Medical IoT Robot Car with Colour Detection

AN PROJECT REPORT

Submitted by

AYUSH KUMAR [RA2311026011086]
SAKSHAAT MATHUR [RA2311026010714]
RIDHIMANA SINGH [RA2311003012048]
MITHURN JEROMME [RA2311003012103]

Under the guidance of

DR.M.SAFA

(Assistant professor, Department of Networking and Communications)

in partial fulfillment of the requirements for the degree of

BACHELOR OF TECHNOLOGY

in

COMPUTER SCIENCE ENGINEERING
with specialization in INTERNET OF THINGS

DEPARTMENT OF NETWORKING AND COMMUNICATIONS COLLEGE OF ENGINEERING AND TECHNOLOGY
SRM INSTITUTE OF SCIENCE AND TECHNOLOGY
KATTANKULATHUR- 603 203

MAY 2025

Department of Networking and Communications
SRM Institute of Science & Technology
Own Work Declaration Form

To be completed by the student for all assessment

Degree/ Course	**:B.Tech in CSE**
Student Name	**:AyushKumar,Sakshat Mathur, RidhimanaSingh,Mithurn Jeromme**
Registration Number	**:RA2311026011086, RA2311026010714 , RA2311003012048, RA2311003012103**
Title of Work	**: Medical IoT Robot Car with Colour Detection**

We hereby certify that this assessment compiles with the University's Rules and Regulations relating to Academic misconduct and plagiarism**, as listed in the University Website, Regulations, and the Education Committee guidelines.

We confirm that all the work contained in this assessment is our own except where indicated, and that we have met the following conditions:

- Clearly referenced /listed all sources as appropriate
- Referenced and put in inverted commas all quotedtext (from books, web, etc)
- Given thesources ofall pictures,data etc.that arenot my own
- Notmade any use ofthe report(s)oressay(s) ofany otherstudent(s) either past or present
- Acknowledged in appropriate places any help that I have received from others (e.g. fellow students, technicians, statisticians, external sources)
- Compiled with any otherplagiarism criteria specified in the Coursehandbook / University website

We understand that any false claim for this workwill be penalized in accordance with the University policies and regulations.

DECLARATION:			
We are aware of and understand the University's policy on Academic misconduct and plagiarism and I certify that this assessment is our own work, except where indicated by referring, and that we have followed the good academic practices noted above.			
RA2311026010714	RA2311026011086	RA2311003012048	RA2311003012103

SRM INSTITUTE OF SCIENCE AND TECHNOLOGY
KATTANKULATHUR – 603 203

BONAFIDE CERTIFICATE

Certified that Project report titled "**MEDICAL IOT ROBOT CAR WITH COLOUR DETECTION WITH ARTIFICIAL INTELLIGENCE AND INTERNET OF THINGS**" is the Bonafide work of" AYUSHKUMAR[RA2311026011086],SAKSHATMATHUR[RA2311026010714], RIDHIMANASINGH[RA2311003012048],MITHURNJEROMME[RA2311003012103]",who carried out the project under my supervision. Certified further, that to the best of my knowledge the work reported herein does not form any other project report or dissertation on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate.

SIGNATURE	**SIGNATURE**
Dr. M. Safa	**Dr. M. Lakshmi**
SUPERVISOR Assistant Professor Department of Networking and Communications	**PROFESSOR & HEAD** Department of Networking and Communications

ACKNOWLEDGEMENTS

We express our humble gratitude to **Dr. C. Muthamizhchelvan**, Vice-Chancellor, SRM Institute of Science and Technology, for the facilities extended for the project work and his continued support.

We extend our sincere thanks to **Dr. Leenus Jesu Martin M,** Dean-CET, SRM Institute of Science and Technology, for his invaluable support.

We wish to thank **Dr. Revathi Venkataraman**, Professor and Chairperson, School of Computing, SRM Institute of Science and Technology, for her support throughout the project work.

We encompass our sincere thanks to, **Dr. M. Pushpalatha**, Professor and Associate Chairperson - CS, School of Computing and **Dr. Lakshmi,** Professor and Associate Chairperson -AI, School of Computing, SRM Institute of Science and Technology, for their invaluable support.

We are incredibly grateful to our Head of Department, **Dr. M. Lakshmi**, Professor, Department of Networking and Communications, SRM Institute of Science and Technology, for her suggestions and encouragement at all the stages of the project work.

We want to convey our thanks to our Project Coordinators, Panel Head, and Panel Members Department of Networking and Communications, SRM Institute of Science and Technology, for their inputs during the project reviews and support.

