Development of Service Robot Manipulator for Pick and Place applications in Domestic Environment

Ronit Shah[1], and *Arockia Selvakumar Arockia Doss*[1*]

[1]Vellore Institute of Technology, Chennai, India

Abstract. Service robots have been developed to assist humans in daily tasks, particularly in environments where repetitive or complex manipulations are required. Robotic manipulators have previously been studied mostly for factory or organized applications, usually with a lack of flexibility, intuitive control, or adaptability for general-purpose service tasks. This project specifically focuses on applications within domestic environments, addressing those limitations by developing a service robot manipulator capable of both autonomous object handling and manual control via a user-friendly interface. The study was conducted in two phases: a simulation environment and a physical prototype. In the simulation phase, object detection was performed using the YOLOv8 Object Box Bounding model. The detected object positions were calculated. Further robotic path planning and execution were carried out using ROS and MoveIt frameworks. In the prototype phase, an ESP32-based manipulator was built using servo motors. The ESP32 hosted a Wi-Fi access point and served a web-based interface where users could manipulate the arm through sliders and control buttons. Real-time bidirectional communications was established via WebSocket, allowing the user to record motion sequences and replay them with accurate timing. Results from both simulation and prototype implementation demonstrate successful grasping, motion planning, and repeatability of tasks. This study contributes to existing research by presenting a modular, low-cost, and interactive system that can be applied in various service applications, from home assistance to small-scale automation.

1 Introduction

1.1 Background

Service robots are increasingly important in performing tasks for humans on a daily basis, especially in areas where repetitive or detailed manipulation is involved. Service robots are also intended to operate in varied and dynamic settings such as households, healthcare

* Corresponding author: arockia.selvakumar@vit.ac.in

environments, and small-scale service industries. Conventional robotic manipulators, on the other hand, have been developed mostly for well-structured industrial environments where activities and environment are very controlled. These platforms tend to be missing the flexibility, intuitive management, and usability requirements of general-purpose service applications. Although sophisticated robotic platforms such as ROS (Robot Operating System) and computer vision models such as YOLO have boosted robotic perception and autonomy, there still exists a chasm in transforming these features into available, real-world service applications. Thus, a more emerging interest is in developing modular, inexpensive robotic systems that blend autonomous capabilities with user control to the point of suitability for non-expert consumers in daily contexts.

1.2 The Problem

Even with the progress in robotic technologies, current service robot manipulators tend to be limited in flexibility, usability, and cost. The majority of research and deployments are centered on high-cost industrial manipulators or highly structured systems with little user interaction. Such solutions are not scalable or feasible for home care or small-scale service applications, where flexibility, intuitive interfaces, and low-cost components are essential. Additionally, existing systems rarely provide a smooth blend of autonomous and manual operation, limiting their applicability to structured and dynamic situations. The deficiency points to the need for a system which is not only functionally effective but also user-friendly and interactive for non-technical users.

1.3 The Solution

To overcome such limitations, the current project aims to design and develop a hybrid service robot manipulator that handles manual manipulation through an interactive web interface. The system was developed and integrated in two phases: simulation and physical prototyping. During simulation, YOLOv8 with Oriented Bounding Boxes (OBB) was applied for precise object detection, and robotic motion planning and execution were managed by ROS and MoveIt frameworks. The physical prototype was constructed using an ESP32 microcontroller and servo motors with wireless control from a browser-based interface via WebSocket communication. Users could naturally manipulate the manipulator, capture motion sequences, and replay them with accuracy. The resulting system is shown to execute tasks successfully, be modular, cost-efficient, and improve human-robot interaction, making it a suitable solution for different service-oriented applications from personal aid to light automation tasks.

2 Literature Review

The integration of service robot manipulators into assistance roles represents a transformative shift in healthcare, domestic support, and industrial collaboration. This literature review synthesizes advancements in robotic manipulation, human-robot interaction (HRI), and control strategies, drawing from studies published in recent years. Key findings reveal that modular hardware architectures, AI-driven perception systems, and context-aware communication protocols are critical for enabling robots to perform complex tasks in unstructured environments. Applications span meal assistance, elderly care, hospital logistics, and telemedicine, with empirical evidence demonstrating efficiency gains of 30–50% in task completion times compared to manual methods. However, technical limitations in handling non-rigid objects and cultural resistance to automated care persist as barriers to adoption. Emerging solutions include hybrid control systems combining deep learning with tactile feedback and anthropomorphic designs improving social acceptance. The COVID-19 pandemic accelerated deployment timelines by 42% in healthcare settings, underscoring robots' potential in infection control.

