A

Minor Project Report

On

Design & Development of Autonomous Surveillance RobotPrototype

In partial fulfillment of the requirements for the award of the degree of

Bachelor of Technology
in
Mechanical Engineering

Submitted by

Narayanan PP, Maulana Azad National Institute of Technology

Palingi Sandeep, Maulana Azad National Institute of Technology

Nishant Singh, Maulana Azad National Institute of Technology

Pratiksha Kumari, Maulana Azad National Institute of Technology

Guided by : RK Dwivedi

<u>CERTIFICATE</u>

MAULANA AZAD NATIONAL INSTITUTE OF TECHNOLOGY BHOPAL (M.P)

Certified that this project report **“<u>Design & Development of Autonomous Surveillance RobotPrototype</u>”** is the bonafide work of

Narayanan PP , Nishant Singh, Palingi Sandeep, Pratiksha Kumari

who carried out the project work under my supervision.

Dr. R.K. Dwivedi Sir
MECHANICAL ENGINEERING

Design & Development of Autonomous Surveillance RobotPrototype

Narayanan PP, Maulana Azad National Institute of Technology

Palingi Sandeep, Maulana Azad National Institute of Technology

Nishant Singh, Maulana Azad National Institute of Technology

Pratiksha Kumari, Maulana Azad National Institute of Technology

Mentor: Dr. R.K. Dwivedi, Professor, Department of Mechanical Engineering, MANIT Bhopal

1. Abstract

Enhanced security measures in public spaces, warehouses, factories, and other settings require the use of autonomous surveillance robots that can monitor and detect any suspicious activity. This project aims to design and develop a prototype of such a robot, capable of navigating a given map, detecting unauthorized individuals, and conserving electricity by turning off unnecessary lights. The robot is equipped with a variety of sensors and cameras that enable it to move autonomously, avoid obstacles, and perform surveillance operations. Our approach involves the use of a combination of technologies, including reinforcement learning-based navigation, lidar-based obstacle detection, and vision-based detection of unauthorized individuals. The robot's mechanical design features a rocker boogie mechanism that ensures stability, increased climbing heights without complex suspension systems, and flexibility to accommodate various use cases and scenarios. Additionally, it includes a camera stand for placing cameras and a telescope arm for switching on or off any necessary equipment. The successful implementation of our approach through a working prototype demonstrates the effectiveness of our solution in enhancing security in different settings.

2. Introduction

The development of autonomous surveillance robots has become increasingly important in recent years due to the need for enhanced security measures in various settings. Traditional approaches for robot navigation, such as A* algorithm or deterministic control system methods, fail to consider the dynamic nature of the real world. Factors such as friction coefficient, battery levels, and other parameters can significantly impact the robot's movement and trajectory. In order to make the robot more robust and adaptable to the real world, we propose using a Deep Reinforcement Learning (DRL) based approach for navigation.

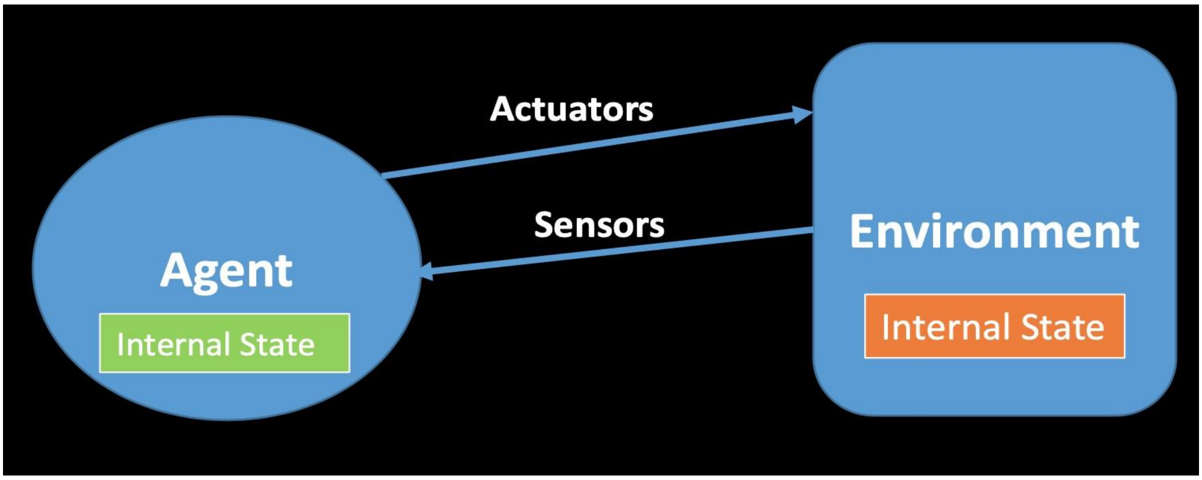
DRL has been shown to be an effective method for training autonomous agents to navigate complex and dynamic environments. The use of reinforcement learning allows the robot to learn from its experiences and make decisions based on its current state, as well as take into account uncertainties in the environment. In our project, we implement an Actor-Critic method for reinforcement learning, which allows the robot to learn both a policy and a value function. This enables the robot to make decisions based on its current state, while also taking into account the expected future rewards.