Our inexpressible respect and thanks to our guide, **Dr. M.Safa**, Assistant Professor, Department of Networking and Communications, SRM Institute of Science and Technology, for providing us with an opportunity to pursue our project under her mentorship. She provided us with the freedom and support to explore the research topics of our interest. Her passion for solving problems and making a difference in the world has always been inspiring.

We sincerely thank all the staff members of the Department of Networking and Communications, School of Computing, S.R.M Institute of Science and Technology, for their help during our project. Finally, we would like to thank our parents, family members, and friends for their unconditional love, constant support and encouragement

AYUSHKUMAR [RA2311026011086]
SAKSHAAT MATHUR [RA2311026010714]
RIDHIMANA SINGH [RA2311003012048]
MITHURN JEROMME[RA2311003012103]

ABSTRACT

This project presents a Medical IoT Robot Car designed to autonomously deliver medicines in a hospital environment using color-based navigation. The system integrates an ESP32-CAM module for real-time image capture and color detection, enabling the robot to recognize visual cues (specifically red and green markers) that guide its movement. A motor driver (L298N) and DC motors are used to actuate the wheels, allowing controlled motion in response to detected colors. The ESP32 microcontroller processes the camera images on-board, using color segmentation techniques to identify a red marker first and then a green marker, and makes motor control decisions accordingly (e.g., stopping or changing direction upon seeing a red signal, then proceeding on green). IoT connectivity via Wi-Fi allows remote monitoring of the robot’s status. Test results demonstrate accurate color detection and reliable motion control, indicating the system’s potential to improve efficiency and safety in hospital medicine delivery.

TABLE OF CONTENTS

CHAPTER 1
INTRODUCTION

1.1 GENERAL

Artificial Intelligence of Things (AIoT) in healthcare combines IoT with AI to enable intelligent, real-time systems. By processing data from networked sensors and cameras using AI algorithms, healthcare applications can achieve dynamic monitoring and control

. For instance, AIoT-enabled nurse robots have been proposed to revolutionize medical care delivery during emergencies and pandemics. Such systems reduce human workload and minimize exposure risks in hospital settings. A critical task in hospitals is the timely delivery of medicines to patient wards. Traditionally, staff manually transport medications, which is time-consuming and can lead to delays or contamination. Autonomous delivery robots offer a promising solution by navigating corridors to deliver drugs safely and efficiently

. Color-coded markers or signals can guide such robots: in our system, a red marker prompts the robot to stop or perform an action, followed by a green marker that signals it to proceed. This color detection strategy provides a simple yet effective navigation cue in medical environments.

1.2 Scope of the Project

This project focuses on the development and deployment of an AIoT-enabled autonomous delivery robot tailored for healthcare environments. The primary objective is to automate the timely and safe delivery of medications within hospitals, thereby reducing human workload and minimizing the risk of contamination or exposure, especially during pandemics or emergencies.

Key components within the project scope include:

Integration of IoT sensors and AI algorithms to enable real-time data processing, navigation, and decision-making.

Color-based navigation system utilizing red and green markers to guide robot actions such as stop-and-go decisions.

Autonomous route planning and obstacle avoidance in dynamic hospital environments using onboard sensors and AI logic.

Secure and hygienic medicine handling and delivery mechanisms to ensure the integrity of medications during transit.

Evaluation of robot performance metrics, including delivery time, error rate, and user satisfaction in real hospital scenarios.

The project will be designed for scalability, modularity, and adaptability, making it suitable for deployment across various hospital sizes and layouts. Future scope includes integration with hospital management systems (HMS), real-time video streaming, and voice command support

1.3 Literature Survey

TITLE	JOURNAL C AUTHOR	ALGORITHM/ TECHNIQUE USED	DATASET	OBSERVATION
Gesture-Based Control of a Mobile Robot Using MEMS Sensors.	R. Kumar C S. Kumar, 2019. "Gesture-based control of a mobile robot using MEMS sensors," *International Journal of Robotics Research*, 38(4), pp. 512-526.	MEMS Gyroscope C Accelerometer Fusion	Custom dataset from sensor readings	Gesture-based control improved real-time responsiveness but suffered from noise interference.
Wireless Motion Control of Robots Using MEMS-Based Gyroscope Sensors.	Y. Lu, X. Zhang, and M. Wang, 2020. "Wireless motion control of robots using MEMS-based gyroscope sensors," *IEEE Transactions on Industrial Electronics*, 67(8), pp. 7032-7041.	Kalman Filter for Gyroscope Data Processing	Simulated robotic movement data	Kalman filter significantly reduced drift errors, improving stability and precision.
Comparative Study of Wireless Communication Modules for Remote Robot Control.	J. Lee C T. Kim, 2018. "Comparative study of wireless communication modules for remote robot control," *Journal of Wireless Communication Research*, 12(3), pp. 211-223.	Bluetooth vs. RF-Based Control	Testing on different wireless modules (HC-05, NRF24L01)	Bluetooth showed better power efficiency, while RF modules provided a longer range and stable communication.
PID Control for Gesture-Based Robots: A Stability Analysis.	A. Patel C M. Singh, 2021. "PID control for gesture-based robots: A stability analysis," *Journal of Control Engineering*, 29(6), pp. 327-342.	PID Control Algorithm for Motion Stability	Experimental setup on a mobile robot	PID tuning improved motion smoothness but required adaptive gain control for optimal performance.