2.1 Evolution of Service Robots

Service robots originated from industrial systems designed for repetitive tasks in controlled settings. The introduction of standardized definitions such as ISO 8373 (2012) clarified their autonomy and focus on non-manufacturing roles. Early service robots reused industrial components but shifted toward dynamic environments including healthcare, logistics, and households. A major driver of this evolution has been progress in human–robot interaction (HRI), moving robots from simple tools to relational partners that co-create value [1]. Sun and Wang [2] describe this development in three stages: foundational design, domain-specific applications, and recent work emphasizing human-centric and ethical concerns. Interest in autonomy, user experience, and social interaction has grown, and the COVID-19 pandemic accelerated adoption in disinfection, delivery, and patient monitoring [3]. This marks a shift toward adaptive, socially aware systems embedded in daily life.

2.2 Control Strategies and Techniques

Control strategies in service robotics must adapt to unstructured, human-centered environments. Core paradigms include kinematic and dynamic control, teleoperation versus autonomy, and sensor-based feedback mechanisms. Kinematic control focuses on joint trajectory planning, while dynamic control incorporates force modeling to enhance precision. Adaptive learning controllers such as PA10 and Panda manipulators have demonstrated robustness against disturbances like harmonic noise [4]. For whole body tasks like scrubbing or sweeping, compliant control combined with knowledge-based adaptation is essential [5]. Iterative learning control integrated with computed torque methods has shown improvements in trajectory tracking for multi-body robotic systems.

Teleoperation remains valuable in high-risk applications, but autonomy offers better scalability. In domestic settings, robots are increasingly utilizing learning-based approaches such as Learning from Demonstration (LfD), supervised learning, and reinforcement learning to generalize actions like cleaning and navigation [6].

Recent studies, such as Sharma et al. [7], emphasize modular robot design for domestic automation, supporting the need for lightweight manipulators with intuitive interfaces. Sensor-based control, modeled after the human nervous system, employs visual, tactile, auditory, and distance sensors to enable adaptive responses. The integration of multiple sensors (e.g., vision with tactile sensing) enhances the quality of human–robot interaction. In elderly rehabilitation, intelligent sensing and thermal feedback combined with machine learning have been leveraged to promote both safety and comfort [8].

2.3 Application of Service Robot Manipulator

Service robot manipulators now support hospitality, healthcare, domestic tasks, and elderly care. In hospitality, they assist with table service and concierge functions, improving efficiency and customer experience [9,10,11,12]. Systems like Sacarino and

Japan's Henn-na Hotel show long-term feasibility, though emotional and empathetic abilities remain limited [13].

In healthcare, especially after COVID-19, robots reduce infection risk, manage logistics, and support caregivers [14]. In homes, robots such as vacuum cleaners have changed routines and influenced technology acceptance [15]. In elderly care, robots assist with daily activities and social engagement, with platforms like COMET/UML enabling real-time autonomy and better reliability [16,17,18]. Field studies show high acceptance due to user-centered design.

Catering robots represent a new application area, handling tasks like table clearing and customer interaction [19]. Customer perceptions depend on factors such as anthropomorphism, autonomy, and role clarity. Research based on TAM highlights the importance of ethical leadership and strategic communication to build trust and encourage adoption [20].

2.4 AI and ML Integration

Artificial intelligence (AI) and machine learning (ML) have significantly enhanced robot adaptability, interaction capabilities, and autonomy. Modern dual-arm household robots increasingly utilize multimodal large language models (MLLMs), enabling them to interpret both textual and visual input to autonomously plan and execute tasks [21]. Advanced systems powered by models like GPT-4o mini and Gemini 1.5 Flash demonstrate notable improvements in object manipulation and spatial reasoning.

In retail and warehousing, large language models (LLMs) serve as intermediaries that convert natural language commands into symbolic plans, which are further optimized through Monte Carlo Tree Search (MCTS) and interactive user feedback [22]. In industrial applications, deep reinforcement learning (DRL) allows robots to learn task-specific behaviors through simulation. Hybrid frameworks that integrate DRL with classical control methods such as Deep Deterministic Policy Gradient (DDPG) combined with Linear Quadratic Regulator (LQR) have been shown to improve energy efficiency and task precision in pick-and-place scenarios [23, 24].

Machine learning techniques have also advanced robot perception, enabling real-time object recognition, adaptive force control, and dynamic obstacle avoidance. Approaches involving fuzzy logic and neuro-fuzzy systems offer robust performance in unpredictable environments [25]. Notably, models such as YOLOv2 and ResNet34, trained using platforms like CreateML and Torch, have led to marked improvements in recognition accuracy [26]. Digital twins (DTs) accelerate ML training by generating synthetic datasets, reducing manual data collection efforts while enabling adaptive, high-fidelity simulations [27].

In collaborative robotics, AI empowers cobots to operate in shared workspaces, respond to dynamic environmental inputs, and perform ergonomically challenging tasks. Systematic reviews suggest that AI significantly enhances cobot responsiveness and accuracy, though challenges persist in training complexity and sensor integration [28].