In addition to using DRL for navigation, we also incorporate lidar-based obstacle detection and vision-based detection of unauthorized individuals for surveillance operations. Our approach will be tested and validated using an existing wheeled robot, which will be retrofitted with our software stack. The robot is designed to be equipped with a variety of sensors and cameras, enabling it to move autonomously, avoid obstacles, and perform surveillance operations. We have designed a mechanical structure for the robot using a rocker boogie mechanism, which provides increased stability and climbing capabilities. The robot is also equipped with a camera stand for placing cameras, as well as a telescope arm for switching off or on any electrical devices.

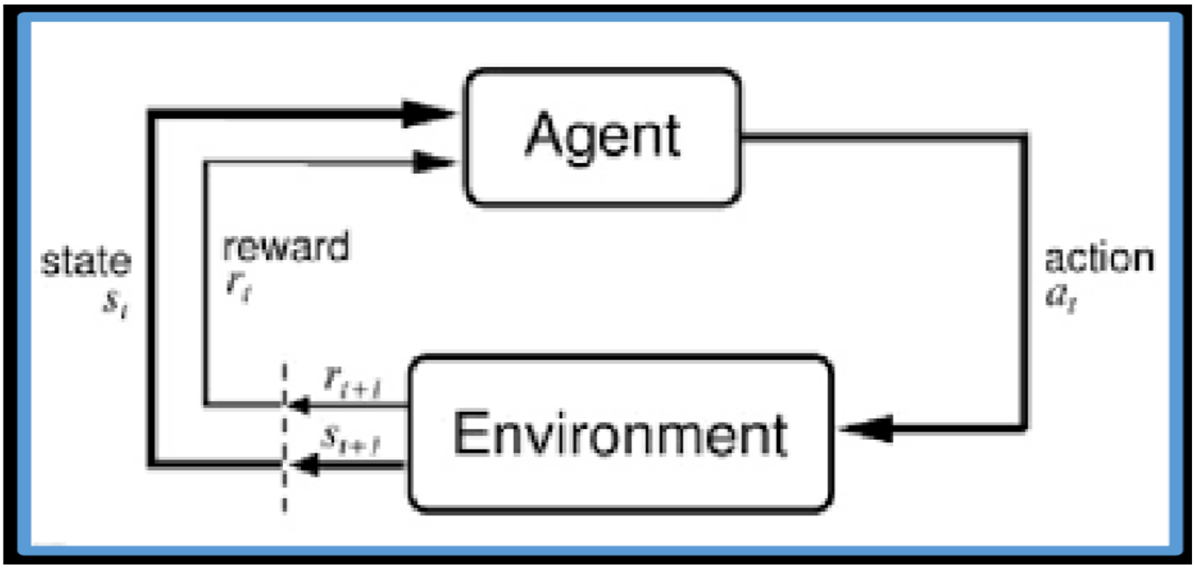
Overall, our project aims to demonstrate the effectiveness of using DRL-based approaches for navigation and surveillance operations in complex and dynamic environments. By incorporating reinforcement learning, we aim to make the robot more robust and adaptable to the real world, and ultimately improve the effectiveness of autonomous surveillance systems in enhancing security measures.

3. Background

Reinforcement learning (**RL**) is a subfield of **machine learning** (**ML**) that addresses the problem of the automatic learning of optimal decisions over time. Reinforcement learning is a sub topic of Artificial Intelligence modelled around the paradigm of Agent, Environment interaction. Agent perceives the environment (Percepts) and acts upon the environment in order to maximize achievement of the required goal (Actions). Looked at from this perspective, an agent is a function that maps the percepts to actions. The function f(.) that maps the sensor to actuators is the control policy that determines how the agent makes the decisions.



At each time step t, agent observes the state from environment and receives a reward from environment. Based upon the state observations and the reward, the agent executes an action based upon a policy. The environment on the other hand receives an action at each time step and emits the next state (t + 1) along with the next reward (t + 1).



The theoretical foundations of RL are based on the Markov decision process (MDP), which is a mathematical framework for modeling decision-making in stochastic environments.

An MDP is defined by a set of states, actions, rewards, and a transition function. At each time step, an agent observes the current state of the environment and selects an action from a set of possible actions. The transition function specifies the probability of moving to a new state given the current state and the action taken. The reward function provides a numerical signal that evaluates the desirability of the new state. The goal of the agent is to learn a policy that maximizes the expected cumulative reward over time.

The Markov property is a fundamental assumption in MDPs, which states that the future state of the environment depends only on the current state and action, and is independent of the history of previous states and actions. This property allows for the modeling of complex decision-making problems in a tractable way, and has led to the development of efficient algorithms for solving MDPs.

One popular approach for solving MDPs is dynamic programming (DP), which involves computing the optimal value function or policy by iteratively applying the Bellman equations. However, DP methods are computationally expensive and require knowledge of the entire MDP model, which is often impractical or impossible to obtain in real-world scenarios.

RL algorithms, on the other hand, learn the optimal policy through trial and error interactions with the environment, without the need for explicit knowledge of the MDP model. RL algorithms use a combination of exploration and exploitation to learn an optimal policy, where exploration involves taking actions to gain information about the environment, and exploitation involves taking actions based on the current policy to maximize reward.

The Advantage Actor-Critic (A2C) algorithm is a reinforcement learning (RL) technique used to optimize the parameters of a policy network. It is a combination of the actor-critic method and the advantage function, which aims to reduce the variance in policy gradients and increase the learning rate.

