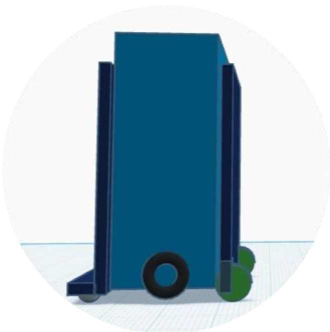
An-Najah National University

Faculty of Engineering & Information Technology

Presented in partial fulfillment of the requirements for

Bachelor degree in Computer Engineering

Hardware Graduation Project



SweepUp Stairs

Prepared By:

Jood Hamdallah

Saleh Sawalha

Supervised By:

Dr. Manar Qamhieh

January, 2025

Acknowledgment

We would like to express our sincere gratitude to all those who supported and guided us throughout the completion of this project.

First and foremost, we would like to thank our supervisor, Dr. Manar Qamhieh for her support, guidance, and invaluable advice.

We also extend our thanks to Eng. Abdullah Hinnawi for his assistance and valuable help during this project.

We are deeply grateful to An-Najah National University for providing the facilities that was crucial for completing this project. The supportive environment and resources greatly contributed to our work.

Finally, we would like to thank our families and friends for their unconditional support and encouragement. Their belief in us kept us motivated, and their help and presence made this journey more enjoyable and memorable.

Disclaimer

This report was written by Jood Hamdallah and Saleh Sawalha in fulfillment of the requirements for a Bachelor's degree in Computer Engineering. It has not been altered or corrected, other than editorial corrections, as a result of assessment and it may contain language as well as content errors. The views expressed in it together with any outcomes and recommendations are solely those of us. An-Najah National University accepts no responsibility or liability for the consequences of this report being used for a purpose other than the purpose for which it was commissioned.

Contents

List of Figures

Abstract

This project aims to create an autonomous stair-cleaning robot capable of detecting, climbing, and cleaning stairs efficiently. Cleaning stairs manually takes a lot of time and effort, especially in buildings with many floors. Most robotic vacuum cleaners are only made for flat surfaces and cannot clean stairs. This project solves that problem by creating a robot that can move up stairs while cleaning them efficiently.

The robot uses an OAK-D camera to detect stairs accurately and guide movement. It is controlled by Arduino and Raspberry Pi, which manage the motors, sensors, and cleaning functions. Smart navigation and detection sensors help the robot move smoothly and maintain balance. The cleaning system includes rotating brushes and dust collectors to remove dust and dirt from each step automatically, reducing the need for manual work.

Tests showed that the robot can detect stairs, climb them, and clean effectively. However, some challenges were found, such as high power consumption and heavy weight. Even with these issues, the robot achieved its main goal and proved that stair cleaning can be automated.

Future improvements could include making the robot go down stairs, improving battery life, and adapting it for outdoor stairs. This project shows how robotics can make cleaning easier, save time, and provide a practical solution for both homes and businesses.

Chapter 1

Introduction

1.1 Background

Stair cleaning is a challenging and time-consuming task, especially in homes, offices, and public places where stairs are common. Manual cleaning takes a lot of effort and can be very tiring, especially for stairs with many steps. Traditionally, cleaning stairs requires a broom or vacuum cleaner, and the person has to climb step by step while carrying the cleaning tool, which is both inconvenient and exhausting. Although cleaning technologies have improved, most machines are designed for flat surfaces and cannot handle stairs. This shows the need for a better and easier way of cleaning stairs.

1.2 Objectives

The goal of this project is to create a robotic stair sweeper that can climb stairs and clean them on its own. The ability to climb stairs is the most important feature because most cleaning machines today cannot handle stairs, leaving them as a challenging area to maintain. This robot will make stair cleaning easier and faster, especially for people who find it difficult to clean manually. This project shows how robots can perform difficult tasks and solve daily difficulties while providing a feasible solution for households and companies.

1.3 Significance

This project is important because it fills a gap in the market. Most robotic cleaners are made for flat floors, but none are designed to clean stairs, which are still hard and take a lot of time to clean. A robot that can climb stairs and clean them solves this problem. More people now want smart cleaning tools to save time and effort, and businesses also need easy ways to keep busy areas like stairs clean. This project offers a simple and useful solution for homes and businesses and could become a valuable tool in a growing market.

1.4 Organization of the Report

This report is structured to clearly present the design, development, and testing of the stair-cleaning robot. It consists of the following sections:

- **Introduction** – Provides an overview of the project, highlighting the need for a stair-cleaning robot and the limitations of existing cleaning methods.
- **Literature Review** – Reviews existing cleaning robots, identifying the lack of stair-cleaning solutions and justifying the project.
- **Methodology** – Explains the robot's design, components (Arduino, Raspberry Pi, OAK-D camera), movement, and cleaning mechanisms.
- **Results, Analysis, and Discussion** – Evaluates the robot's stair detection, climbing, and cleaning performance, noting strengths and areas for improvement.
- **Conclusion and Recommendations** – Summarizes achievements and suggests future improvements.

Chapter 2

Literature Review

Robotic cleaning has improved a lot in recent years, with automation, smart navigation, and AI object detection making robots more efficient. However, most cleaning robots are designed for flat surfaces, and few can climb stairs. Stair-climbing is difficult because it requires stable movement, height adjustment, and obstacle detection. Recent advancements in AI, depth sensing, and modular movement have helped develop robots that can move up stairs while cleaning.

The Migo Ascender [1] was one of the first vacuum robots that could climb stairs. It used smart sensors and a mechanical climbing system to move between floors. However, it was limited to vacuuming and could not handle mopping, dust removal, or different stair types.

A study published in MDPI [2] discusses how modern cleaning robots use LiDAR, ultrasonic sensors, and AI to detect obstacles, create 3D maps, and plan cleaning paths. These technologies have improved cleaning on flat surfaces, but most commercial robots still cannot climb stairs.

One advanced stair-cleaning robot is sTetro, introduced in MDPI [3]. It uses AI-powered stair detection, LiDAR sensors, and a conveyor-based climbing system to move up and down stairs. It can detect dirt, recognize obstacles, and adjust its cleaning method. The automated movement system helps it climb efficiently, making it better than regular robotic vacuums.

While sTetro uses a conveyor system, our project takes a different approach. Instead of conveyors, our robot moves in all directions using omni wheels and stepper motors. It detects stairs using an AI-powered OAK-D camera and avoids obstacles using IR sensors and limit switches. For cleaning, it uses rotating brushes and dust collectors to remove dust and dirt.