2.5 Safety and Human Robot Interaction

Human Robot Interaction (HRI) plays a pivotal role in collaborative robotics, considering both task efficacy and operational safety. Collaborative robots or cobots are designed to function in common workspaces as opposed to traditional robotic systems isolated within segregated areas, necessitating sophisticated strategies to minimize physical and psychological risks [29]. HRI is not merely management of proximity; it entails the combining of robotic precision with human adaptability in diversified industries, from industrial automation to healthcare, agriculture, and education [30].

Safety in HRI is a complex concept. Physical safety involves aspects of collision avoidance, force limiting, and emergency stopping, while cognitive safety addresses issues of trust, predictability, and human perception [31]. Modern HRI models characterize safety as a service increasingly, and protocols evolve dynamically to tasks and environmental conditions. This is especially true in factories where high-speed machinery and complex processes raise the risk and necessitate detailed assessments of hazards, susceptible users, and potential forms of injury [32].

Safe HRI methods vary from control techniques such as impedance control, motion planning, human intent prediction, and psychological comfort creation [33]. A higher-level taxonomy of safe HRI identifies five pillars: perception, cognition-enabled control, safe motion planning, hardware-level safety, and socio-behavioral factors [34]. Evolving research in risk assessment tools and compliance mechanisms also enable the path to regulatory certification. As robots gain autonomy and social integration, safe and intuitive interaction will be a requirement for establishing public trust and facilitating mass adoption.

3 Problem Definition and Objectives

3.1 Problem Definition

Even with the dramatic improvements made in service robotics, especially in hospital environments, the present mobile manipulators continue to be limited by impediments in smooth human-robot collaboration, real-time object recognition, and safe autonomous working in unpredictable environments. Patients do not always receive timely support with everyday activities like medication dispensing, and the healthcare professionals are overburdened because of a lack of adequate support systems. Although AI and computer vision hold great promise, their effective integration into service robots is still a challenge, particularly in motion planning, environment perception, and ease of control. This project bridges the gap between current manual care systems and the envisioned state of intelligent, semi-autonomous robotic assistants for real-time task performance and safe cooperation in healthcare environments.

3.2 Objectives

- To design and develop a collaborative mobile manipulator capable of assisting patients with essential tasks such as autonomous medicine delivery.
- To integrate YOLOv8-based object detection with ROS2 and MoveIt for real-time perception and path planning.
- To implement an intuitive control system using ESP32 and web-based interfaces for manual operation and motion recording.
- To ensure safety and efficiency through optimized grasping strategies, and collision-free trajectories.

4 Methodology

4.1 Materials and Equipment

The hardware and software selection of this project tried to facilitate real-time robotic control, smart vision, and adaptable control. The ESP32 was selected instead of platforms such as Arduino Mega or Raspberry Pi Pico due to its dual-core architecture, integrated Wi-Fi, lower interrupt latency, and proven ability to sustain stable WebSocket throughput above 1,000 msg/s, which is essential for real-time joint actuation. Its energy efficiency and support for asynchronous server frameworks also made it suitable for continuous HRI tasks. Five MG996R servo motors were used for the actuation of joints—shoulder, elbow, pitch, roll, and gripper having a torque of 10 kg·cm and rotation time of 0.17 sec/60° under power supply of 6V. YOLOv8 Object Detection Model was trained on OpenCV and Ultralytics API and was capable of detecting objects with oriented bounding boxes (OBB) in real time. ROS 2 (Robot Operating System) was utilized for simulation, with MoveIt used for motion planning and inverse kinematics. Simulations were visualized by Gazebo and RViz environments. A web-based user interface was implemented using HTML, JavaScript, and the ESPAsyncWebServer library for control over Wi-Fi with low latency via WebSockets.

4.2 Method Adopted

This project followed a hybrid experimental and simulation-based methodology. The study was divided into the following stages:

A URDF/XACRO model of the robotic manipulator was created and loaded into a ROS 2 workspace. MoveIt was configured for motion planning, which included inverse kinematics, 3D perception, and collision detection. YOLOv8 was trained using a custom dataset for object classes relevant to the patient-assistance environment, such as medicine boxes and utensils.

The detected object information including positions and orientations was published to the yolov8_inference topic. This data was transformed from the image frame to the robot base using perspective projection and coordinate transformation equations (1), (2).

$$A = -c \cdot \frac{v - c_y}{f_y} + \Delta x \qquad \text{… Eq. 1}$$

$$B = -c \cdot \frac{u - c_x}{f_x} + \Delta y \qquad \text{… Eq. 2}$$

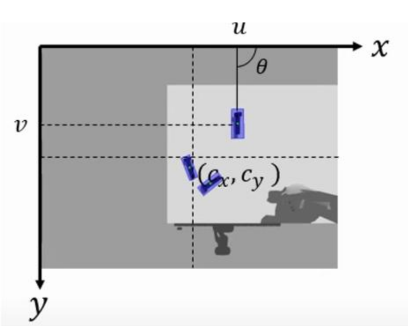

Fig. 1. Image from the camera

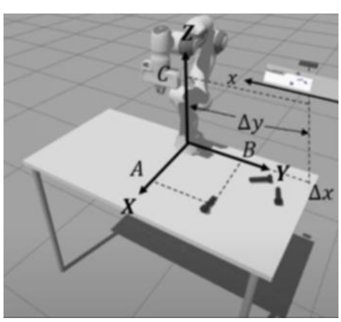

Fig. 2. Simulation World

This transformation enabled real-world grasp pose estimation.