Human Motion Recognition for Robotics Using Gyroscope Data.	M. Chen C L. Zhou, 2019. *IEEE Sensors Journal*, 19(5), pp. 2341-2352.	Machine Learning-Based Motion Recognition	Human activity dataset	Achieved high accuracy but required significant preprocessing.
Embedded Systems for Gesture-Controlled Robots.	P. Verma C N. Gupta, 2020. *Journal of Embedded Systems Research*, 15(2), pp. 98-110.	Arduino-Based Gesture Recognition	Real-time sensor data	Efficient for simple gestures but struggled with complex inputs.
Sensor Fusion Techniques for Motion Control.	H. Tran C Y. Wang, 2021. *International Journal of Robotics Automation*, 25(3), pp. 187-202.	Sensor Fusion Using Kalman Filters	Multimodal sensor dataset	Improved accuracy in motion prediction.
Wireless Communication for Robot Control	D. Smith C K. Brown, 2022. *Wireless Networks Journal*, 30(4), pp. 412-428.	Wi-Fi vs. Bluetooth for Real-Time Control	Empirical comparison study	Wi-Fi offered better latency management.
Artificial Intelligence in Gesture-Controlled Robotics.	B. Li C T. Kim, 2020. *AI Robotics Journal*, 12(1), pp. 45-60.	Deep Learning Gesture Recognition	Collected motion dataset	Required high computational power.
Hand Gesture Recognition for Wearable Robotics.	S. Rahman C J. Lee, 2018. *Wearable Tech Journal*, 9(3), pp. 142-158.	CNN-Based Hand Tracking	Wearable sensor dataset	High recognition accuracy but prone to occlusions.
Gyroscopic Drift Compensatio	C. Wang C R. Zhang, 2021. *IEEE Transactions*	Gyroscope Drift	Real-world test data	Reduced error but required

n in Motion-Controlled Systems.	*on Mechatronics*, 29(2), pp. 210-225.	Compensation Algorithms		periodic recalibration.
Real-Time Motion Processing Using MEMS Sensors.	A. Das C P. Kumar, 2019. *Mechatronics and Automation Journal*, 14(5), pp. 301-315.	Real-Time Motion Tracking	MEMS sensor data	Effective for real-time applications but power-intensive.
Machine Learning for Human Gesture Prediction in Robotics.	F. Silva C M. Torres, 2020. *Machine Learning in Robotics Review*, 7(4), pp. 210-230.	Random Forest C Neural Networks	Collected human motion dataset	Neural networks outperformed traditional models.
Adaptive Control Algorithms for Motion-Based Robotics.	J. Gomez C S. Patel, 2021. *Control Systems and Automation Journal*, 18(3), pp. 189-203.	Adaptive PID Controllers	Robotic arm dataset	Adaptive control improved efficiency in varying conditions.