By combining smart stair detection, flexible movement, and efficient cleaning, this project aims to create a cost-effective, fully autonomous stair-cleaning robot.

Chapter 3

Methodology

The system structure, hardware components, the overall system design, and mobile app will shown in detail in this chapter.

3.1 System Structure

The body of the robot consists of three wooden cuboids: a large central cuboid and two smaller ones on each side. The central cuboid is connected to the side cuboids (the first and third) using lead screw nuts. This connection allows the side cuboids to be raised or lowered as needed. Figure 3.1 shows the project's design in, Tinkercad, providing a clear visualization of the robot's frame and structure.

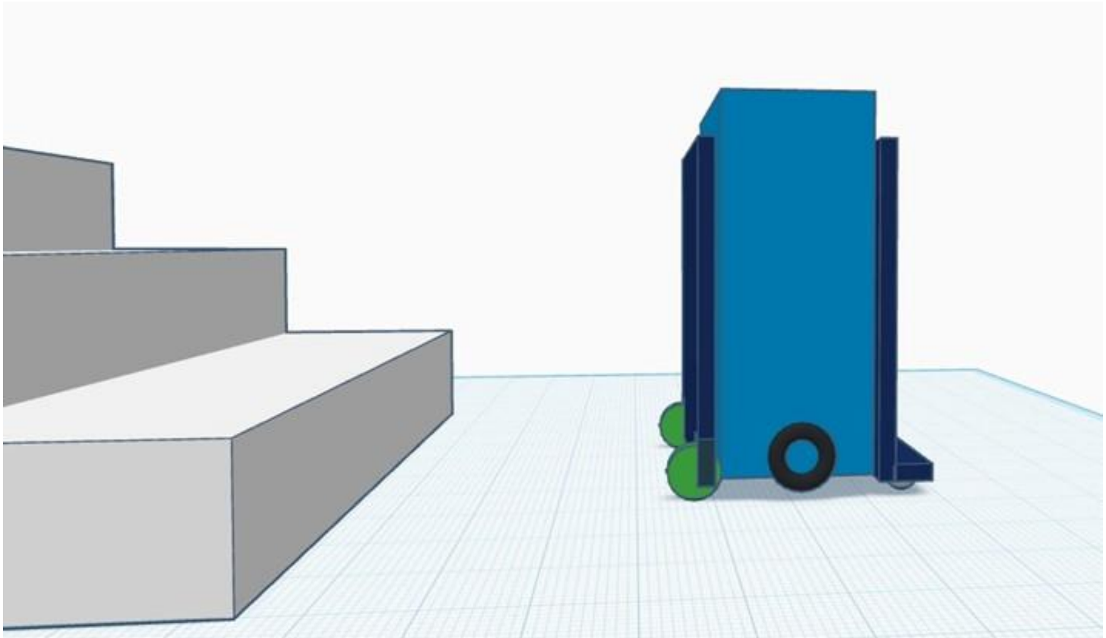


Figure 3.1: Robot Body Structure.

3.1.1 Key Components of the Robot

3.1.1.1 Front Side Cuboid:

This is the front part of the robot that faces the stairs. It includes several components essential for detecting and interacting with the stairs including: limit switches, IR sensors, ultrasonic sensors, two brushes for cleaning, and two omni wheels attached with two DC gear motors controlled by L298N bridge drivers. 3.2.

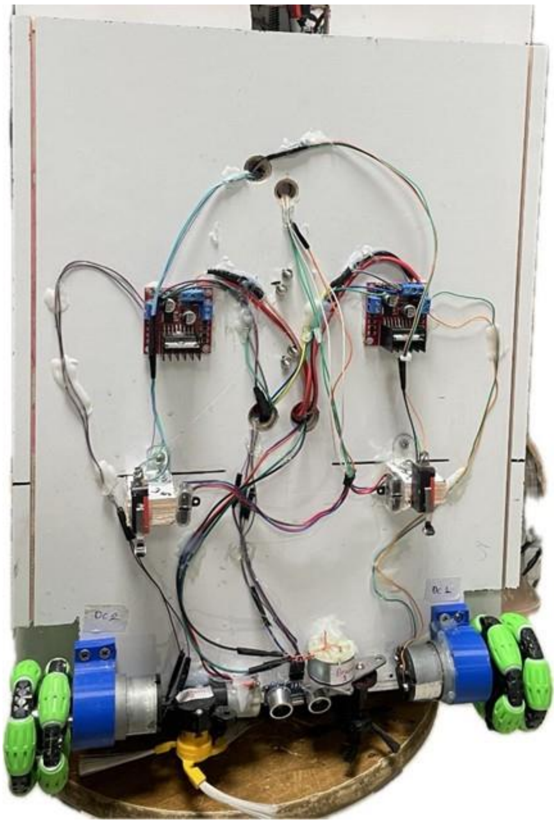


Figure 3.2: Front Side Cuboid of the Robot.

3.1.1.2 Central Cuboid (Main Body)

The central part is the most important section of the robot, as it handles all the main operations. It includes:

1. **Two Lead Screws:** One on each side, powered by NEMA 23 stepper motors, for raising and lowering side parts.
2. **Four Omni Wheels Motors:** Driven by NEMA 23 stepper motors, two motors for movement in the X-direction and two for movement in the Y-direction, ensuring smooth and controlled navigation.
3. **Four TB6600 Motor Drivers:** Each stepper motor is controlled by its corresponding TB6600 driver, ensuring smooth and accurate operation.
4. **Two Swivel Ball Caster Rollers:** Provide stability and steadiness during operation, especially

when the robot is climbing stairs.

5. **Vacuum Fan:** Located on each horizontal side to perform effective cleaning.
6. **Arduino Controller:** The main control unit that manages all movements and operations of the robot. See figure 3.3.

Figure 3.3: Central Cuboid of the Robot.

3.1.1.3 Back side cuboid:

This is the back part of the robot. It has two swivel rollers that help with smooth movement and keep the robot stable while it operates

3.1.2 Robot Operation Process

The robot's process involves detecting, climbing, and cleaning stairs autonomously. Using a combination of a camera, IR sensors, ultrasonic sensor, limit switches, and motors to move step by step, making sure it is in the right position and cleans properly. It can work on both regular and flat stairs, adjusting its actions as needed for smooth and efficient cleaning. The next sections explain the steps of this process in detail.