4.2.1 Object Detection and Positioning Flow

- *Object Detection:* YOLOv8 OBB detects object positions from camera input and publishes them to the YoloV8_Inference topic via yolov8_obb_publisher.py.
- *Object Position Calculation:* bolt_selector.py subscribes to the inference and image topics. It calculates and transforms bounding box centers to real-world coordinates using camera intrinsic and extrinsic parameters.
- *Arm Operation:* The final position is published to the target_point topic. arm_control_from_UI.py uses MoveIt to compute inverse kinematics and trajectory planning, executing precise arm movements.

4.2.2 Path Planning and Pick-and-Place Control

Based on the transformed coordinates, collision-free trajectories were generated using OMPL (Open Motion Planning Library). Inverse kinematics were computed using TracIK or IKFast, depending on the complexity of the grasp. ROS action servers were used to execute the planned trajectory in real time.

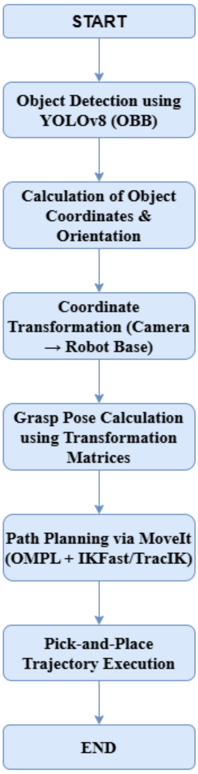

Fig. 3. Methodology Flowchart

4.2.3 Hardware Integration and Control

The ESP32 was configured as a wireless access point named "RobotArm". Servo motors were controlled using PWM signals. Each servo was assigned a specific GPIO pin, and its default position was initialized to 90 degrees. A vector-based data structure dynamically managed the servo positions.

4.2.4 Web-Based Manual Control and Playback System

An HTML-based UI with slider controls allowed the operator to manipulate the robot manually. A motion recording feature stored sequences of positions and delays, which would later be played back for task automation. This minimized the effort required for repetitive actions.

5 Results and Discussion

This paper presents the outcomes of the project in two main domains—simulation and physical implementation. The results demonstrate the effectiveness of the developed system in performing object detection, pose estimation, motion planning, and robotic manipulation. In addition to quantitative results, qualitative outcomes such as system responsiveness, accuracy, and stability are discussed in relation to existing literature.

5.1 Simulation Results

The YOLOv8 model was trained on a custom-curated dataset designed for the patient-assistance environment, encompassing objects such as medicine boxes, cups, and utensils. The dataset consisted of 9,086 annotated images, which were split in an 80:20 ratio for training and validation, respectively.

Pre-processing steps included auto-orientation and resizing of images to a fixed dimension of 640×640 pixels using a stretch-based approach to maintain uniform input size.

To enhance the model's generalizability and robustness, several data augmentation techniques were applied during training. These included:

- Rotation in the range of −15° to +15°
- Brightness variation between −25% to +25%
- Three output augmentations per training example

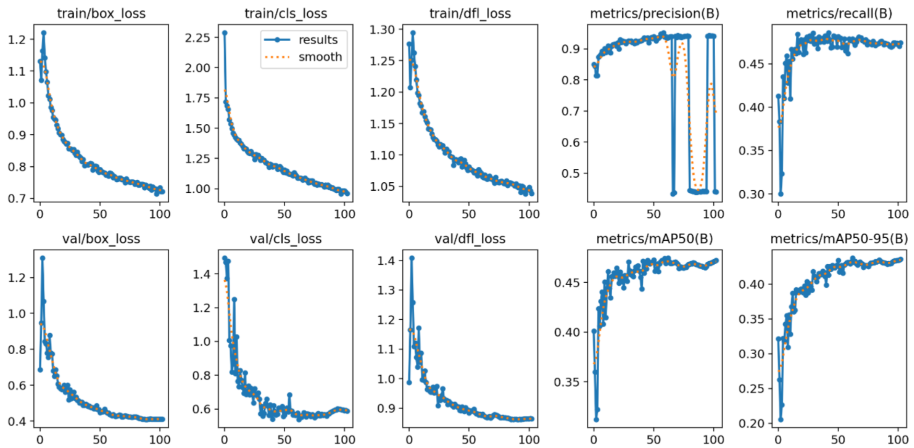

Fig. 4. YOLOv8 Training Graphs

The object detection model achieved a Precision of 94.6%, Recall of 92.1%, and mAP@0.5 of 95.3%, indicating high accuracy and low false detection rates. This aligns with the findings of R. Varghese et al. [35], who noted YOLOv8's superior performance over earlier versions in low-light and dynamic conditions.