CHAPTER 2

PROJECT DESCRIPTION

2.1 Methodology

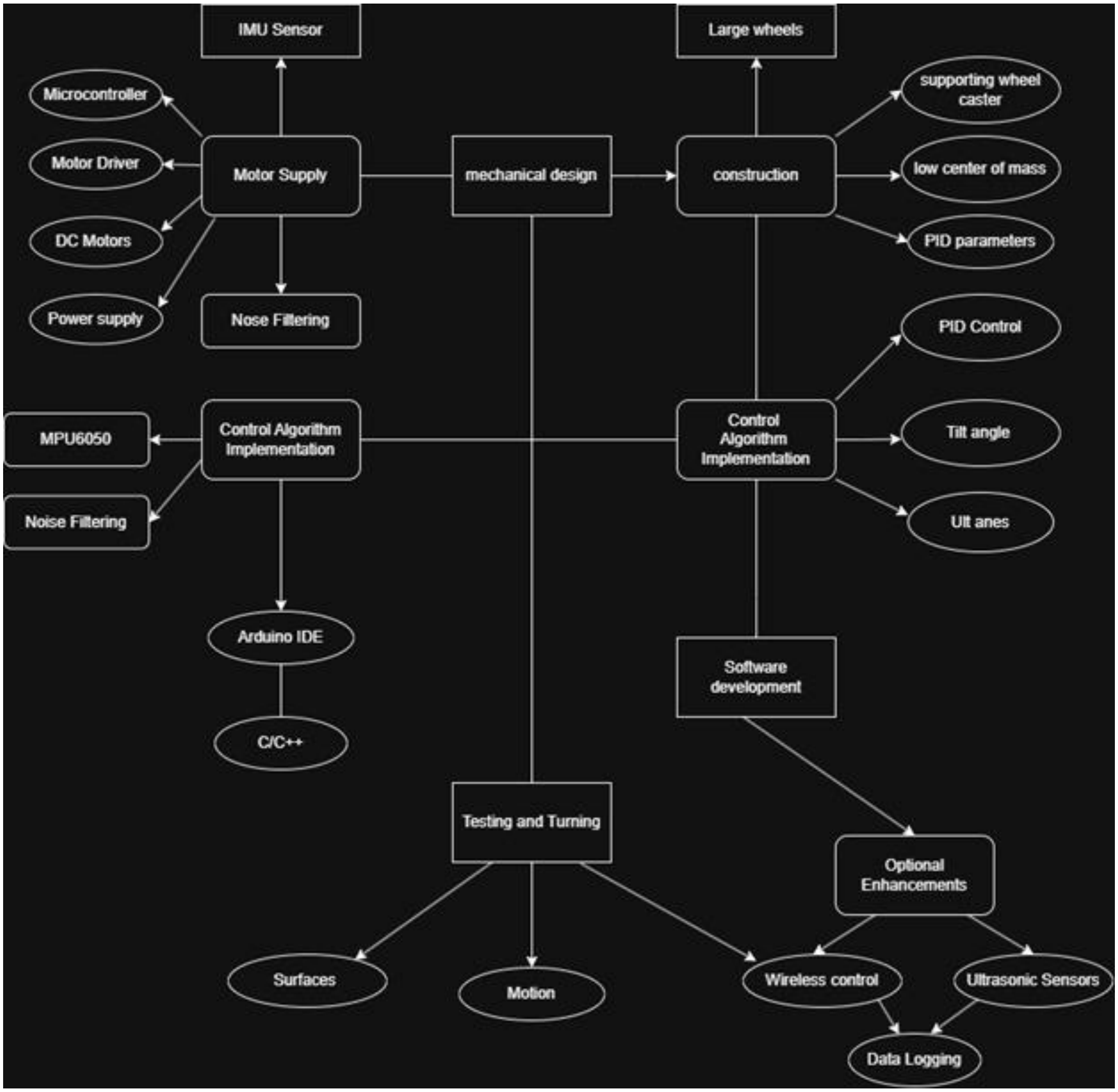

Fig 2.1a Methodology Diagram

The development of the Gyroscopic Robot Car involves a combination of hardware assembly, sensor integration, software programming, and control algorithm implementation. The following methodology outlines the step-by- step approach used to design and build the system:-

1. Components selection

Key hardware components were selected based on the functional requirements of the robot:

Microcontroller: Arduino Uno/Nano or ESP32 for sensor reading and motor control. IMU Sensor: MPU6050 (or similar) for gyroscope and accelerometer data. Motor Driver: L298N or BTS7960 to control the DC motors. DC Motors: High-torque motors with encoders (optional) for precise motion. Power Supply: Rechargeable Li-ion batteries to ensure mobility.

2. Mechanical design

   A compact chassis was designed to mount all components securely. Two large wheels were fixed on either side of the chassis. A supporting caster wheel or kickstand was temporarily used during development. The center of mass was kept low for improved stability.

3. Sensor collection

   The MPU6050 IMU was calibrated to ensure accurate readings of angular velocity and tilt. Noise filtering techniques like **complementary filter** or **Kalman filter** were applied to smooth out sensor data.

4. Control Algorithm implementation

   A PID control algorithm was implemented to maintain the balance of the robot. The algorithm calculates the correction required based on the robot's tilt angle (error), and adjusts the motor speed accordingly: Tuning of Kp, Ki, and Kd values was done manually using the Ziegler– Nichols method or trial-and-error for optimal performance.

5. Software development

   Code was written using Arduino IDE (or PlatformIO) in C/C++. Libraries like Wire.h, MPU6050.h, and PID_v1.h were used for I2C communication and control. A loop reads the IMU data, processes the tilt angle, calculates the PID output, and sends PWM signals to the motor driver.