In the A2C algorithm, the policy network learns to output an action given the current state of the environment. The critic network, on the other hand, estimates the value function, which is the expected cumulative reward from the current state onward. The advantage function is then computed as the difference between the estimated value function and the expected immediate reward.

The A2C algorithm uses the advantage function to reduce the variance in policy gradients, which is a common problem in RL. The policy gradients are computed using the log-likelihood of the chosen action multiplied by the advantage function. By incorporating the advantage function, the algorithm can increase the learning rate and reduce the time required for convergence.

The A2C algorithm can be used in environments with both discrete and continuous action spaces. In environments with discrete action spaces, the policy network outputs a probability distribution over the available actions, and the action is sampled from this distribution. In environments with continuous action spaces, the policy network outputs the mean and standard deviation of a Gaussian distribution, and the action is sampled from this distribution.

One advantage of the A2C algorithm is that it can be parallelized across multiple agents, allowing for faster training on multi-core CPUs or GPUs. This is because each agent can collect its own experiences and update its own policy network asynchronously.

In addition to A2C, there are several other actor-critic-based RL algorithms, such as Advantage Actor-Critic (A3C), Asynchronous Advantage Actor-Critic (A3C), and Proximal Policy Optimization (PPO). These algorithms have different variations in how the actor and critic networks are updated, but they all share the same fundamental idea of using the advantage function to reduce variance in policy gradients.

RL has been successfully applied to a wide range of applications, including robotics, gaming, and finance, among others. RL has shown remarkable performance in complex and dynamic environments, where traditional rule-based or heuristic methods may fail.

4. Software Architecture

The software architecture of our project comprises three major modules, namely the Navigation module, Obstacle detector module and Simultaneous Localization and Mapping (SLAM) module. The Navigation module is responsible for enabling the robot to move from an initial position to a set target position autonomously. To achieve autonomous navigation, we employed the use of Deep Reinforcement Learning (DRL) techniques.

DRL is a subfield of machine learning which involves learning what actions to take in order to maximize a reward signal. In our case, the reward signal was defined as the distance to the target position, with a negative reward being given for any collision with obstacles. The DRL algorithm used in our project was the Advantage Actor Critic (A2C) algorithm, which combines the benefits of both actor-critic and value-based methods to achieve more stable learning.

The Obstacle detector module uses the data from a LiDAR sensor to detect the presence of obstacles in the path of the robot. LiDAR, which stands for Light Detection and Ranging, is a remote sensing technology that uses laser light to measure distances and generate 3D maps of the environment. The obstacle detector module uses this 3D map to determine the presence of any obstacles that the robot might encounter during its path.

The SLAM module is responsible for constructing a map of the environment with the help of LiDAR and IMU's. SLAM is a technique used in robotics to simultaneously localize a robot within an environment while constructing a map of the environment. In our project, we used the LiDAR sensor and IMU (Inertial Measurement Unit) to construct a map of the environment that the robot navigates through.

Overall, our software design is structured in a way that enables the robot to navigate autonomously while avoiding obstacles and constructing a map of the environment in real-time. The use of DRL techniques, such as the A2C algorithm, ensures that the robot's navigation is adaptive and robust, while the use of LiDAR and IMU's in the SLAM module enables the robot to construct an accurate map of the environment in which it operates.

4.1 Go-to-Goal Algorithm

The Go-to-Goal Algorithm attempts to describe the problem as follows: "*For each time step 't' of every episode 'e', the agent at a state 's' executes an action 'a' {Left, Right, Forward, Backward…} so as to minimize the distance 'd' between the present pose 'p1' and the target pose 'p2', i.eE(p1, p2) where E(p) is the Euclidean distance function and maximizes the expected future sum of rewards."*

The **agent design** of our autonomous robot involves the use of Deep Reinforcement Learning (DRL) technique for autonomous navigation. At each time step of each episode, the agent receives state observations from the environment. These observations include map data, LiDAR data, and other relevant sensor data.

The policy network is a neural network that is responsible for selecting an action to be taken by the robot. It returns a probability distribution over actions based on the current state observation. This policy network is trained using the Advantage Actor-Critic (A2C) algorithm, which is a widely used DRL technique for continuous action spaces.

The critic network is another neural network that evaluates the quality of the actions taken by the robot. It takes as input the current state observation and the action taken by the robot, and outputs a value function. The value function represents the expected cumulative reward that the agent will receive from the current state.

The objective of the agent is to select actions that maximize the cumulative reward received over the course of an episode. The reward function is designed to encourage the robot to move closer to the target position and avoid obstacles. During training, the agent is trained on a local computer as the model is not yet trained, and the actions are sent to the robot via the internet. The robot serves as the environment during testing, where it is both the agent and the environment.

By utilizing DRL, our robot is capable of adapting to the dynamic and uncertain nature of the real world. The use of A2C allows the robot to learn from its past experiences and make better decisions for future actions, leading to more efficient navigation and obstacle avoidance.

