

Enhancing Industrial Automation: An Adaptive ROS-Controlled Robotic Arm for Diverse Tasks

Ayush Ajmera^{1,*}, Deepti Tupake², Ritesh Shambharkar³, Tejas Dherange⁴, R.A. Kapgate⁵

Abstract

In robotics and industrial automation, there is a growing need for robotic systems that can work on their own and do tasks quickly. This research is about creating an Intelligent Robotic Arm that uses the Robot Operating System and works with a UR3 platform and an OAK-D depth camera for better vision. The Robotic Arm uses the Robot Operating System as its software, which helps it talk to its parts, move accurately, and do tasks well. With computer vision and depth sensing, the Robotic Arm can find objects, know where they are and do tasks with accuracy. The Robotic Arm can work in changing environments. Still be stable and accurate. It can do industrial tasks like picking and placing things, sorting objects, and simple assembly. The Robotic Arm uses motion planning algorithms and sensor feedback to adjust its movements in time, making it safe and efficient. The system was tested through simulation and practical testing. The results show that the Robotic Arm is scalable and adaptable, making it suitable for use in manufacturing, healthcare, logistics, and warehouse automation. The Robotic Arm can be used in various areas, and its ability to work well in different situations makes it a valuable tool. The Robot Operating System and OAK-D camera make the Robotic Arm smart and capable. The UR3 robotic platform and computer vision help the Robotic Arm do tasks with precision. The Robotic Arm is a system that can help in many industries.

Keywords: ROS, robotic arm, UR3, OAK-D camera, motion planning, industrial automation

INTRODUCTION

Industrial automation has been revolutionized by the arrival of robotic arms, which have become essential for enhancing accuracy, scalability, and productivity across various sectors. The incorporation of advanced technologies, such as real-time sensors, computer vision, and the Robot Operating System (ROS) [1], has greatly expanded the capabilities and applications of these systems. Nevertheless, conventional robotic systems continue to struggle with adapting to unpredictable and dynamic environments, particularly in areas such as motion planning, object identification, and instantaneous decision-making. Robotic arms have improved productivity in several industries while also increasing precision and extendibility.

*Author for Correspondence

Ayush Ajmera
E-mail: ayushajmera003@gmail.com

¹⁻²Student, Department of Mechatronics Engineering, Sanjivani College of Engineering, Kopergaon, Maharashtra, India

⁵Professor, Department of Mechatronics Engineering, Sanjivani College of Engineering, Kopergaon, Maharashtra, India

Received Date: January 19, 2026

Accepted Date: March 05, 2026

Published Date: March 10, 2026

Citation: Ayush Ajmera, Deepti Tupake, Ritesh Shambharkar, Tejas Dherange, R.A. Kapgate. Enhancing Industrial Automation: An Adaptive ROS-Controlled Robotic Arm for Diverse Tasks. Journal of Advancements in Robotics. 2026; 13(2):22–32p.

Their incorporation alongside modern technologies, such as live sensors, computer vision, and the Robot Operating System from the Open Source Robotics Foundation, has expanded the capabilities and applications of these platforms. Nonetheless, standard robotic frameworks continue to struggle in more dynamic and unpredictable environments. Movement strategies, real-time object identification, swift decision-making, and multi-layered heuristic algorithms all pose challenges. Furthermore, the most recent advancements in machine learning have enabled the automation of robotic processes capable of self-

improvement and real-time adaptation. This opens up new applications to accomplish comprehensive design and verification procedures for accurate collision-free motion segmentation and trajectory carving.

The project uses simulation tools from MoveIt [2], RViz, and Gazebo. The system built from the prototype testing phase was able to perform active tasks such as sorting and assembly with high precision and adaptability. The goal of this project is to develop a novel, versatile, and robust automation framework for robotic systems used in industrial, medical, and storage facilities in Japan.

RELATED WORK

The use of robotic arms in healthcare, warehousing, and industrial automation has been extensively studied. Research on the Robot Operating System (ROS) has reached new heights with the development of modular and scalable robotic systems. Research on flexible and precise robotics, including motion planning and vision-based object detection, has been conducted. Aragon-Camarasa [3], Audonnet [3], and Hamilton [3] provided significant robotic arm manipulation techniques. These techniques can be emulated in the ROS framework and other ecosystem simulations using OAK-D smart cameras, which provide situational awareness to the robots so that they can update their maps dynamically. Mokhtarzadeh [4], Xiao [4], and Han [4] also developed a unique mapper that enhanced indoor robot localization and navigation. These studies highlight new difficulties in synchronizing motion planning with vision systems at runtime, while also demonstrating progress in deploying vision systems to improve modularity and adaptability [5].

This study attempts to provide a solution, such as a system with improved robotics. ROS, an open robotic system integration platform, was introduced in 2009, suggesting a substantial development in modular, adaptable robotic systems. By placing a UR3 robotic arm under ROS control and proving the viability of the system, Muller [6] and Friedl [6] (2016) focused on automating industrial processes (2012) and demonstrated how to execute complicated kinematic maneuvering within the ROS limitations of a motion planning library that has been approved by MoveIt.