3.1.2.1 Initialization and Stair Detection

- The process begins with the Python code running on the Raspberry Pi 4. The OAK-D camera, mounted above the robot, starts executing its task.

- The camera scans the area for stairs. Once stairs are detected, the IR sensors check for objects within their range.
- If the IR sensors detect an object (indicating the stairs are nearby), the robot moves forward toward the stairs using the omni wheels driven by stepper motors.

This figure 3.4 shows a live output of the OAK-D camera detecting a staircase with a high confidence level of 92%, as part of the robot's stair detection process.

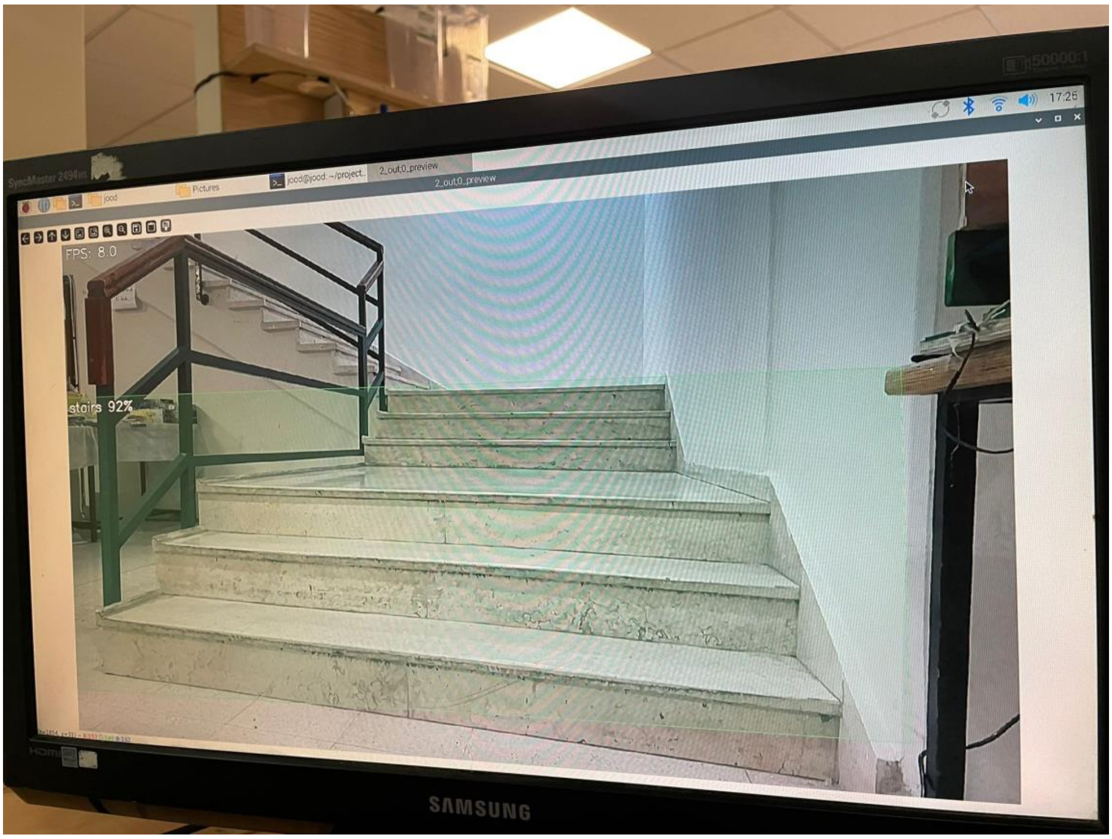


Figure 3.4: Stair Detection Output from the OAK-D Camera.

3.1.2.2 Front Part Movement

- When the robot reaches the stairs and the limit switches on the front cuboid are pressed, the front part is raised.
- The front cuboid continues to rise until it hits the maximum limit (20 cm), as determined by the limit switch. There's also ultrasonic used if the step height varies.

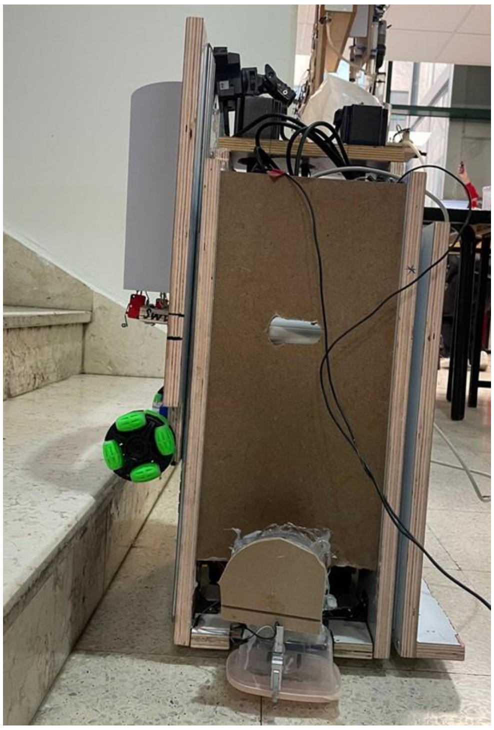


Figure 3.5: Raising the Front Part for Stair Climbing.

3.1.2.3 Central (main) Part Movement

- After the front cuboid is raised, the central part moves forward in the Y-direction. It travels a distance equal to the width of the front part (approximately 9 cm) to ensure the front cuboid is properly positioned on the tread of the step.
- The IR sensor at the bottom of the front cuboid confirms contact with the step. This ensures the front part is securely on the stair tread.

The figure shown below 3.6 illustrates the state after the central part has completed its forward movement, ensuring that the front cuboid is positioned securely on the stair tread.

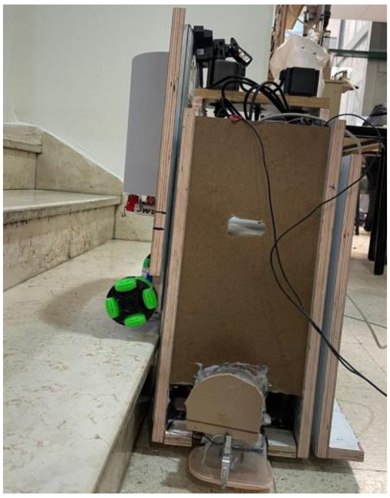


Figure 3.6: Position of Front Cuboid on Stair Tread After Moving Central Part.