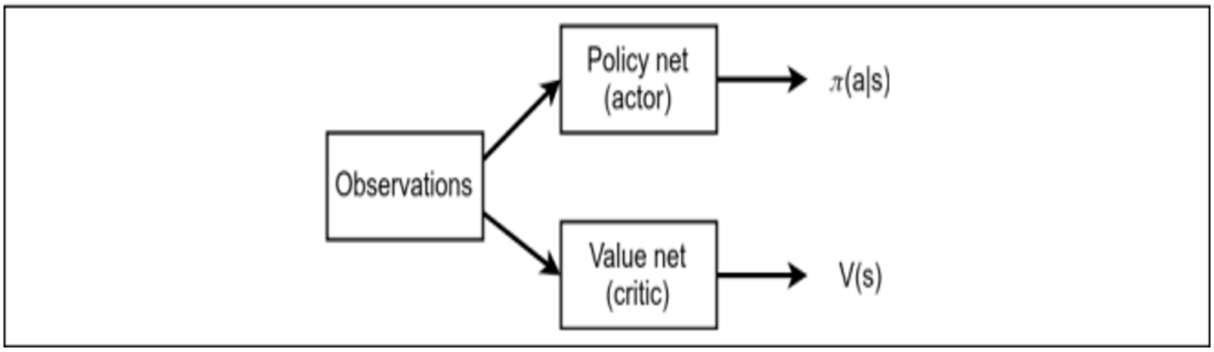


Fig1. Advantage Actor Critic Method

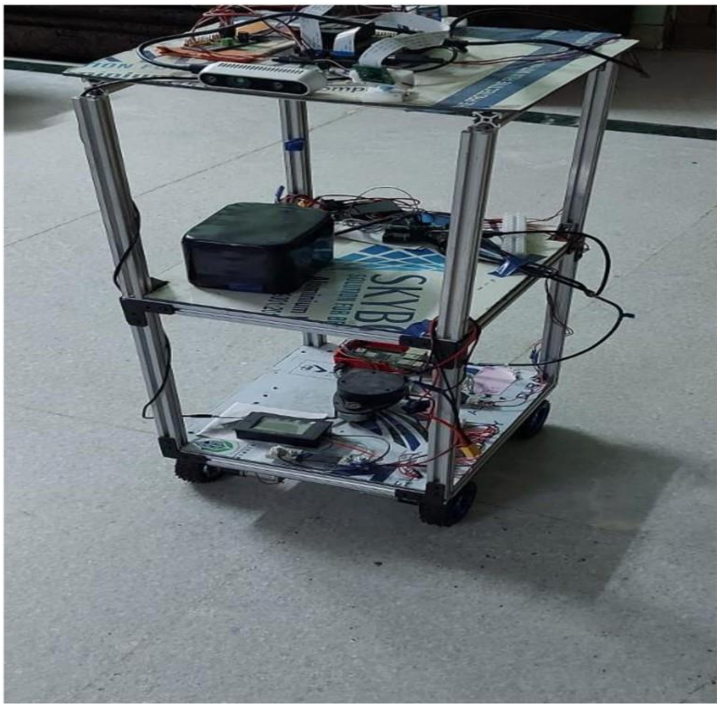
The **environment design** of our project is crucial in enabling the agent to learn and make informed decisions based on the received state observations and rewards. In our case, the environment is the wheeled robot itself, which receives actions from the agent at each time step and executes them. The environment then calculates the reward for the action and returns the updated state observations and reward back to the agent.

In order to design an effective environment, we incorporated several features that reflect the nature of the real world. One important feature is the obstacle detection module, which utilizes LiDAR sensor data to detect the presence of obstacles. This module plays a critical role in shaping the reward policy, as the robot receives a high negative reward/penalty for colliding with obstacles. This helps to ensure that the robot learns to avoid obstacles and navigate towards the target in a safe and efficient manner.

Another key feature of our environment design is the reward policy itself. We shaped the reward policy to be inversely proportional to the Euclidean distance between the robot position and the target position. This encourages the robot to move towards the target and rewards it for making progress towards the target. Additionally, a high bonus reward is given to the robot for reaching the target, which marks the end of the episode.

Overall, our environment design is a critical component of our project, as it provides the necessary feedback for the agent to learn and make informed decisions. By incorporating real-world features such as obstacle detection and shaping the reward policy to incentivize safe and efficient navigation towards the target, we are able to train our agent to navigate autonomously in a variety of environments.

4.2 Testing Robotic Platform



The testing robot platform has been designed to serve as a versatile tool for experimentation in a variety of AI applications. The robot has been equipped with a diverse range of sensors and actuators, which are categorized into three broad categories: navigation control, vision, and audio.

The navigation control category comprises sensors and actuators that allow the robot to navigate autonomously. These include sensors such as LiDAR and IMU, as well as motors and wheels that enable movement. The robot's navigation module utilizes data from these sensors to plan and execute a path towards a given target position. Additionally, the robot's obstacle detector module utilizes LiDAR data to detect obstacles and avoid collisions.

The vision category includes sensors such as cameras that provide visual data about the robot's surroundings. This data can be utilized for a range of applications, such as object detection, recognition, and tracking. Furthermore, the robot's design includes a camera stand that allows for the placement of additional cameras or sensors as required.

The audio category includes microphones that can capture audio data. This data can be utilized for various applications, such as sound localization and recognition. The testing robot's design enables the development and implementation of a wide range of AI algorithms, and its sensors and actuators facilitate experimentation and testing in various AI applications.

