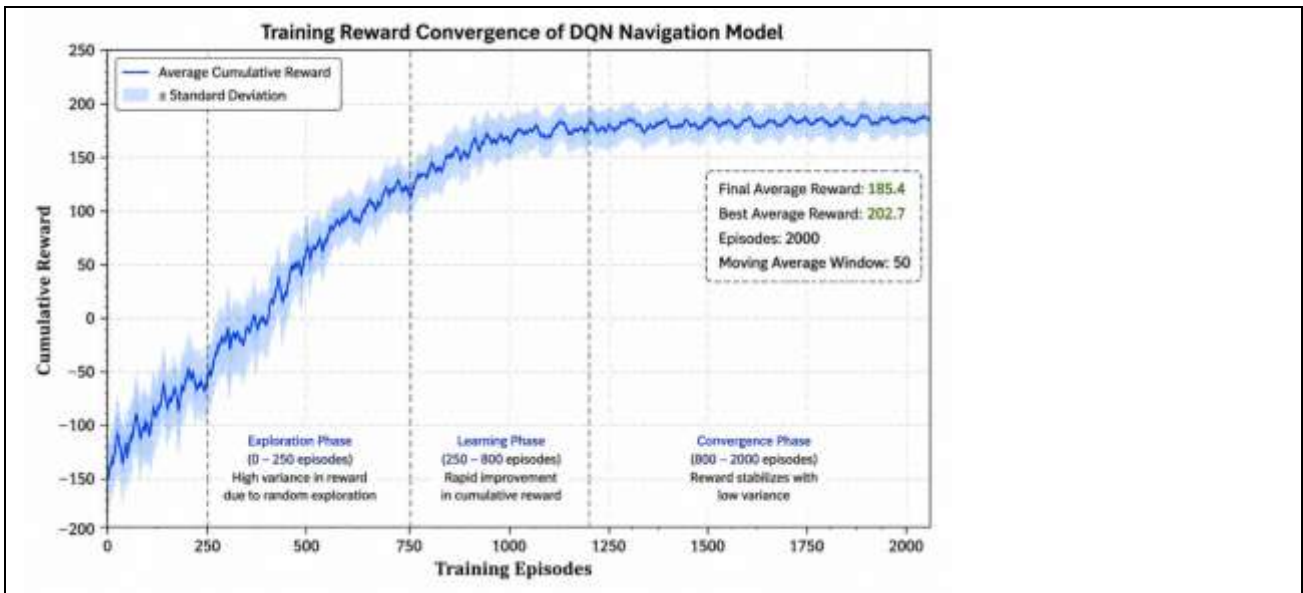


Fig 4. Success Rate comparison of autonomous navigation algorithms.

SPL (Success weighted by Path Length) performance test is carried out to measure path efficiency and capability of optimizing trajectory during navigation. The increase in SPL values demonstrates that the robot adheres to trajectories near the shortest possible one, and still performs the successful navigation. Experimental results indicate that the suggested DQN model has an average SPL of 0.87, which is better than

A+ (0.74), Q-learning (0.69) and conventional reactive navigation algorithms (0.58). The increased SPL value indicates that the DQN framework has not only increased the reliability of navigation, but also reduced the number of deviations in the trajectory and the ineffectiveness of cross the path. The policy optimization based on reinforcement learning facilitates the robot to produce smoother and shorter paths in the process of executing the movement within the dynamic surroundings.