3.1.2.4 Lifting the Central Cuboid

- Once the front cuboid is confirmed to be on the step, the central cuboid begins to rise.
- The central cuboid lifts until it reaches the same level as the front cuboid and the stair tread.
- After reaching the same level, the DC motors move the front small green wheels forward until both limit switches are pressed. At this point, both the front and central cuboids are securely positioned on the step.

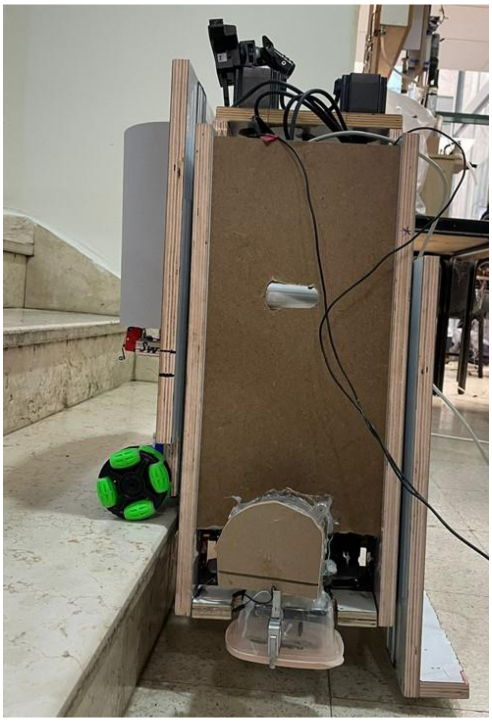


Figure 3.7: Lifting the Central Cuboid- In process

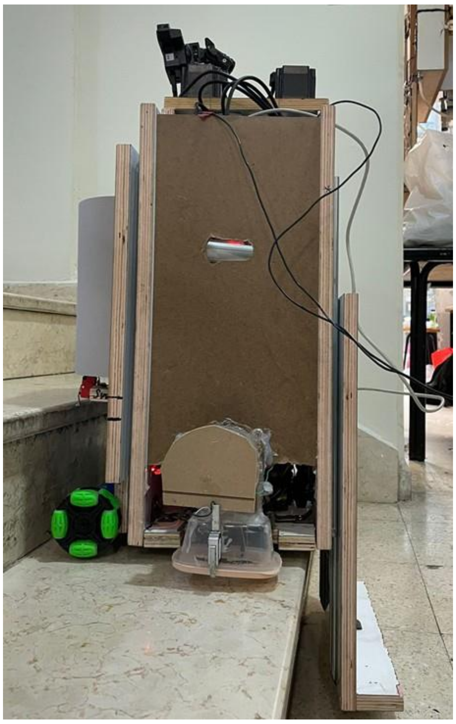


Figure 3.8: Central Cuboid on Stair Tread

3.1.2.5 Cleaning

- With both the front and central parts securely positioned on the step, the robot activates its vacuum fans and brushes.
- The robot moves right and left across the step, ensuring thorough sweeping and cleaning of the entire surface.

Figure 3.9: Robot Moving Left Across the Step During Cleaning

Figure 3.10: Robot Moving Right Across the Step During Cleaning

3.1.2.6 Back Part Movement

- Once cleaning is complete, the back cuboid is raised until it aligns with the other parts at the same level.

Figure 3.11: Back Cuboid Raised After Step Cleaning Completion.

3.1.2.7 Repeating the Process

- The robot continues this process step by step:
  1. Detecting the next stair.
  2. Raising the front part.
  3. Moving the central part forward.

4. Lifting the back part.
5. Cleaning the step before moving to the next one.

- During this process, the IR sensors continuously check for objects, and the camera continues to detect stairs to guide the robot.

3.2 Robot Features and Implementation Details

3.2.1 Stair Detection Using Camera

The stair detection system is powered by a Raspberry Pi 4 model B and an OAK-D camera, which work together to identify stairs. The camera uses a Python script that incorporates a trained model from the Roboflow Universe, specifically designed for stair detection.
Here's how the detection process works:

1. **Detection Process:** The OAK-D camera captures visual data and processes it using the trained model.
2. **Communication with Arduino via Serial:** When stairs are successfully detected, the Raspberry Pi communicates with the Arduino via a serial connection. It sends a specific character to signal the Arduino to execute the stair-climbing code.
3. **No Detection:** If the camera does not detect stairs or the confidence level is below the threshold, the Raspberry Pi sends a ”no detection” signal over the serial connection to inform the Arduino that no stairs are present.

3.2.2 Lifting Mechanism

The lifting mechanism allows the robot to raise and lower the side cuboids as needed. It uses lead screws and nuts connected to angle brackets.See figures 3.12 and 3.13. The stepper motor is connected to the lead screw through a coupler as shown in figure 3.17. When the motor rotates, it drives the lead screw to turn in the corresponding direction. This rotation causes the lead screw nut, which is threaded onto the screw, to move either upward or downward along the screw threads, depending on the direction of the motor's rotation. This movement adjusts the height of the side cuboids, enabling precise raising or lowering as needed.

Limit switches are used to ensure the movement is precise. When a cuboid reaches a limit switch, the mechanism stops to prevent it from going too far. The number of steps needed to raise and lower the parts was counted and programmed, ensuring consistent and accurate movement. This mechanism is critical for the robot to climb stairs smoothly and safely.

For the central part, when the stepper motors rotate downward, the lead nuts move down. However, since the side parts are fixed to the ground, this causes the lead screws to move upward, effectively lifting the central cuboid.

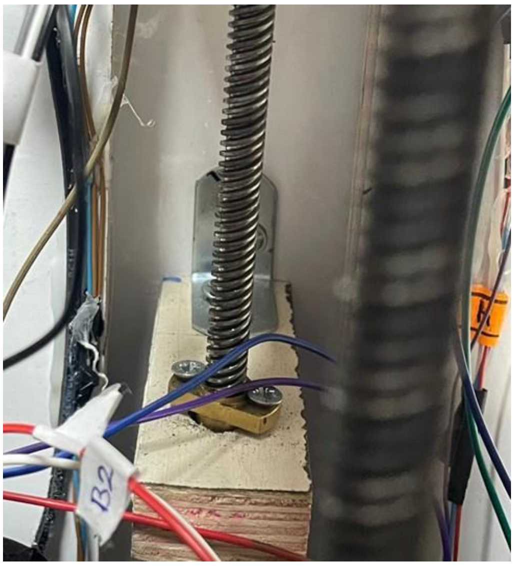


Figure 3.12: Close-up of Lead Screw and Nut Mechanism.