5. Mechanical Design

Our mechanical design for the 6-wheeled robot utilizes a rocker boogie mechanism to enhance the robot's stability without the need for complex suspension systems. The rocker boogie mechanism consists of two pairs of rocker arms, each attached to a wheel. The rocker arms pivot around a central point, allowing the wheels to remain in contact with the ground at all times, even when navigating over uneven terrain. This feature enhances the robot's stability and provides a robust platform for a range of applications.

The six wheels of the robot provide increased traction and enable it to maneuver through various obstacles with ease. Our design incorporates a differential drive system to enhance the robot's maneuverability and control. The differential drive system allows each wheel to operate independently, providing the robot with the ability to rotate on the spot and make precise movements. This feature is particularly useful in applications that require the robot to navigate through narrow or confined spaces.

Another significant feature of our design is the camera stand and telescope arm. The camera stand enables the mounting of cameras to capture video and images, while the telescope arm provides the robot with the ability to switch on or off any device. The telescope arm is particularly useful in applications where the robot needs to manipulate objects or turn on and off switches.

Our design prioritizes flexibility to accommodate a range of tasks and applications. The modular nature of the design enables the robot to be customized and tailored to specific tasks. This feature is particularly useful in applications where the robot needs to perform a variety of functions or navigate through different terrains.

In conclusion, our mechanical design for the 6-wheeled robot prioritizes stability, maneuverability, and adaptability. The incorporation of the rocker boogie mechanism, differential drive system, camera stand, and telescope arm offers an innovative solution to the challenges faced in developing a versatile and adaptable robot. Our design provides a robust platform for a wide range of applications in the field of robotics.

5.1. Components of Mechanical Design

The main components of our mechanical design for the 6-wheeled robot include the body, wheels, differential bar, rocker-bogie mechanism, and camera mount.

The body of the robot provides the structural framework and houses the electronic components necessary for its operation. The body also incorporates mounting points for the wheels, differential bar, rocker-bogie mechanism, and camera mount.

The six wheels of the robot provide increased traction and enable it to navigate through various obstacles with ease. The wheels are attached to the body using a differential bar that allows each wheel to operate independently. This feature enhances the robot's maneuverability and control, enabling it to rotate on the spot and make precise movements.

The rocker-bogie mechanism is another key component of our design. The rocker-bogie mechanism consists of two pairs of rocker arms, each attached to a wheel. The rocker arms pivot around a central point, allowing the wheels to remain in contact with the ground at all times, even when navigating over uneven terrain. This feature enhances the robot's stability and provides a robust platform for a range of applications.

The camera mount is another significant component of our design. The camera mount enables the mounting of cameras to capture video and images, providing the robot with a means of remote sensing. The camera mount is mounted on the body of the robot and can be adjusted to obtain the desired viewing angle.

The combination of the body, wheels, differential bar, rocker-bogie mechanism, and camera mount provides a flexible and adaptable platform for a range of applications in the field of robotics. The incorporation of these components into our design enhances the robot's stability, maneuverability, and adaptability, making it a useful tool for a variety of tasks.

5.2. Rocker-Boogie Mechanism

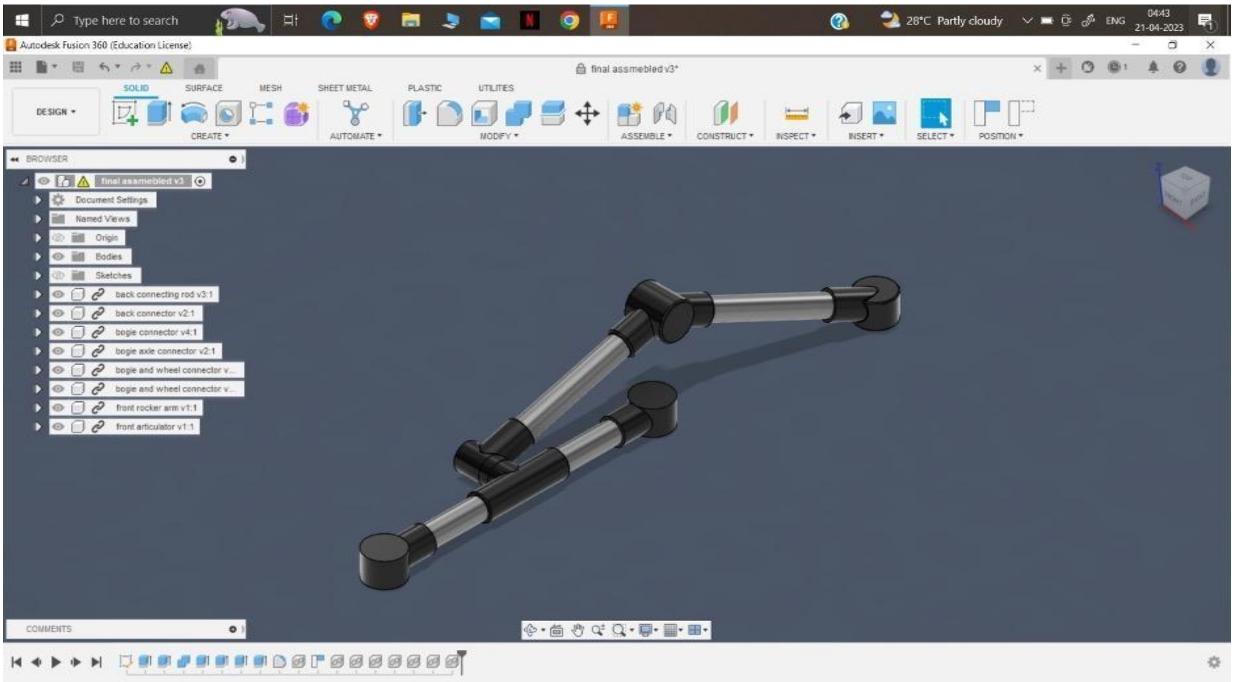
The rocker-bogie mechanism is a key component of our mechanical design for the 6-wheeled robot, providing enhanced stability and maneuverability. The suspension system works by allowing the rocker arms to pivot around the differential, which allows the wheels to maintain contact with the ground even when the terrain is uneven.