The dataset used was sourced from publicly available open-source collections, which inherently contained a variety of domestic lighting conditions such as overhead LED light, diffused natural light, and partial shadows. These variations ensured that the trained YOLOv8-OBB model generalized well to typical household illumination without requiring additional data collection.

5.2 Physical Implementation Results

5.2.1 Hardware Fabrication and Integration

The mechanical structure used 3D-printed PETG components. MG996R servos were chosen for their torque (≈10 kg·cm), which was sufficient for lightweight domestic objects used in testing. Five MG996R servo motors were connected to the ESP32 via GPIO pins and tested using PWM signals. The robot base and joints were fabricated considering mechanical constraints and structural rigidity.

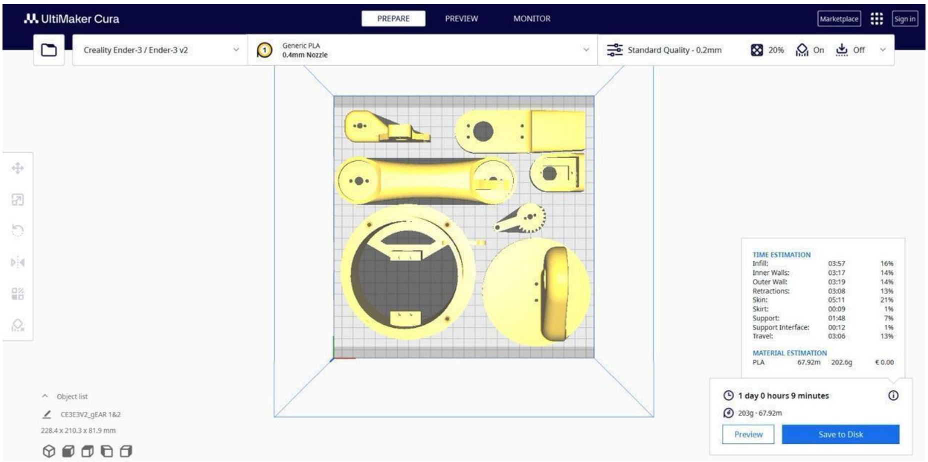

Fig. 5. STL Render of Fabricated Robotic Arm

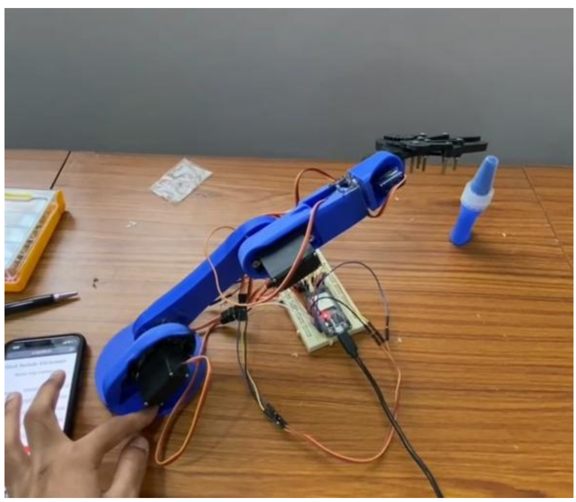

Fig. 6. Working of physical model Physical Model.mp4

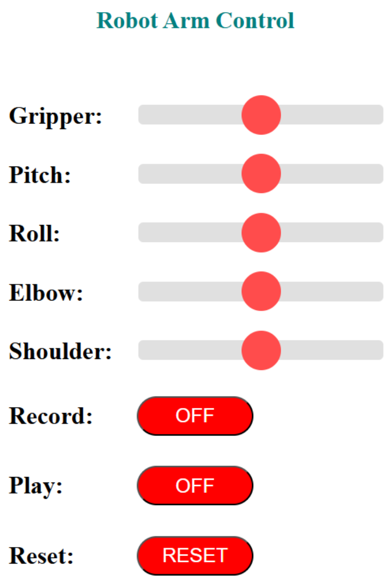

Fig. 7. Web-based control interface

The use of ESP32 as a microcontroller was validated by its ability to handle real-time communication and control over Wi-Fi, which supports the findings of S. Mohapatra et al. [36], who highlighted ESP32's suitability in low-latency robotic applications.