Figure 3.13: Angle Bracket Connection for Lead Screw.

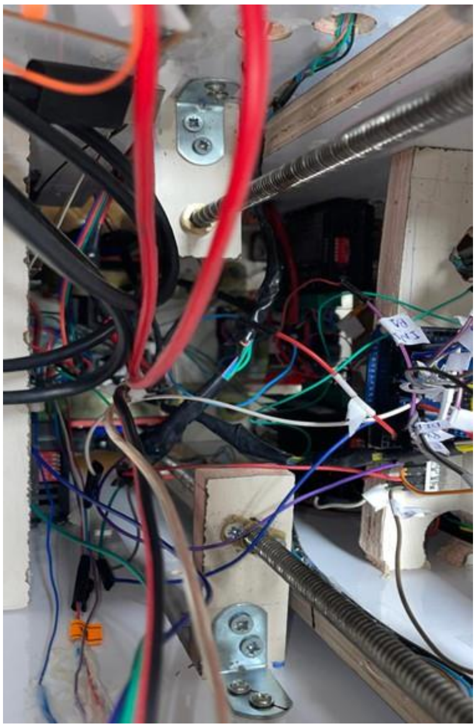


Figure 3.14: Two Lead Screws on Either Side of the Main Part.

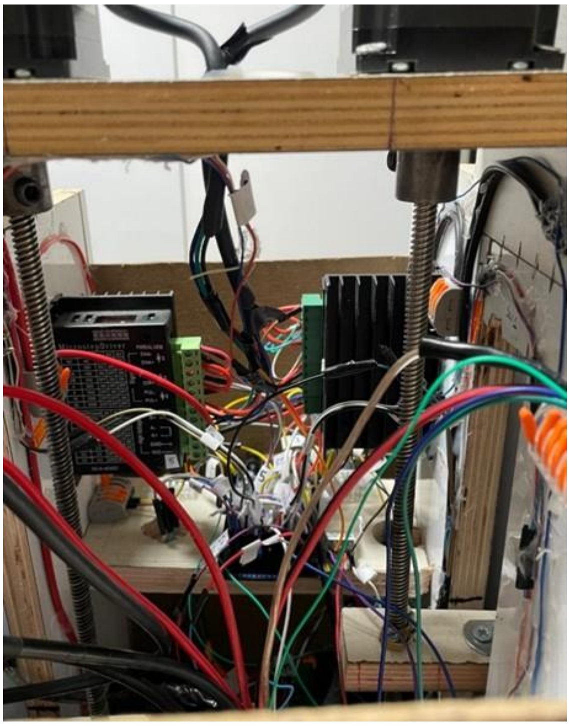


Figure 3.15: Motors Attached with Coupler and Lead Screws.

As shown in the figures below, the slider mechanism is mounted on the front and back of the central cuboid, connecting it to the front and back cuboids. These sliders ensure smooth and stable movement when raising or lowering the cuboids, keeping them aligned and steady. This helps the robot operate more efficiently and stay balanced, especially while climbing stairs.

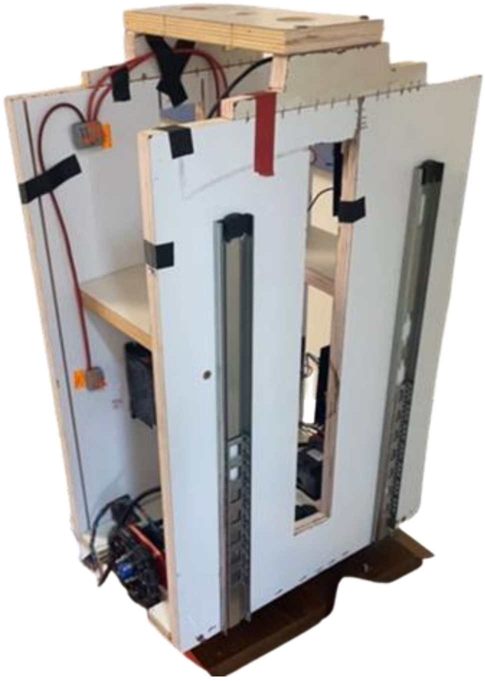


Figure 3.16: Central Cuboid with Slider Mechanisms.

Figure 3.17: Slider Mechanism Alignment in the Side Cuboid.

3.2.3 Movement (X and Y Direction):

The robot moves in both the X and Y directions using omni wheels. These wheels have small rollers around the edges that can spin sideways, making it easy for the robot to move smoothly in any direction. Omni wheels can provide strong forward and backward motion while also sliding sideways with ease.

The movement is powered by NEMA 23 stepper motors with a torque of 1.3 N.m. These motors are connected to the omni wheels using 3D-printed couplers. Stepper motors were chosen because they are very precise, and microstepping makes their movement even more accurate.

When the robot moves right and left for cleaning, limit switches on each dust collector (on both sides of the main part) control when the movement stops. This setup ensures the robot moves smoothly and accurately while cleaning.

Below are the wheels, which make up the base of the central (main) cuboid. This setup helps the robot move easily and accurately.

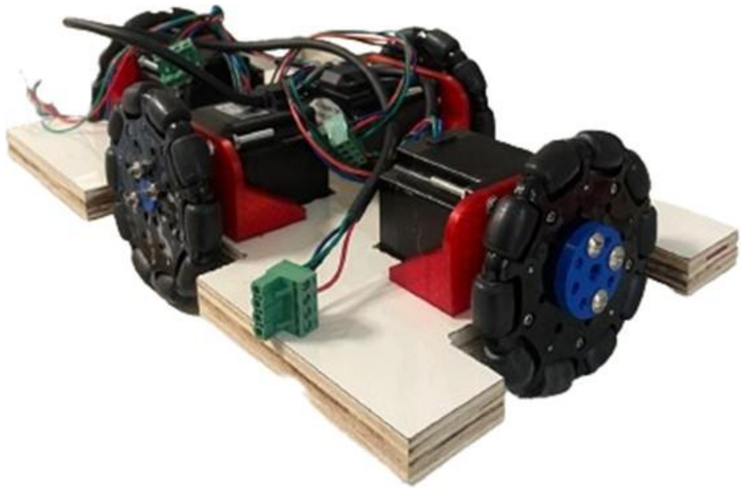


Figure 3.18: Omni Wheel Base Supporting Main Part Movement.