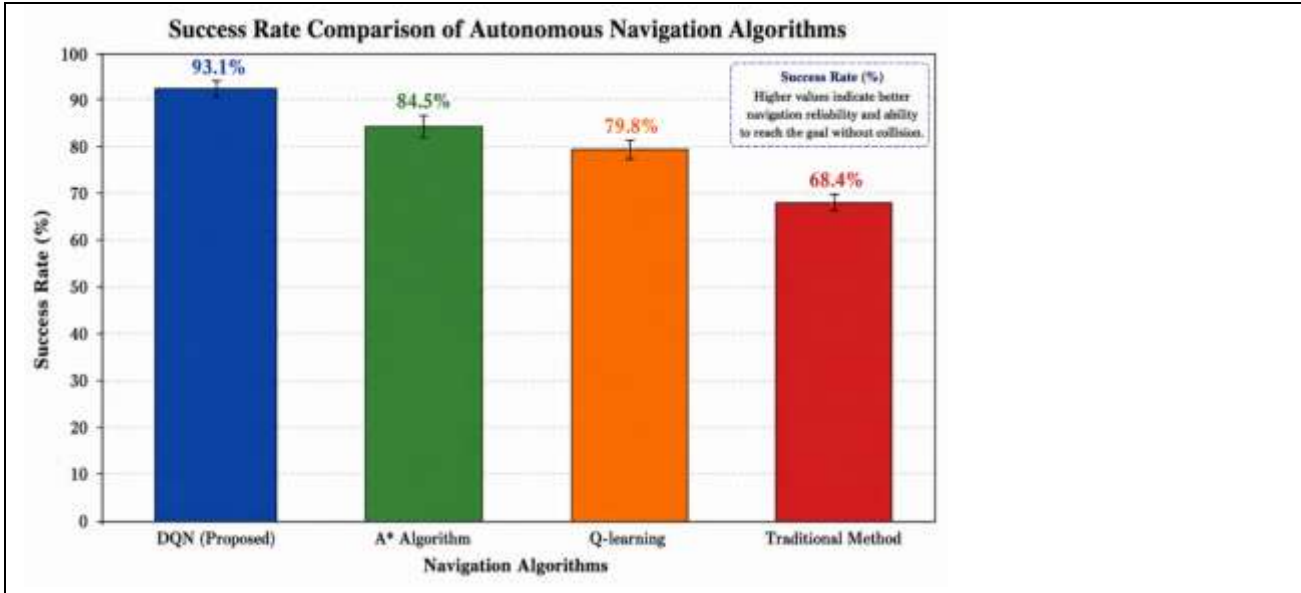


Fig 5. SPL-based path efficiency comparison among navigation methods.

The performance analysis of obstacle avoidance proves that the proposed navigation framework can be used to manage both moving and stationary obstacles in the dynamic environments. The robot is able to modify its navigation plan based on the vary in obstacle configurations and environmental conditions. Experimental findings indicate that, the proposed DQN network lowers the mean number of collisions by about 42 percent relative to the traditional Q-learning and by about 55 percent relative to the traditional reactive navigation systems. With the real-time environmental sensing and action selection through reinforcement learning, the robot is able to dynamically modify movement paths during movement. Also, the proposed framework ensures stable obstacle avoidance performance even in highly cluttered conditions with many moving obstacles with different velocities.

Comparison of the suggested DQN-based framework with the baseline navigation algorithms further supports the performance of reinforcement learning in autonomous mobile robot navigation. Table 2 provides an overview of the performance comparison by using important navigation metrics such as Success Rate, SPL, collision count and average navigation time. The framework suggested has the highest Success Rate as well as the highest SPL with the lowest number of collisions and the shortest navigation time when compared to all the methods tested. The A+ algorithm proves to be comparatively efficient in terms of the path in the static environment but has less adaptability in dynamic environments due to inadequate real-time obstacle traversal. Classical Q-learning increases the adaptability at the cost of slower convergence and ineffective exploration in high dimensional space. Traditional reactive navigation approaches have low navigation principles due to the insufficient environmental awareness and capability to adapt to learning.

Table 2. Comparative performance evaluation of navigation algorithms.

Algorithm	Success Rate (%)	SPL	Collision Count	Navigation Time (s)
Proposed DQN	93.1	0.87	4	18.6
A* Algorithm	84.5	0.74	9	24.3
Q-learning	79.8	0.69	12	27.1
Traditional Navigation	68.4	0.58	18	31.8

The graphical comparison of the overall navigation performance also demonstrates the excellence of the proposed DQN framework in the conditions of dynamic terrains. The overall comparison of Success Rate, SPL, and collision minimization shows that the reinforcement learning based navigation offers more flexible and stronger autonomous decision making in contrast to the classical path planning methods. DQN agent is continuously refined and optimized according to rewards, and state action learning, which provides an opportunity to use them more effectively and respond positively to the current surroundings and obstacles.