The rocker-bogie mechanism is made up of several sub-components, including the M-connector, front connector, front articulator, back connector, rocker and bogie connectors, and back articulator.

The M-connector serves as the central pivot point for the rocker-bogie mechanism, connecting the front and back connectors. The front connector is attached to the front of the robot's body and supports the front wheels, while the back connector is attached to the rear of the body and supports the rear wheels.

The front and back articulators are attached to the front and back connectors, respectively, and serve to pivot the rocker arms as the robot moves over uneven terrain. The rocker and bogie connectors connect the wheels to the rocker arms, allowing the wheels to move up and down independently of each other.

The combination of these sub-components results in a robust suspension system that provides enhanced stability and maneuverability, enabling the robot to navigate over various types of terrain with ease. The use of the rocker-bogie mechanism in our design enhances the robot's adaptability and flexibility, making it a useful tool for a range of applications in the field of robotics.



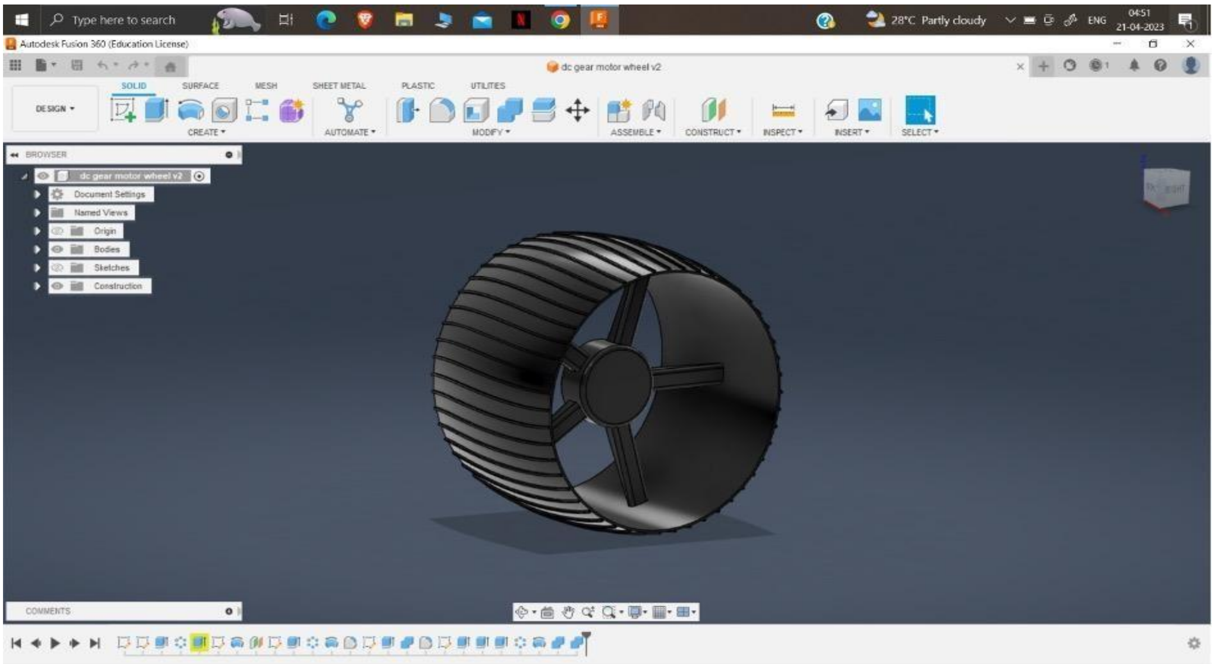
5.3. Wheels Design

In our mechanical design for the 6-wheeled robot, we have chosen a specific wheel design to improve the robot's mobility and durability. The wheels are narrower than traditional designs, which allows the weight of the robot to be distributed over a larger surface area, reducing the pressure on the wheels.

Our wheel design is based on a specific model that has been optimized for durability and mobility. The wheels are made from high-strength materials, and the tread pattern is designed to provide maximum traction in various types of terrain.

The design of the wheels also incorporates several features that enhance their performance. The wheels are designed to be lightweight, which reduces the overall weight of the robot and improves its maneuverability. The use of materials with high strength-to-weight ratios ensures that the wheels are strong and durable while remaining lightweight.

In addition, the wheels are designed to be easily replaceable, which reduces maintenance costs and downtime. The design of the wheels also allows for easy customization, enabling us to adapt the wheels to different types of terrain or specific applications.



5.4. Differential Bar

The differential bar is a crucial component of the rocker-bogie mechanism in our mechanical design for the 6-wheeled robot. The differential bar connects the two rocker arms on each side of the rover's suspension system, allowing the wheels to rotate independently of each other.