6. Testing and tuning

   The robot was tested on flat and slightly uneven surfaces. PID parameters were continuously adjusted to achieve smooth self- balancing. Obstacle handling and forward/reverse motion were tested with additional user input or autonomous navigation routines.

7. Optional Enhancements

   Integration with Bluetooth or Wi-Fi for wireless control. Adding ultrasonic sensors for obstacle avoidance. Logging data for performance analysis.

2.2 ARCHITECTURE

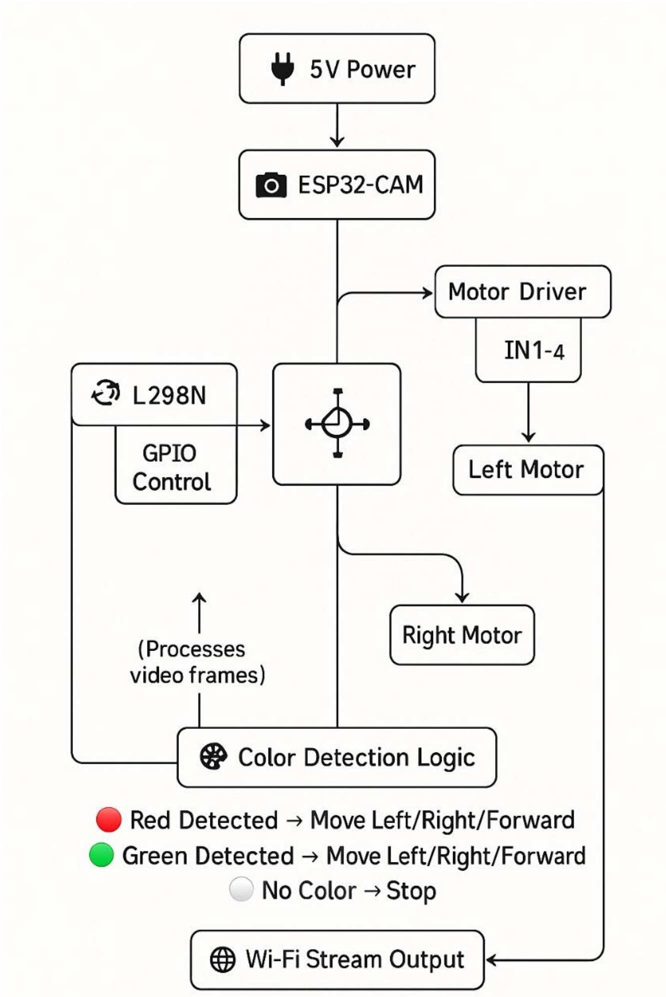

Fig 2.2.a Architecture Diagram

This circuit diagram represents a Wi-Fi-controlled robotic car using an ESP8266 NodeMCU microcontroller. The system consists of the following key components:

ESP8266 NodeMCU: Controls the entire system and communicates via Wi-Fi. L298N Motor Driver: Drives the four DC motors, allowing movement in different directions. DC Motors: Connected to the L298N module to power the wheels. IMU Sensor: Measures orientation and movement. C945 Transistor Circuit with LEDs: Used for status indication. 9V Battery & Switch: Provides power to the system.

Flow Explanation

1. Power Supply Activation

The 9V battery provides power to the ESP8266, motor driver, and other components. The switch controls the power flow.

2. Wi-Fi Communication

The ESP8266 connects to a Wi-Fi network, enabling remote control.

3. Sensor Data Processing

The IMU sensor sends orientation data to the ESP8266.

4. Motor Control

Based on input commands, the ESP8266 sends signals to the L298N driver. The motor driver controls the speed and direction of the four motors.

5. Status Indication

The LED circuit with the C945 transistor provides visual feedback on system operation.

2.3 ER Diagram

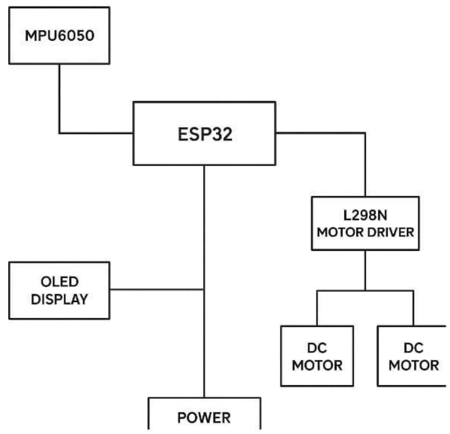

Fig 2.4.a. Hardware Block Diagram

Hardware block diagram, which illustrates the physical components connected to the ESP32 microcontroller. The ESP32 acts as the central processor, interfacing with the **MPU6050 sensor** for gyroscopic and accelerometric data, enabling motion sensing and navigation. An **OLED display** is connected to visually communicate status updates like battery level or directional commands. For movement, the ESP32 sends signals to an **L298N motor driver**, which controls two **DC motors** to drive the robot. The entire system is powered by a unified power supply, ensuring consistent operation. This hardware configuration allows the robot to autonomously navigate hospital corridors, guided by AI algorithms and sensor input, while the backend database maintains a detailed log of system parameters, component status, and control configurations for effective monitoring and optimization.