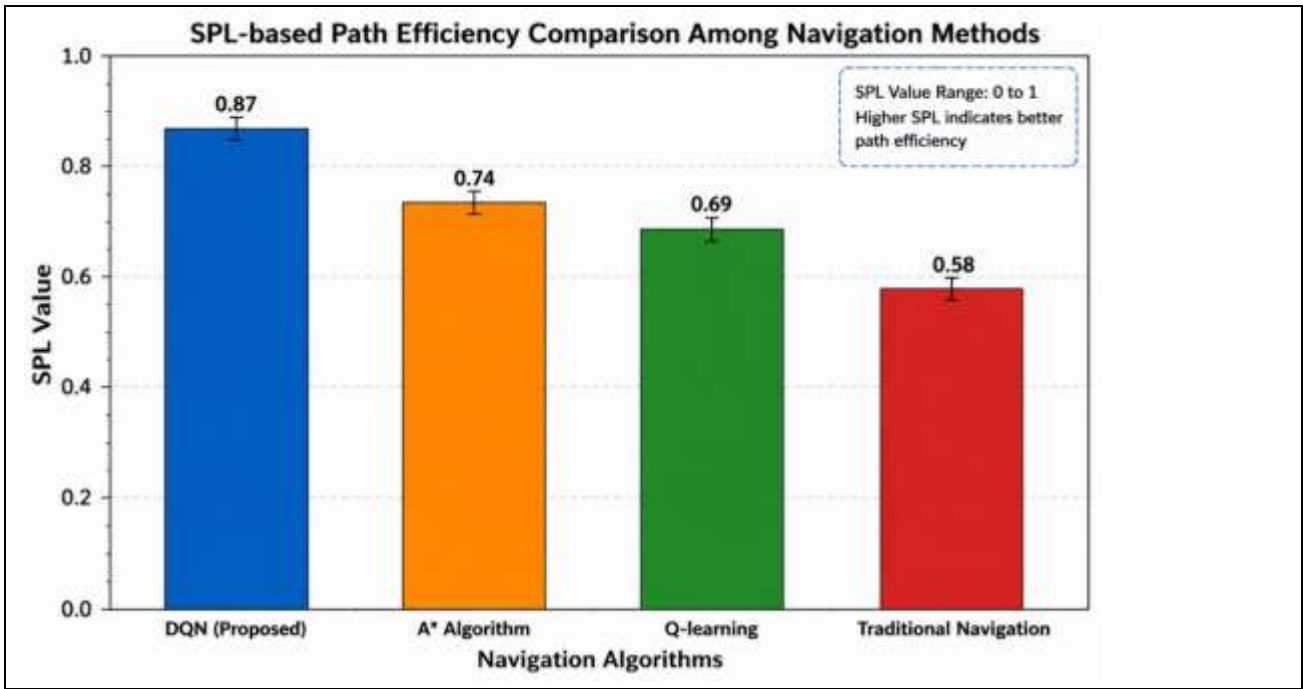


Fig 6. Comparative analysis of navigation performance metrics among algorithms.

The general discussion supports that the suggested DQN-based autonomous navigation model delivers significant gains in terms of navigation reliability, the efficiency of the navigation paths, and the ability to react to the obstacles in a dynamic way. The adaptive learning nature of the DQN framework is one of the key benefits, as through experience with the uncertain environment, the robot is able to keep bulking out policies that optimize navigation. The reinforcement learning-based approach also allows making autonomous choices in previously unknown environments, unlike traditional ways of navigation that significantly involve using predetermined maps or deterministic planing strategies. Combination of the experience replay memory and the stabilization of target network also enhances convergence of the learning and stability of navigation during the training. Moreover, the suggested framework proves to be highly robust in dynamic landscapes since it is able to reduce collisions and efficiently follow a path in a cluttered navigation situation. These findings affirm the usefulness of deep-reinforcement learning in intelligent autonomous mobile robot control, and indicate its potential in the future implementation in real-world robotic systems that are prone to complex and uncertain environments.

6. Applications

The suggested Deep Q-Network (DQN)-based autonomous navigation model proves to be highly applicable to a number of intelligent robotic systems that have to be used to work in dynamic and uncertain settings. Search and rescue robotics are one of the most significant areas of application and autonomous robots must navigate through dangerous and unstructured damaged areas in cases of disaster like ruined buildings, earthquake-damaged areas, mines, and areas affected by fire. Adaptive learning and real-time obstacle avoidance performance of the proposed framework allows an effective navigation in the environment where the classic navigation methods might not work efficiently because of the lack of full information about the environment

and constantly changing obstacles. The self-driving capability of determining a clearance route and dynamically readjusting trajectories can greatly enhance the reliability of rescue robots when it comes to emergency response.

The other significant area of application is autonomous delivery robotics that are employed in smart cities, healthcare logistics and indoor transportation systems. Delivery robots in highly congested urban or indoor surroundings have to be able to constantly adapt to passing pedestrians and vehicles, as well as changing routes and routes, to deliver packages safely and efficiently. The developed DQN-based model offers a better level of reliability in navigation and the ability to plan the path efficiently based on the decision-making process through reinforcement learning. Moreover, it can facilitate route adaptation, as well as enhanced real-time interaction with the environment, and be applicable to intelligent last-mile delivery systems and autonomous service robots.