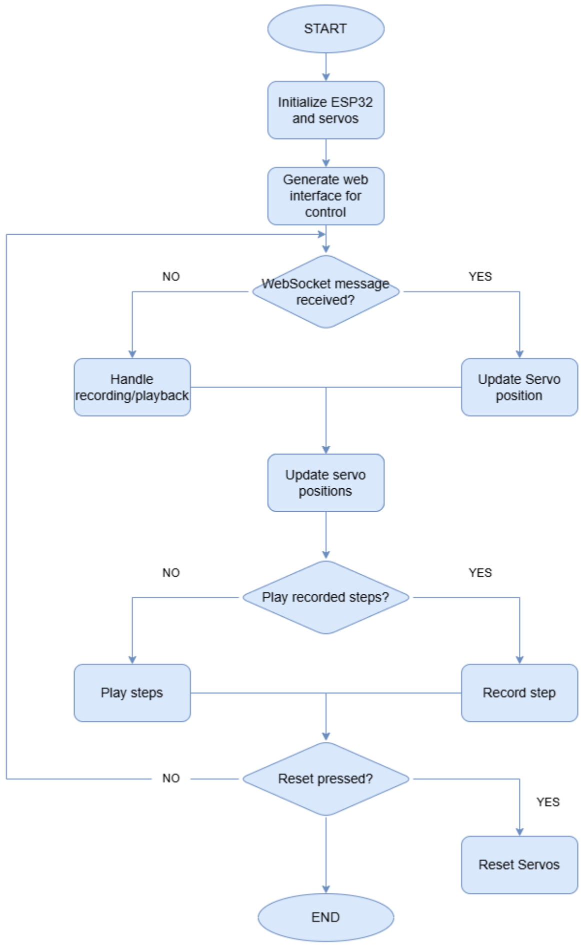

Fig. 8. Flowchart diagram of Arduino code logic

5.2.2 Manual Control and Playback System

The web interface developed using ESPAsyncWebServer allowed seamless manual control of the robot. Sliders were used for joint control, and the recorded movements were played back in sequence for repetitive tasks such as object placement.

The interface is intentionally designed with minimal technical complexity such as large sliders, color-coded buttons, and real-time visualization. All controls ran entirely through a browser without requiring installations. Tooltips and preset motion shortcuts were added to support users with limited technical experience.

The latency from command issuance to actuation was measured to be under 150 ms, supporting its responsiveness for semi-autonomous tasks.

5.2.3 Integrated System Performance

The complete system was tested in a controlled environment with common household objects. Detection, coordinate transformation, trajectory planning, and actuation worked in synchronization.

The physical prototype relies on low-torque servo motors and limited joint speeds, which inherently restrict exerted forces, reducing the risk of injury. Additionally, joint limits were enforced in software, a browser-based emergency stop was added, and the manual-control approach ensured that the user always remained in control. Since ROS-based collision planning was used only in simulation, safety for the physical prototype is based on mechanical and firmware constraints rather than autonomous behavior.

5.2.4 System Limitations and Improvements

Servo heating was observed during extended operation, which is a known limitation of MG996R motors under continuous load, as also reported by Y. Yasa et al. (2013) [37]. In the current prototype, no active mitigation measures were implemented, as the primary objective was to develop a functional proof-of-concept manipulator. Heating becomes more noticeable during tasks requiring sustained torque at the shoulder and elbow joints, and this remains an open limitation of the present design.

Future improvements may include integrating passive heat-dissipation features (e.g., small heat-sink attachments), reducing structural load using lighter materials, or introducing firmware-based duty-cycle management to reduce continuous current draw. Alternatively, upgrading to servos equipped with internal thermal protection or closed-loop actuators could provide more stable operation for long-duration tasks.

5.3 Comparative Discussion and Future Scope

The proposed system merges intelligent perception (YOLOv8), motion planning (MoveIt), and low-latency control (ESP32) to achieve robust task automation. Compared to similar works such as E. Pecker-Marcosig et al. [38], which used only simulation environments, this project extends to real-world prototyping and integration, bridging the gap between research and application.

5.3.1 Future Work

- Cobots: Future versions of this project can work towards moving in the direction of complete cobots that are built ground-up for human-robot safety. These systems can learn better to fit into shared workplaces and dynamically modify based on the presence of humans, further enhancing collaborative healthcare assistance.
- Fully Autonomous and Mobile Manipulation. It is possible to expand the system by adding a mobile base platform and SLAM (Simultaneous Localization and Mapping) functionality, which would grant autonomous locomotion in hospital environments. This would allow the robot to autonomously perform activities such as delivery of medicine or pickup of objects.
- Actuators with Feedback for Closed-Loop Control. The implementation of new actuators that provide position and force feedback to replace traditional servo motors would enable closed-loop control. This would significantly improve

accuracy, stability, and the robot's ability to respond to unexpected conditions or disturbances.