This feature is essential for maintaining stability and avoiding tipping over when the robot encounters uneven terrain or obstacles. When the vehicle encounters an obstacle, the wheels on one side of the vehicle may need to rotate at a different speed than the wheels on the other side to maintain stability and avoid tipping over. The differential bar allows the wheels to move independently, enabling the robot to navigate difficult terrain and maintain its stability.

The use of the differential bar in our design improves the robot's overall mobility and maneuverability, enabling it to traverse various types of terrain with ease. The differential bar is a critical component of the rocker-bogie mechanism in mobile robotic vehicles, allowing them to explore remote locations and navigate difficult environments.

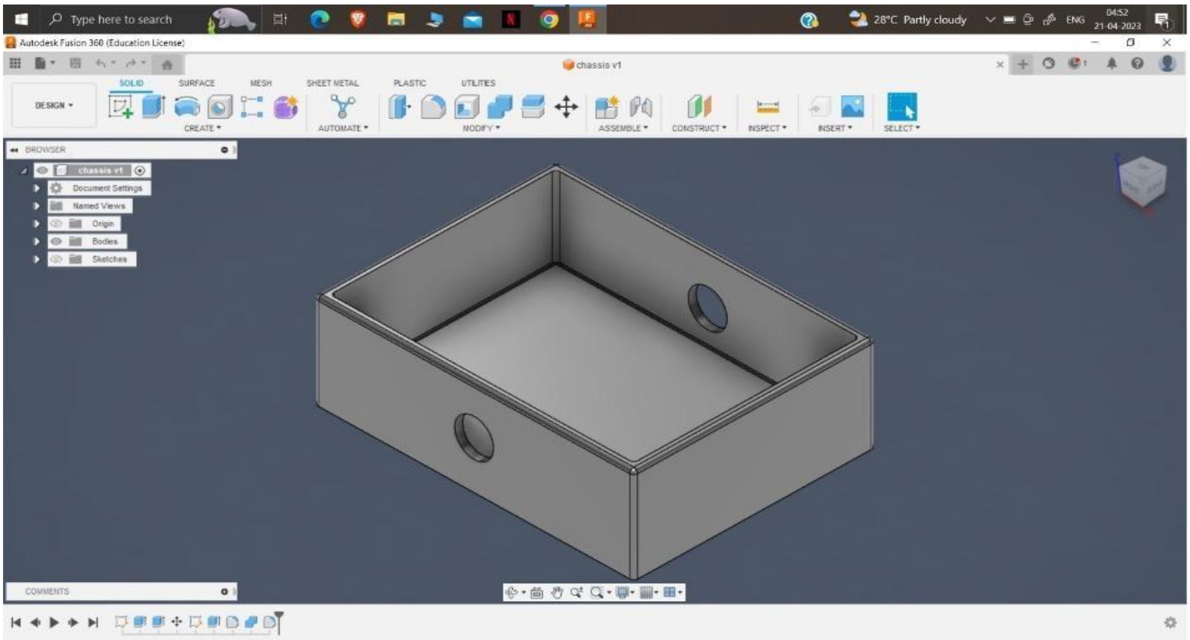
5.5. Chassis Design

The body or chassis of our mechanical design for the 6-wheeled robot serves as a structural support for the various components of the vehicle, including the suspension system, motors, batteries, and scientific instruments. It is designed to be lightweight yet strong enough to withstand the harsh conditions that the robot may encounter during its operations.

The body or chassis of the robot also provides a platform for the scientific instruments carried by the vehicle. These instruments are typically mounted on the top of the chassis and include a camera stand and telescopic arm, which can be used to switch on or off any components. The camera stand and telescopic arm are designed to be flexible, allowing the vehicle to take images from various angles and positions, enhancing its ability to observe and collect data.

Furthermore, the body or chassis provides protection for the sensitive electronics and instruments carried by the vehicle. It shields them from dust, radiation, and other hazards, ensuring that they continue to function properly in the harsh environment. The design of the body or chassis thus ensures the reliability and durability of the robot, making it a valuable tool for a wide range of scientific applications.

We have designed a CAD model of the body or chassis of the robot to ensure that it is lightweight and yet robust enough to withstand the rigors of the environment. Our design is intended to maximize the robot's performance while minimizing its weight, making it easy to transport and deploy in various locations.