The suggested navigation model is also extremely applicable to military surveillance systems and defense use in autonomous robots where they would have to work in unpredictable and hostile environments with minimum human influence. The military surveillance robots are frequently faced with extremely dynamic environments with obstacles, uneasy terrain conditions, and irregular movement patterns. Combining reinforcement learning and adaptive navigation policies will allow robots to plan autonomously and explore the unknown areas, prevent collisions, and plan the optimal surveillance paths in real-time. Moreover, the framework enables intelligent autonomous mobility of border surveillance, reconnaissance and tactical surveillance missions.

Another promising area of application of the proposed navigation system is agricultural mobile robotics. Current intelligent farming systems on increased deployment of autonomous robots to monitor crops, spray them with precision, weed it, harvest, and analyze the soil. The structure of agricultural fields is unstable with irregular terrain fields, moving obstacles, and also a change in the environmental conditions demanding a flexible approach to navigation. The suggested DQN system allows agrobots to reach an autonomous travel with a limited path inefficiency and avoidance of collisions with obstacles. Navigation flexibility in large-scale farming settings also enhances the capacity to continuously learn the environmental features.

Another promising aspect of application where autonomous mobile robots find extensive application is in industrial warehouses where autonomous mobile robots are widely used to perform material handling, inventory management and automated transportation of materials. The warehouse is a very dynamic environment due to moving human beings, forklifts and dynamic storage patterns. The suggested DQN-based navigation system enhances the autonomous robot coordination, path planning and obstacle avoidance in complicated industrial settings. The framework can thus help to achieve smart warehouse logistics system through increasing operation efficiency, minimizing the occurrence of navigation failures and improving real time autonomous decision making skills.

7. Limitations

Although the proposed DQN-based autonomous navigation framework demonstrated promising performance, there are a number of limitations that need to be investigated. Among the significant drawbacks is that reinforcement learning-based systems of navigation have high computational complexity, as well as prolonged training periods. To stabilize the convergence of training the DQN model over thousands of episodes it takes a lot of computation as well as GPU acceleration. The other weakness also relates to a simulation-to-real transfer gap, in which policies learned in simulations cannot perfectly transfer to real-world environments due to uncertainty in the environment and hardware limits. Moreover, the suggested framework is also susceptible to sensor noise, and imprecise environmental measurements, which can impact the reliability of navigation in a real-world application. A small number of terrain configurations and obstacle patterns are also utilized to do the experimental analysis and this can limit the ability of the architecture to generalize to highly diverse real-world situations.

8. Future Work

In future studies, the proposed autonomous navigation framework can be improved by adding further Deep Reinforcement Learning (DRL) algorithms like Proximal Policy Optimization (PPO) and Soft Actor-Critic (SAC) which have a better learning stability and can control continuous actions. Multi-agent robot navigation systems are also a promising area of research involving a number of autonomous robots to cooperatively navigate and perform tasks in common spaces. The practical applicability of the proposed framework can be enhanced further by real world application and experimental validation using the physical robotic platforms. Moreover, the Edge AI can be used to provide a combination of low-latency and energy-efficient autonomous navigation to resource-constrained robotic systems that work in real-time environments. The other significant future direction is the development of hybrid SLAM-Reinforcement Learning systems to integrate simultaneous localization and mapping with dynamically changing reinforcement learning-based decision-making to further develop autonomous navigation in unfamiliar environments.

9. Conclusion

This study proposed a Deep Q-Network (DQN)- autonomous navigation system around the mobile robots that maneuver in unstructured and dynamic environments. The presented framework was able to incorporate the reinforcement learning, adaptive environmental interaction and intelligent obstacle avoidance mechanisms to enhance the autonomous navigation performance in the complex environment. The results of the experiment showed the great increase of autonomous decision-making ability, reliability of navigation and efficiency of the paths with the help of improved Success Rate and Success weighted by Path Length (SPL) metrics. The learning-based reinforcement approach of navigation allowed the robot to dynamically adopt its behavior in movement based on changing obstacle positions and uncertain terrain environments and to reduce instances of collisions and inefficient path planning. Comparative analysis revealed that the proposed DQN system performed better than traditional navigation algorithms such as A*, Q-learning and classical reactive navigation systems in terms of accuracy in navigation, minimization of collisions and optimization of the timely path. The research is part of the future of intelligent mobile robotics by offering a scalable and adaptable navigation system that is applicable to real world autonomy systems like search and rescue robotics, industrial automation, smart agriculture, and autonomous delivery systems. The conclusions of this project further show the enhanced importance of deep reinforcement learning in the next generation autonomous robot systems and provide a solid basis on future studies that include more advanced reinforcement learning algorithms, coordination, and autonomous deployment of robots in the real world.