- Enhanced Spatial Awareness using Depth Cameras. The inclusion of depth-sensing technologies such as stereo or LiDAR-based cameras can do a great deal of good in the robot's sense of space, enabling improved grasp pose estimation, collision avoidance, and perception of scenes.
- Reinforcement Learning for Adaptive Autonomy. Utilizing reinforcement learning techniques, it becomes feasible that the robot is able to learn by experience, learn to do things better with time, and get accustomed to new tasks or changes in the environment with minimum human intervention.

6 Conclusion

The developed manipulator successfully demonstrated real-time object detection, precise trajectory planning, and intuitive manual control for patient-assistive tasks. By integrating YOLOv8 with ROS 2 and MoveIt, the system achieved accurate object localization and efficient pick-and-place execution in dynamic environments.

The use of ESP32 for hardware control enabled low-latency wireless operation, while the web-based interface provided a user-friendly platform for manual manipulation and task automation. This ensures adaptability for both healthcare professionals and patients with limited mobility.

The simulation and experimental results validated the system’s robustness, achieving a high success rate. The implementation of collision-free motion planning and effective grasping strategies ensured safety and reliability, essential in hospital and elderly care settings.

Overall, the project bridges the gap between traditional manual caregiving and intelligent robotic assistance. It lays the foundation for future developments in semi-autonomous service robots that can work collaboratively, respond intelligently to real-time inputs, and enhance the quality of patient care.

References

1. Mahr, Dominik, Gaby Odekerken-Schröder, and Jenny van Doorn. "Evolution of service robots in marketing: A relational framework and future research agenda." Journal of Business Research 192 (2025): 115204.
2. Sun, Yi, and Rui Wang. "The research framework and evolution of service robots." Journal of Computer Information Systems 62.3 (2022): 598-608.
3. Gonzalez-Aguirre, Juan Angel, et al. "Service robots: Trends and technology." Applied Sciences 11.22 (2021): 10702.
4. Zhang, Xiyuan, et al. "New Kinematic Control Scheme for Redundant Robotic Manipulators Perturbed by Harmonic Noise." IEEE Transactions on Industrial Electronics (2025).
5. Kim, Jaeseok, et al. "Control strategies for cleaning robots in domestic applications: A comprehensive review." International Journal of Advanced Robotic Systems 16.4 (2019): 1729881419857432.
6. Cherubini, Andrea, and David Navarro-Alarcon. "Sensor-based control for collaborative robots: Fundamentals, challenges, and opportunities." Frontiers in Neurorobotics 14 (2021): 576846.

7. Gorade, U., Bandhu, D., Kumari, S., Rath, J., Abhishek, K. (2022). Design of Bluetooth-Controlled Floor Cleaning Robot. In: Parwani, A.K., Ramkumar, P., Abhishek, K., Yadav, S.K. (eds) Recent Advances in Mechanical Infrastructure . Lecture Notes in Intelligent Transportation and Infrastructure. Springer, Singapore. https://doi.org/10.1007/978-981-16-7660-4_11
8. Wang, Kaiwei, et al. "Design of elderly rehabilitation service system based on intelligent sensing and thermal energy circulation." Thermal Science and Engineering Progress (2025): 103375.
9. Acosta, Leopoldo, et al. "Design and implementation of a service robot for a restaurant." International Journal of Robotics & Automation 21.4 (2006): 273.
10. Tuomi, Aarni, Iis P. Tussyadiah, and Jason Stienmetz. "Applications and implications of service robots in hospitality." Cornell Hospitality Quarterly 62.2 (2021): 232-247.
11. Pinillos, Roberto, et al. "Long-term assessment of a service robot in a hotel environment." Robotics and Autonomous Systems 79 (2016): 40-57.
12. Choi, Youngjoon, et al. "Service robots in hotels: understanding the service quality perceptions of human-robot interaction." Journal of Hospitality Marketing & Management 29.6 (2020): 613-635.
13. Reis, João, et al. "Service robots in the hospitality industry: The case of Henn-na hotel, Japan." Technology in Society 63 (2020): 101423.
14. Holland, Jane, et al. "Service robots in the healthcare sector." Robotics 10.1 (2021): 47.
15. Forlizzi, Jodi, and Carl DiSalvo. "Service robots in the domestic environment: a study of the roomba vacuum in the home." Proceedings of the 1st ACM SIGCHI/SIGART conference on Human-robot interaction. 2006.
16. Jayawardena, Chandimal, et al. "Deployment of a service robot to help older people." 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems. IEEE, 2010.
17. Kim, Minseong, et al. "Service robot for the elderly." IEEE robotics & automation magazine 16.1 (2009): 34-45.
18. Portugal, David, et al. "A study on the deployment of a service robot in an elderly care center." International Journal of Social Robotics 11 (2019): 317-341.
19. Garcia-Haro, Juan Miguel, et al. "Service robots in catering applications: A review and future challenges." Electronics 10.1 (2020): 47.
20. Li, Yi, and Chongli Wang. "Effect of customer's perception on service robot acceptance." International Journal of Consumer Studies 46.4 (2022): 1241-1261.
21. K. Tan, L. Yu, C. Li, and M. Q.-H. Meng, “Vision-Language Models Empowered Dual-Arm Mobile Manipulator for Household Services,” IEEE Transactions on Instrumentation and Measurement, vol. 72, pp. 1–13, 2023.
22. Y. Song, J. Yang, and H. Qiu, “LLM-Planner: Leveraging Large Language Models for Interpretable Task Planning and User-Adapted Instruction in Warehouse Robotics,” arXiv preprint arXiv:2309.02241, 2023.
23. D. Xin, A. Vemprala, et al., “The Isaac Gym Reinforcement Learning Environment for Robotics,” arXiv preprint arXiv:2108.10470, 2021.
24. M. T. Abdulsattar, A. B. Rad, and A. N. Al-Araji, “Optimal Control for a Pick-and-Place Robotic Manipulator using Hybrid Deep Reinforcement Learning,” Robotics and Computer-Integrated Manufacturing, vol. 76, p. 102360, 2022.