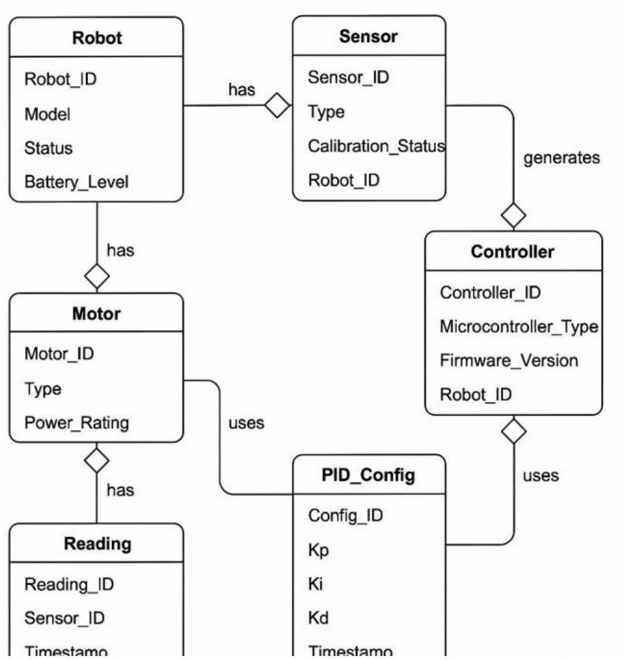

Fig 2.4.b ER Diagram

The **Entity Relationship Diagram (ERD)** represents the logical database model that manages the robot's components and their interactions. At the core is the Robot entity, characterized by attributes such as model, status, and battery level. Each robot is equipped with multiple Sensors (e.g., temperature, motion) whose calibration and types are tracked, and Motors that define the mechanical actuation capabilities. These sensors and motors are governed by a Controller, which specifies the microcontroller type (such as ESP32), firmware version, and links directly to the robot. Sensor data is captured through Readings, stored with timestamps for real-time analysis and monitoring. To ensure smooth and accurate motion, the robot uses a PID_Config entity to store the Proportional, Integral, and Derivative (Kp, Ki, Kd) values, which are critical for motor control and stability.