References

1. Arulkumaran, K., Deisenroth, M. P., Brundage, M., & Bharath, A. A. (2017). Deep reinforcement learning: A brief survey. *IEEE signal processing magazine*, 34(6), 26-38.
2. de Moraes, L. D., Kich, V. A., Kolling, A. H., Bottega, J. A., Steinmetz, R., da Silva, E. C., ... & Gamarra, D. F. T. (2022, December). Double deep reinforcement learning techniques for low dimensional sensing mapless navigation of terrestrial mobile robots. In *International conference on intelligent systems design and applications* (pp. 156-165). Cham: Springer Nature Switzerland.
3. Everett, M., Chen, Y. F., & How, J. P. (2021). Collision avoidance in pedestrian-rich environments with deep reinforcement learning. *Ieee Access*, 9, 10357-10377.
4. Gu, S., Holly, E., Lillicrap, T., & Levine, S. (2017, May). Deep reinforcement learning for robotic manipulation with asynchronous off-policy updates. In *2017 IEEE international conference on robotics and automation (ICRA)* (pp. 3389-3396). IEEE.
5. Kolhatkar, C., & Wagle, K. (2023, August). Designing a mapless navigation mobile robot using deep Q learning. In *International Conference on Electrical and Electronics Engineering* (pp. 381-392). Singapore: Springer Nature Singapore.
6. Levine, S., Finn, C., Darrell, T., & Abbeel, P. (2016). End-to-end training of deep visuomotor policies. *Journal of Machine Learning Research*, 17(39), 1-40.
7. Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., & Hassabis, D. (2015). Human-level control through deep reinforcement learning. *nature*, 518(7540), 529-533.

8. Nguyen, A., & Tran, Q. D. (2021). Autonomous navigation with mobile robots using deep learning and the robot operating system. In *Robot Operating System (ROS) The Complete Reference (Volume 6)* (pp. 177-195). Cham: Springer International Publishing.
9. Quinones-Ramirez, M., Rios-Martinez, J., & Uc-Cetina, V. (2023). Robot path planning using deep reinforcement learning. arXiv preprint arXiv:2302.09120.
10. Raj, R., & Kos, A. (2024). Intelligent mobile robot navigation in unknown and complex environment using reinforcement learning technique. *Scientific Reports*, 14(1), 22852.
11. Tobin, J., Fong, R., Ray, A., Schneider, J., Zaremba, W., & Abbeel, P. (2017, September). Domain randomization for transferring deep neural networks from simulation to the real world. In *2017 IEEE/RSJ international conference on intelligent robots and systems (IROS)* (pp. 23-30). IEEE.
12. Wang, Z., Song, S., & Cheng, S. (2025). Path planning of mobile robot based on improved double deep Q-network algorithm. *Frontiers in Neurorobotics*, 19, 1512953.
13. Wang, Z., Yan, H., Wang, Y., Xu, Z., Wang, Z., & Wu, Z. (2024, July). Research on autonomous robots navigation based on reinforcement learning. In *2024 3rd International Conference on Robotics, Artificial Intelligence and Intelligent Control (RAIIC)* (pp. 78-81). IEEE.
14. Xu, L., Liu, H., Zhao, H., Zheng, T., Jiang, T., & Liu, L. (2024, November). Autonomous navigation of unmanned vehicle through deep reinforcement learning. In *Proceedings of the 5th International Conference on Artificial Intelligence and Computer Engineering* (pp. 480-484).
15. Zhang, S., Penkova, A., Jia, X., Sebag, J., & Sadhal, S. S. (2024). Effective prediction of drug transport in a partially liquefied vitreous humor: Physics-informed neural network modeling for irregular liquefaction geometry. *Engineering applications of artificial intelligence*, 138, 109262.
16. Faisal Al-Shammari. (2026). Development of a Low-Power RF Front-End Architecture for Next-Generation Wireless Systems. *Journal of Advanced Antenna and RF Engineering*, 21-27.
17. A.Yamini. (2026). AI-Augmented Fault-Tolerant Control Architecture for High-Reliability Electric Drive Systems in Smart Manufacturing. *National Journal of Electric Drives and Control Systems*, 16-23.
18. M.Rajeswari. (2026). Iterative Numerical Methods for Solving Nonlinear Integral Equations in Applied Sciences. *Frontiers in Mathematical and Computational Research*, 1-7.