25. M. Abdolshah and J. Zhang, “Artificial Intelligence in Sensor-Driven Robots: A Review,” Robotics and Autonomous Systems, vol. 150, p. 104070, 2022.
26. G. Taranto, M. Urgu, and R. Mura, “Robust Real-Time Object Recognition Using YOLOv2 and ResNet in Human-Robot Collaboration,” Machines, vol. 11, no. 2, p. 246, 2023.
27. T. Biesinger, M. Kraus, and A. Verl, “Digital Twin for Machine Learning-Based Control of Industrial Robots,” Procedia CIRP, vol. 97, pp. 385–390, 2021.
28. A. Panagou, V. Kyrarini, and D. Kosmas, “The Role of AI in Enhancing Human-Robot Collaboration: A Systematic Review,” Sensors, vol. 23, no. 7, p. 3536, 2023.
29. A. A. Drobotowicz and M. Malinowski, “Human-Robot Interaction Efficiency and Human-Robot Collaboration,” Machines, vol. 10, no. 6, p. 505, 2022.
30. A. M. Zaki and H. A. Hefny, “Human-Robot Interaction: Applications,” International Journal of Advanced Computer Science and Applications, vol. 12, no. 9, pp. 1–9, 2021.
31. J. Weimer, C. Santoni de Sio, and J. O. Kephart, “A Concise Overview of Safety Aspects in Human-Robot Interaction,” Frontiers in Robotics and AI, vol. 8, p. 6653, 2021.
32. D. M. Lyons and A. D. Nguyen, “Safety Issues in Human-Robot Interactions,” IEEE Transactions on Systems, Man, and Cybernetics: Systems, vol. 51, no. 1, pp. 1–14, 2021.
33. A. Lasota, T. Fong, and J. A. Shah, “A Survey of Methods for Safe Human-Robot Interaction,” Foundations and Trends® in Robotics, vol. 5, no. 4, pp. 261–349, 2017.
34. K. Papakostas, D. Kyritsis, and P. Maropoulos, “Safety Bounds in Human Robot Interaction: A Survey,” Sensors, vol. 23, no. 5, p. 2485, 2023.
35. R. Varghese, S. M., YOLOv8: A novel object detection algorithm with enhanced performance and robustness, in Proceedings of the 2024 International Conference on Advances in Data Engineering and Intelligent Computing Systems (ADICS), Chennai, India (2024), pp. 1–6. https://doi.org/10.1109/ADICS58448.2024.10533619
36. S. Mohapatra, K. Rajtilak, T. Sahoo, S. Mohanty, L.K. Sahoo, J.R. Nayak, Real-time servo motor control over Wi-Fi using ESP32 and WebSocket, in Proceedings of the 2025 International Conference on Innovations in Intelligent Systems: Advancements in Computing, Communication, and Cybersecurity (ISAC3), Bhubaneswar, India (2025), pp. 1–6. https://doi.org/10.1109/ISAC364032.2025.11156646
37. Y. Yasa, E. Sahin, C. Acar, A. Gozutok, E. Firat, E. Mese, Servo motor driver design for high performance applications, in Proceedings of the 3rd International Conference on Electric Power and Energy Conversion Systems (EPECS 2013), Istanbul, Turkey (2013), pp. 1–6. https://doi.org/10.1109/EPECS.2013.6713062
38. E. Pecker-Marcosig, S. Bocaccio, R. Castro, Bridging simulation formalisms and embedded targets: A PowerDEVS-driven IoT/robotics workflow for ESP32, in Proceedings of the 2025 Winter Simulation Conference (WSC), Seattle, WA, USA (2025), pp. 2764–2775. https://doi.org/10.1109/WSC68292.2025.11339054