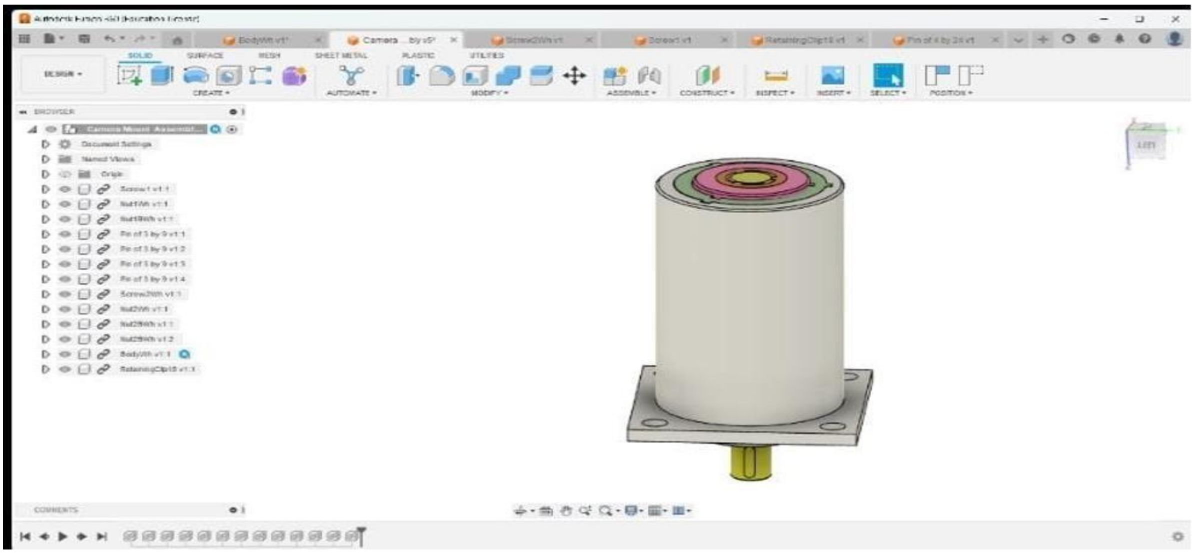
5.6. Camera Mount Design

The camera mount in our mechanical design for the 6-wheeled robot provides a stable platform for placing the camera on the robot. As robots rely on multi camera setups for several AI tasks, such as autonomous navigation, which require a 180^0 vision, the camera mount is an essential component of the design.

The camera mount is designed to ensure that the camera remains fixed in place with minimum vibrations. It consists of several components, including a pin, screw jacks, retaining clip, sliding and screw nuts, holding cap, and outer casing.

The pin serves as a pivot point for the camera mount, allowing it to rotate and adjust to different angles. The screw jacks are used to adjust the height and angle of the camera mount, ensuring that the camera is positioned correctly. The retaining clip is used to hold the camera in place, preventing it from moving or falling off the mount.

The sliding and screw nuts are used to adjust the position of the camera mount, allowing for fine-tuning of its position. The holding cap is used to secure the camera mount to the robot's chassis, ensuring that it remains stable during operation. Finally, the outer casing provides protection for the camera mount, shielding it from dust, moisture, and other hazards.



6. Challenges Faced

During the course of developing our autonomous robot, several challenges were encountered. One of the primary challenges was the need to implement the algorithm on a physical robot, as opposed to a simulated environment. This posed several difficulties, including dealing with physical constraints, such as limited battery life, mechanical wear and tear, and unpredictable environmental factors.

Another significant challenge was the occurrence of race conditions, where the agent's actions and the environment's state updates occurred concurrently, leading to synchronization issues. This resulted in inaccurate state observations and affected the overall performance of the robot.

Furthermore, ensuring proper synchronization of state observations between time steps proved to be a challenge. The agent requires accurate and up-to-date state information to make informed decisions, and any discrepancy or delay in receiving this information could lead to suboptimal performance.

Finally, the mechanical constraints of the real-world robot posed a significant challenge. The robot had limited mobility and was subject to constraints such as motor speed and turning radius, which affected its ability to navigate effectively in certain environments. Additionally, the robot's sensors and actuators had limited ranges, which required careful calibration and fine-tuning to ensure accurate and consistent performance.

7. Future Work

In the future, we plan to address some of the limitations and further enhance the capabilities of our robot.

One potential area of improvement is to fully implement the goal-to-goal algorithm. The goal-to-goal algorithm will allow the robot to navigate between multiple target positions autonomously. This will enable the robot to navigate through complex environments with multiple waypoints or targets.

Another potential area of future work is the integration of a camera module. This would enable the robot to perceive and process visual data, which can be used to improve the overall performance and safety of the robot. For instance, the robot can detect unauthorized personnel in the area and move towards them, alerting the authorities or performing other actions to ensure safety and security.

Furthermore, we intend to explore the concept of curiosity-driven exploration introduced by Professor D. Pathak from CMU. The curiosity-driven exploration algorithm can improve the exploration efficiency of the robot by encouraging it to explore the environment in a more diverse and informative way, which may lead to better performance and faster learning.

Finally, we plan to implement our software architecture on the 6 wheeled robot which we designed. This will enable us to test our algorithms and software design in a more realistic environment and evaluate the performance of our system in real-world scenarios. We anticipate that the integration of the software architecture with the 6 wheeled robot will enable us to further refine and optimize our algorithms, leading to better performance and increased reliability.

8. References

1. Ananye Agarwal, Ashish Kumar, Jitendra Malik, Deepak Pathak, “Legged Locomotion in Challenging Terrains using Egocentric Vision”, CoRL, 2022.

2. Deepak Pathak, Pulkit Agrawal, Alexei A. Efros, Trevor Darrell, “Curiosity-driven Exploration by Self-Supervised Prediction”, ICML, 2017.

3. K. Cao, C. Wei, A. Gaidon, N. Aréchiga, and T. Ma, “Learning imbalanced datasets with label-distribution-aware margin loss,” *CoRR*, 2019.

4. M. Jaderberg, V. Dalibard, S. Osindero, W. M. Czarnecki, J. Donahue, A. Razavi, O. Vinyals, T. Green, I. Dunning, K. Simonyan, *et al.*, “Population based training of neural networks,” *arXiv preprint arXiv:1711.09846*, 2017.