3.2.4 Cleaning

Cleaning is performed using a combination of sweeping brushes and dust collector fans:

- **Dust Collector Fans:** There are two dust collector fans (DC motors) located on each horizontal side of the main part. These fans are controlled by an H-bridge. When the robot moves right and left to clean each step, the fans turn on, sucking up dust and storing it in small containers.
- **Sweeping Brushes:** Two brushes are located in the front part of the robot. These brushes are controlled by relays and sweep dust toward the main part, where it is sucked up by the dust collector fans.

Figure 3.19: Dust Collector Fan One.

Figure 3.20: Dust Collector Fan Two.

Figure 3.21: Sweeping Brush One.

Figure 3.22: Sweeping Brush Two.

3.3 Hardware components

3.3.1 Microcontrollers

Arduino Mega 2560

The Arduino Mega 2560 is the main controller of the project. It receives signals from the Raspberry Pi and sensors, processes the data, and controls motors, switches, and relays. It ensures the robot moves correctly based on staircase detection, making real-time decisions for smooth operation.

Figure 3.23: Arduino Mega2560

Raspberry Pi 4 Model B

Runs the Python code for the OAK-D camera, handling real-time image processing to detect stairs and send the detection results to the Arduino Mega for movement control.

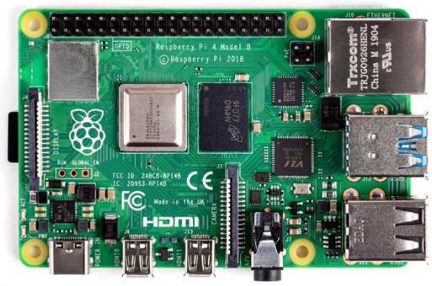


Figure 3.24: Raspberry Pi 4 Model B

3.3.2 Motors and Drivers

3.3.2.1 Stepper Motors

- **Six NEMA 23 Stepper Motors:** Four motors drive the omni wheels for the robot's movement, while two control the lead screw mechanism.

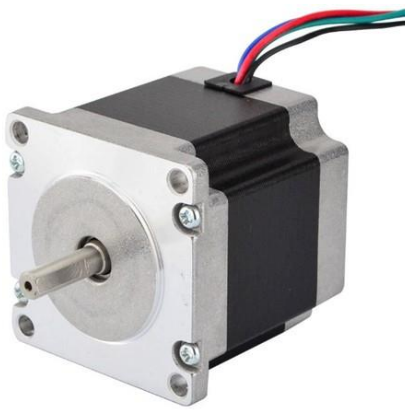


Figure 3.25: NEMA23 Stepper Motor

- **Six Stepper Motor Drivers (TB6600):** Provide the necessary control and power to the stepper motors.

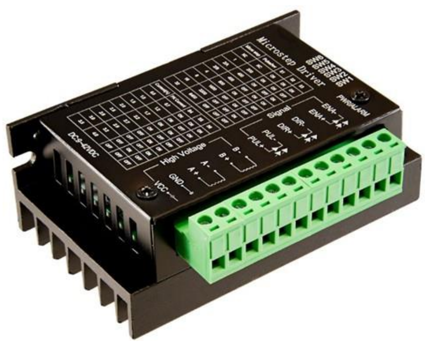


Figure 3.26: TB6600 Stepper Motor Driver

3.3.2.2 DC Motors

- **Six 12V DC gear motors:** two drive the front omni wheels for precise movement, two operate the brushes, and two power the dust collector fans.

Figure 3.27: DC Gear Motor

- **Dual H-Bridge L298 motor driver:** used to control DC motors, regulating speed via PWM and managing direction.

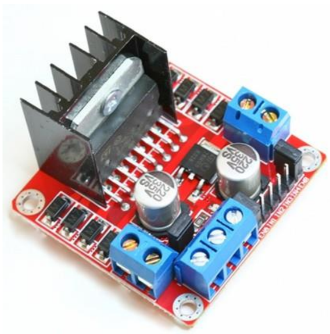


Figure 3.28: L298N Motor Driver

3.3.3 Sensors

Ultrasonic Sensor: used to detect variations in step height for adaptive navigation.

Figure 3.29: Ultrasonic Distance Sensor-(HC-SR04)

IR Sensor: used to detect nearby objects and confirm contact with the stair tread for precise navigation

Figure 3.30: Sharp ir sensor

Limit Switches: stop movement at the right points to prevent over-extension and control cleaning range.

Figure 3.31: Limit Switch

3.3.4 Other Components

3.3.4.1 Omni wheels:

allow the robot to move smoothly in both X and Y directions. Their unique design with small rollers enables movement in any direction, including sideways and diagonal.

Figure 3.32: Omni Wheel

3.3.4.2 Relays:

Two 2-channel relays: one controls the on/off switching of the brush DC motors, while the other manages the LEDs that indicate the direction of movement.

Figure 3.33: 2 Channel 5V Relay Module

3.3.4.3 OAK-D Camera:

The OAK-D camera is an advanced AI-powered depth camera designed for real-time object detection and depth perception, making it ideal for applications like stair detection in robotics projects. It combines a high-resolution 12MP RGB camera with two monochrome global shutter cameras to achieve accurate stereo vision.It's used for stairs detection in the project.[4]

Figure 3.34: OAK-D Camera

3.3.4.4 Main Switch

Acts as the primary power control, allowing the user to turn the robot on and off.

Figure 3.35: On/Off Switch

3.4 Constraints

3.4.1 Cost Limitations

- **High-Cost Omni Wheels:** Finding omni wheels locally was difficult, so we had to buy them online at a much higher price. This made budget management a challenge.

3.4.2 High Power Consumption

- **High Energy Consumption of Stepper Motors:** Stepper motors consumed a lot of power, causing the battery to drain quickly. This limited the robot's working time and required frequent recharging. A bigger battery was not an option due to space and weight constraints.

3.4.3 Size Limitations and Crowded Space

- **Space Constraints:** The robot had to be small due to the dimensions of the step stairs. This made the internal layout very crowded with wires and components, making it difficult to assemble, repair, and maintain.

3.4.4 Heavy Weight and Durability Issues

- **Structural and Weight Challenges:** The robot's heavy motors and structure caused stress on the 3D-printed couplers, leading to frequent breakages. This resulted in constant repairs and the need for more durable parts.