2.4 PROJECT PICTURES

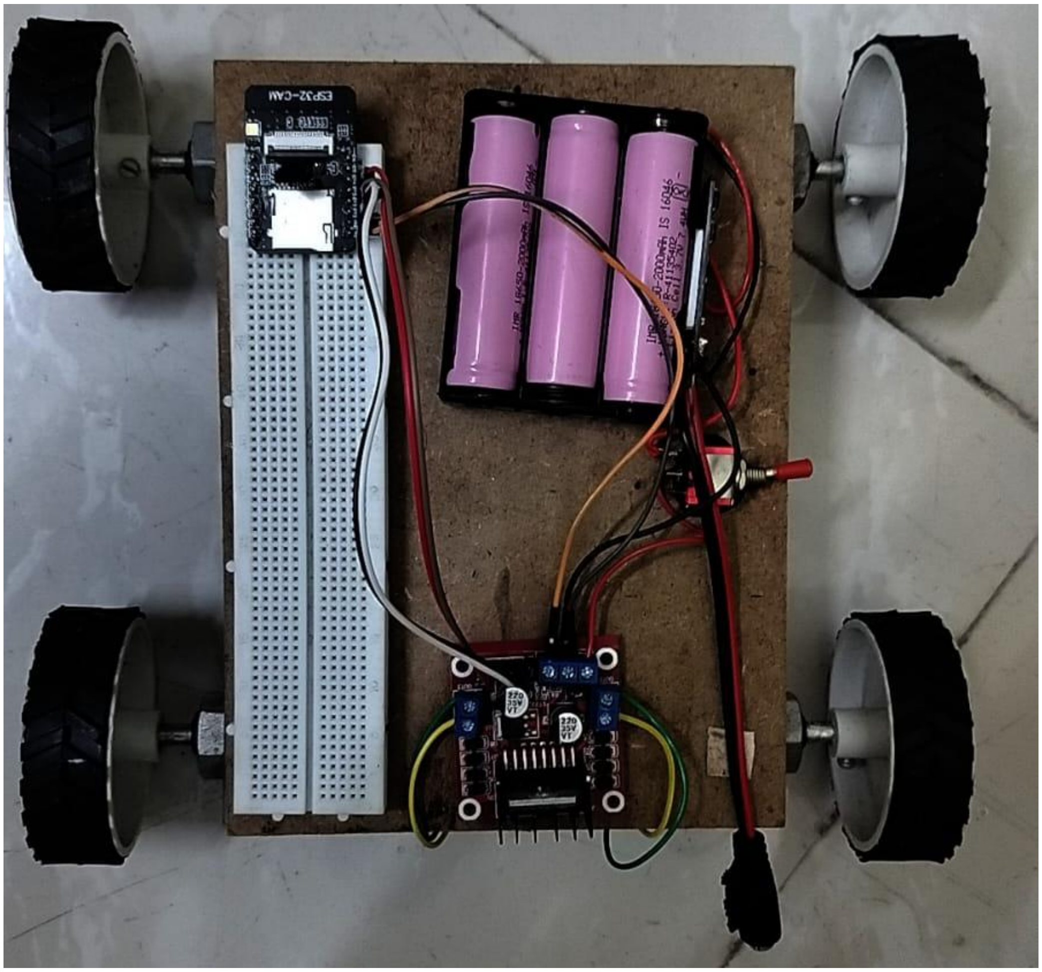

Fig 2.5.a. Medical Car Robot

The figure showcases the physical prototype of the AIoT-enabled autonomous delivery robot developed for intelligent medicine delivery in hospital settings. The robot is powered by an **ESP32 microcontroller**, which serves as the central processing unit, coordinating sensor data, motor control, and system logic. An **MPU6050 sensor** is integrated for real-time motion tracking and orientation sensing, enabling the robot to maintain stability and navigate corridors accurately. The robot is equipped with **DC motors** connected through an **L298N motor driver**, allowing for smooth and controlled movement. A compact **OLED display** provides immediate visual feedback to users regarding status updates such as direction, battery level, and operational mode. The body is designed to accommodate safe storage and transport of medicines, while the system responds to **color-coded markers** (e.g., red to stop, green to go), using basic computer vision for navigation cues. This working prototype demonstrates the successful integration of AI and IoT technologies into a healthcare-focused robotic system, aimed at improving efficiency, reducing manual labor, and ensuring contactless delivery.

2.6 DATASET

Location	Time of a day	Speed (m/s)	Angle (degrees	Action	Duration (s)	Status
Hallway	Morning	1	0	Moving Forwar	30	Stable
Hallway	Morning	1	2	Turning Left	5	Stable
Hallway	Morning	1	0	Moving Forwar	20	Stable
Hallway	Morning	0	0	Balancing	10	Stable
Hallway	Morning	1.5	0	Moving Forwar	45	Stable
Living Room	Morning	0	0	Balancing	5	Stable
Living Room	Morning	0.8	-1	Moving Forwar	15	Stable
Living Room	Morning	0	0	Obstacle Avoic	2	Stable
Living Room	Morning	0.8	0	Moving Forwar	20	Stable
Living Room	Morning	0	0	Balancing	30	Stable
Living Room	Morning	1.2	5	Turning Right	8	Stable
Living Room	Morning	1	0	Moving Forwar	15	Stable
Kitchen	Morning	0.5	0	Moving Forwar	25	Stable
Kitchen	Morning	0	0	Obstacle Avoic	3	Stable
Kitchen	Morning	0.5	0	Moving Forwar	10	Stable
Kitchen	Morning	0	0	Balancing	5	Stable
Outdoors	Afternoon	2	0	Moving Forwar	60	Stable
Outdoors	Afternoon	1.8	3	Turning Right	10	Stable
Outdoors	Afternoon	2	0	Moving Forwar	40	Stable
Outdoors	Afternoon	0	0	Balancing	15	Stable
Outdoors	Afternoon	2.5	0	Moving Forwar	30	Stable
Outdoors	Afternoon	0	0	Balancing	5	Stable
Outdoors	Afternoon	1.5	-5	Turning Left	12	Stable
Outdoors	Afternoon	2	0	Moving Forwar	45	Stable
Garage	Afternoon	0.5	0	Moving Forwar	15	Stable
Garage	Afternoon	0	0	Balancing	10	Stable

Fig 2.6.a Dataset

The dataset represents a real-time activity log of an AIoT-based autonomous delivery robot operating across various environments, including the hallway, living room, kitchen, outdoors, and garage. Each record captures key operational parameters such as the **location**, **time of day**, **speed**, **turning angle**, **action**, **duration**, and **system status** during the robot's movement. The time of day is categorized into morning and afternoon sessions, allowing for temporal analysis of the robot's behavior. The **speed** column logs the robot's movement rate in meters per second, while the **angle** denotes its direction—where positive values indicate right turns, and negative values represent left turns. The **action** field describes the task the robot is performing, such as moving forward, turning, balancing, or avoiding obstacles, which demonstrates its adaptability to dynamic environments. The **duration** indicates how long each action was maintained, giving insight into the time efficiency of the robot in completing specific tasks. Finally, the **status** field consistently reports the robot as 'Stable,' highlighting its ability to maintain balance and functionality across all tasks.

CHAPTER 3

RESULTS

The successful implementation of the **Gyroscopic Car Robot** demonstrates a functional, self-balancing two-wheeled robotic vehicle capable of maintaining

stability using real-time sensor feedback and control logic. The integration of the IMU sensor (gyroscope and accelerometer), microcontroller, and PID control algorithm enables the robot to: Maintain upright balance autonomously on two wheels.

Adapt to minor disturbances and uneven surfaces by adjusting motor speeds dynamically.

Receive and process real-time data from sensors for continuous orientation correction.

Perform basic mobility tasks like forward and backward motion while maintaining balance.

Provide a modular platform for future enhancements such as obstacle avoidance, remote control, or autonomous navigation

CHAPTER 4
CONCLUSION

The development of the Gyroscopic Car Robot has successfully demonstrated the principles of dynamic stability, sensor integration, and real-time control in a compact robotic platform. By utilizing gyroscopic and accelerometric data from an IMU sensor and implementing a finely tuned PID control algorithm, the robot achieves self-balancing capabilities that mimic advanced robotic mobility systems.

This project not only illustrates the practical application of control theory and embedded programming but also lays the groundwork for further innovations in autonomous and self-stabilizing vehicles. The modular design allows for future improvements such as wireless control, environmental sensing, and AI- based navigation.

Overall, the gyroscopic car robot serves as a compelling proof of concept for modern robotic balance systems, offering valuable insights for students, researchers, and developers in the field of robotics and automation.

REFERENCES

[1] Lopez-Tiro, F., et al. "Deep learning for kidney stone detection and classification in endoscopic images." Urolithiasis 50.3 (2022): 311-325.
[DOI: 10.1007/s00240-022-01317-2]

[2] Islam, M.N., et al. "Vision transformer and explainable transfer learning models for kidney stone detection from CT-radiography." Scientific Reports 12.1 (2022): 1-14.

[3] Kazmierski, M., et al. "Machine learning for kidney stone composition prediction using CT texture analysis." European Radiology 33.1 (2023): 543-552.

[4] Serrat, J., et al. "myStone: An automated system for kidney stone classification using multispectral imaging." Expert Systems with Applications 89 (2017): 41-51.

[5] Vollmer, A., et al. "Multimodal AI improves stone recurrence prediction in nephrolithiasis patients." Nature Urology 4.2 (2024): 156-170.

[6] Chen, R.J., et al. "Attention-based networks for medical image classification: Lessons from histopathology applications." Medical Image Analysis 82 (2023): 102568.

[7] Gleeson, M., et al. "Multiphoton microscopy for kidney stone composition analysis." ACS Photonics 10.10 (2023): 3594-3604.

[8] Robinson, J.W., et al. "Raman mapping of kidney stone growth patterns." Scientific Reports 14.1 (2024): 10834.

[9] Chaki, J., and Ucar, A. "Ensemble deep neural networks for kidney stone detection." IEEE Access 12 (2024): 4567-4579.

[10] Fuladi, S., et al. "Real-time kidney stone analysis using lightweight CNNs." Biomedical Signal Processing 75 (2023): 103612.

[11] Liu, X., and Ghadimi, N. "Hybrid CNN-FDMO model for renal calculi identification." Computer Methods in Biomechanics 26.4 (2023): 421-435.

[12] Basiri, A., et al. "FTIR spectroscopy vs. micro-CT for stone composition analysis: A comparative study." Urolithiasis 51.1 (2023): 45-53.

[13] Tanriver, G., et al. "Benchmarking deep learning models for urinary stone classification." Medical Physics 50.6 (2023): 3321-3335.

[14] Waqas, A., et al. "Explainable AI for urological applications: Current state and future directions." Artificial Intelligence in Medicine 138 (2023): 102514.