3.5 Mobile Application

The mobile application is designed to allow users to control and monitor the robot's functions with ease. It provides a user-friendly interface where users can:

- **Turn the Robot On or Off:** Simple buttons to start or stop the robot's operation.
- **Monitor Battery Status:** Displays the current battery percentage of the robot in real-time.
- **Adjust Speed:** Users can set the robot's speed using a slider.
- **Select Cleaning Mode:** Options to choose between different cleaning modes, such as "Full" or "Stairs."
- **View Logs:** A log section displays recent actions and changes made, such as speed adjustments and cleaning mode selections, with timestamps.

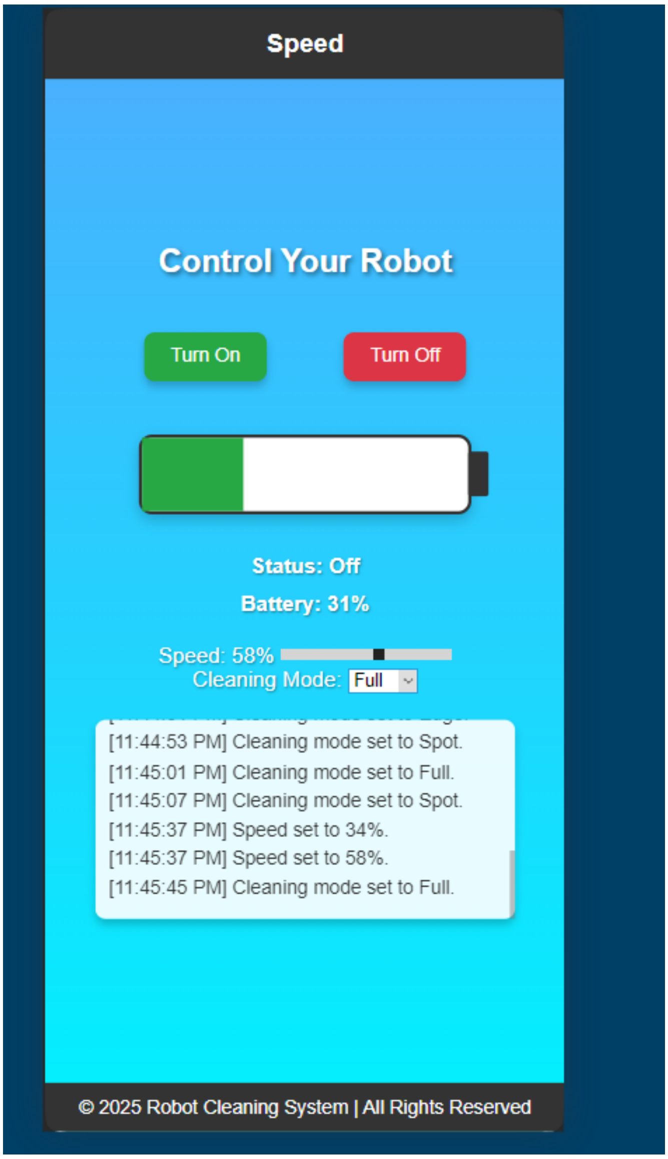


Figure 3.36: Mobile Application Interface

This app ensures full control and real-time updates for efficient operation of the robot.

Chapter 4

Results and Discussion

4.1 Data Collected

The robot was tested under different conditions to evaluate stair detection accuracy, climbing performance, and cleaning efficiency. The collected data was analyzed to identify key strengths and limitations.

4.1.1 Stair Detection Performance

The OAK-D camera was tested on various types of stairs with different heights, materials, and lighting conditions. The results showed:

- The camera successfully detected stairs with high accuracy across multiple trials.
- It maintained reliable performance in different lighting conditions.
- Errors were minimal but could occur when stair edges had low contrast or uneven surfaces.

4.1.2 IR Sensors and Limit Switches

The IR sensors and limit switches were essential for ensuring accurate movement and positioning.

- The IR sensors helped detect objects and confirm the robot's position on the stairs.
- Limit switches played a critical role in controlling movement, ensuring that the robot stopped at the correct positions and preventing overextension. They also helped regulate side-to-side cleaning motion.

4.1.3 Cleaning Efficiency

To evaluate dust collection, different types of dust and debris were placed on the test surface. The results showed:

- The dust collector fans effectively gathered and stored dust in containers.
- Lighter particles were collected more efficiently, while heavier debris required stronger suction.
- The brushes successfully swept dust toward the collector, ensuring thorough cleaning.

4.1.4 Robot Movement and Climbing Performance

The robot's movement and stair-climbing ability were tested to ensure smooth operation and precise positioning:

- The lead screw mechanism raised and lowered the cuboids as expected.
- Stepper motors drove the omni wheels effectively, allowing controlled movement in both X and Y directions.
- The robot's weight affected performance, requiring reinforced 3D-printed parts to handle stress.

4.1.5 Error Estimation, Challenges, and Solutions

The goal of this project was to build a stable and accurate stair-climbing robot that can efficiently detect stairs, move smoothly, and clean effectively. The results indicate that the robot met these goals, but during development and testing, several challenges were encountered that required problem-solving.

1. **Size Constraints and Crowded Space**
   - The robot's small design, limited by stair dimensions, resulted in a very compact internal layout. This made wiring, assembly, and maintenance difficult.
   - **Solution:** We tried to implement effective cable management to optimize space and ensure easier access for maintenance.
2. **Weight and Durability Issues**
   - The robot's heavy structure put stress on 3D-printed couplers, causing frequent breakages.
   - **Solution:** New couplers were purchased often, and broken components were temporarily repaired using strong AB adhesive materials.
3. **Dust Collector Fan Interference**
   - When the fan operated at full speed, it generated noise that interfered with the limit switch readings, sometimes causing false detections.
   - **Solution:** Instead of using a relay, the fan was controlled using an H-Bridge with PWM control, which provided better speed regulation and reduced interference.
4. **Voltage Drops and Unstable Power Supply**
   - The relay used for the fan sometimes caused voltage drops in the Arduino, leading to random execution of code and unexpected behavior.
   - **Solution:** Replacing the relay with an H-Bridge improved stability.
   - Another power issue involved providing power to the Raspberry Pi and OAK-D camera. Since the camera uses AI and neural networks, it required high power consumption. A power bank was used to provide 5V and 3A, but it did not always deliver a stable 5V output, causing voltage drops, especially when the camera was processing images.
   - **Solution:** Reducing the camera's frame size lowered power consumption. In some cases, the camera was powered separately using its own charger to ensure stable performance.
5. **Inaccurate Right-to-Left Rotations for Cleaning**

   - Initially, the robot moved right until the limit switch was pressed, and then it returned to the left for the same number of rotations. However, this method was not accurate, as the robot often moved back fewer rotations than required.
   - **Solution:** We added a limit switch on the left side along with an IR sensor, making the movement more accurate and reliable.

6. **Unreliable Ultrasonic Sensor Performance**
   - Initially, we used an ultrasonic sensor to determine when the robot should stop. However, the sensor's performance was inconsistent and often inaccurate.
   - **Solution:** We replaced the ultrasonic sensor with limit switches, which provided more precise and dependable control.

7. **Code Execution and Processing Delays**
   - While the robot was moving, it needed to continuously check sensors and switches while driving the stepper motors. Initially, delays in the code caused blocking, preventing the Arduino's CPU from handling multiple tasks efficiently.
   - **Solution:** Replacing delay functions with `millis()` and `delayMicroseconds()` allowed non-blocking execution and smoother operation.

Key Contributions

This project introduces an innovative approach to autonomous stair cleaning by developing a robotic stair sweeper capable of detecting, climbing, and cleaning stairs efficiently. The key contributions of this work include:

1. **Autonomous Stair Cleaning Capability:** Unlike traditional robotic cleaners designed for flat surfaces, this system integrates stair-climbing and cleaning functionalities, addressing a major gap in the market.
2. **Advanced Stair Detection and Navigation:** Utilizes an OAK-D AI-powered camera and IR sensors to accurately detect stairs and ensure precise positioning, making navigation highly reliable in various environments.
3. **Innovative Lifting Mechanism:** Implements a lead screw-driven lifting system that allows the robot to adjust its height dynamically, ensuring smooth stair climbing and descent.
4. **Multi-Directional Mobility Using Omni Wheels:** Integrates NEMA 23 stepper motors and omni wheels, enabling smooth and controlled movement in both X and Y directions, enhancing movement on stairs.
5. **Efficient Dust Collection System:** Features a dual cleaning mechanism combining rotating brushes and dust collector fans for thorough debris removal from each stair step.

Chapter 5

Conclusion and Future work

5.1 Conclusion

This project successfully developed a stair-cleaning robot that can detect, climb, and clean stairs autonomously. Unlike most cleaning robots designed for flat floors, this robot can move up stairs while cleaning, making it a practical solution for homes, offices, and public spaces.

The robot uses an AI-powered camera (OAK-D) to detect stairs, omni wheels for smooth movement, and a lifting system to climb steps. It cleans using rotating brushes and dust collectors to remove dust and debris. Sensors and limit switches ensure accurate movement and proper cleaning of each step.

Testing showed that the robot effectively detects stairs, climbs them, and cleans well. However, some challenges were identified, including high power consumption and heavy weight. While some improvements were made, further refinements can make the robot even better.

This project was a significant learning experience, allowing us to apply the knowledge and skills we gained throughout our five years of studying computer engineering. From identifying a real-world problem to designing and implementing a functional solution, we put our expertise into action. The development of the stair-cleaning robot required us to integrate hardware and software components, work with sensors and motors, and develop efficient control systems. We also improved our problem-solving skills, as we encountered challenges along the way and found creative solutions to overcome them.

Through this process, we learned how to design and build a complete system from scratch, ensuring that it could detect stairs, navigate efficiently, and perform cleaning tasks effectively. We also gained a deeper understanding of hardware-software integration, particularly in sensor-based automation and motor control. Beyond the technical aspects, this project helped us improve our ability to analyze problems, think critically, and innovate practical solutions that can make everyday tasks easier for people.

5.2 Recommendations for Improvement

Although the stair-cleaning robot successfully achieved its main objectives, several areas for improvement were identified. Implementing these enhancements would increase its efficiency, reliability, and ease of use while maintaining a cost-effective design.

1. **Weight Reduction and Material Optimization**
    - **Using Lighter Materials:** The wooden frame makes the robot heavy, which affects its movement and power consumption. Replacing it with lighter materials like aluminum, carbon fiber, or strong plastic will make the robot more efficient and easier to handle.
2. **Improved Power Efficiency**
    - **Optimizing Battery Consumption:** The high power demand of six stepper motors causes the battery to drain quickly. Implementing low-power motors and smart power management (such as sleep modes for idle components) could extend battery life.
    - **Alternative Power Source:** Exploring higher-capacity lithium-ion batteries with fast-charging capabilities could enhance runtime and convenience.
3. **Enhanced Cleaning System**
    - **More Effective Dust Collection:** The current suction mechanism works well for fine dust but struggles with larger debris. Increasing suction power and optimizing the design would improve efficiency.
    - **Adding a Wiping Function:** The robot currently sweeps and collects dust, but adding a water-based wiping system with a small water tank and microfiber pads would allow it to clean spills and remove sticky dirt, making the cleaning process more thorough.
4. **Improved Navigation Capabilities**
    - **Going Down Stairs:** The current model can climb stairs but cannot descend them. Future versions should have a safe and controlled way to move down, making it fully automatic in both directions.
    - **Outdoor Adaptation:** Modifying the robot to handle outdoor stairs by improving weather resistance, grip on uneven surfaces, and dust-proofing electronics will expand its usability beyond indoor environments.
5. **Self-Emptying and Maintenance Features**
    - **Self-Emptying Dustbin:** A self-cleaning dustbin would make the robot more autonomous, requiring minimal human intervention.
    - **Automatic Charging Station:** Incorporating an automatic charging dock would allow the robot to recharge itself when the battery is low, enhancing usability and convenience.

References

[1] Gizmodo. *Migo Ascender: The First Stair-Climbing Vacuum Robot*. https://www.gizmodo.com/migo-ascender. Accessed: 2024-06-22. 2024. url: https://www.kickstarter.com/projects/migoascender/migo-ascender-the-stair-climbing-robot-vacuum.

[2] MDPI. "Advancements in Sensor-Based Navigation for Cleaning Robots". In: *Sensors* 20.15 (2023). Accessed: 2024-06-22, pp. 1–10. url: https://www.mdpi.com/1424-8220/24/5/1377.

[3] J. Park H. Kim and Y. Kim. "sTetro: A Reconfigurable Stair-Climbing Cleaning Robot". In: *Sensors* 21.18 (2021). Accessed: 2024-06-22, p. 6279. url: https://www.mdpi.com/1424-8220/21/18/6279.

[4] Luxonis. *OAK-D Camera Documentation*. Accessed: 2024-08-30. 2024. url: https://shop.luxonis.com/products/oak-d.