Vision-based object detection systems are considered differently, and two theoretical branches are linked to them. J. Redmon [7] et al. (2016) increased mechanical capabilities with YOLO real-time object identification and further improved the SSD system's visual information processing accuracy and efficiency. Kragic [8] investigated the visual servoing paradigm to enable robotic operation in dynamic conditions, and the coupling of these robotic systems with arms has been intensively investigated.

CIR poses a challenge in human-robot interaction for the safe automation of collaborative industrial robots. Wurman et al. (2008) [9] focused on the capabilities of cardinal robotic arms in terms of operational efficiency, flexibility, accuracy augmentation, and warehouse automation. Advances in depth-sensing technologies have led to further advancements in robotic systems. applicability of the OAK-D Camera for real-time object detection and situational awareness in autonomous systems. The effectiveness of the system by integrating the OAK-D Camera into indoor mobility robots for navigation and localization in changing conditions [10].

COMPONENT SELECTION

Tables 1 and 2 present the selected hardware components and software tools used to develop the robotic pick-and-place system. The hardware provides precise motion control, sensing, power management, and processing capabilities, whereas the software stack enables robot communication, motion planning, visualization, programming, debugging, and seamless system integration for reliable autonomous operation.

Table 1. Hardware component.

Component	Image	Description	Role in system
NEMA 17 stepper motor (42Ncm)		HIGH-torque stepper motor with fine rotational control	DRIVES robotic hand for precise pick-and-place motion
Minebea PM35L-048		COMPACT, high-precision stepper motor	Controls motion of the gripper
24 V 350 W power supply		Industrial-grade power supply unit	Powers motors and control electronics
Esp-32		Microcontroller with Wi-Fi/Bluetooth and high performance	Main processing unit managing control logic.
Oak-d camera		Depth-sensing camera with on-board AI using Intel Myriad X VPS	Detects objects and surroundings for manipulation tasks

Table 2. Software stack

Layer	Tool/Framework	Version/Notes	Role in system
Middleware	Ros 2	Humble hawkbill	Core communication between nodes (pub/sub, actions, services)
Motion planning	MoveIt setup assistant	Integrated with ROS 2	Configures robot model, planning groups, and control interfaces
Visualization	Rviz 2	Ros 2-compatible	3D visualization of robot state and environment
Operating system	Ubuntu	22.04 lts	Linux OS hosting ROS 2 and development tools
Ide	Visual Studio Code	With ROS 2 extensions	Writing, debugging, and managing code and launch files

RESEARCH METHODOLOGY

The adopted method for this investigation guarantees precision in the design and verification steps of the Shrewdly ROS (Robot Operating System)-Powered Mechanical Arm. Each step is planned to achieve optimal performance, adaptability, and agility in real-world environments. Figure 1 shows a rudimentary block diagram of our arm (Figure 1).

Mechanical Design

The design started from the extension of the basic angle of the armrest position on the UR3 mechanical arm. The arm was outlined in SolidWorks, where every component of the base bears upper

arm, elbow, and wrist was modelled so that each would integrate accurately and function correctly. The design was optimized for strength-to-weight ratio so that the tasks would be performed without breaking the arm structure. The components were manufactured using a Creality 3D printer because it is cheap and light, with PLA as the main fabric. To validate the design, pus analysis and kinematic animation were conducted in SolidWorks for a large range of load scenarios.

Equipment Integration

Equipment integration included combining advanced sensors, actuators, and controllers to form a cohesive framework.

1. *UR3*: Automated Arm Planned for lightweight applications capable of exact movement with a payload capacity of 3 kg.
2. *OAK-D*: Camera-empowered real-time question discovery profundity detecting and spatial mindfulness significant for energetic situations.
3. *ESP-32*: Microcontroller that provides consistent communication between equipment components and the ROS system, guaranteeing real-time responsiveness.

Extra components, such as NEMA stepper engine step-down converters and custom-built end effectors, were chosen based on execution necessities and operational limitations (Figure 2).

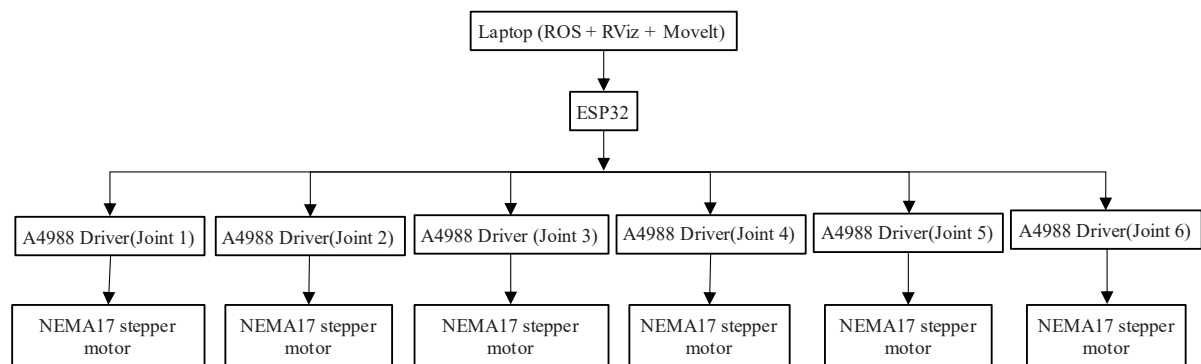


Figure 1. Block representation

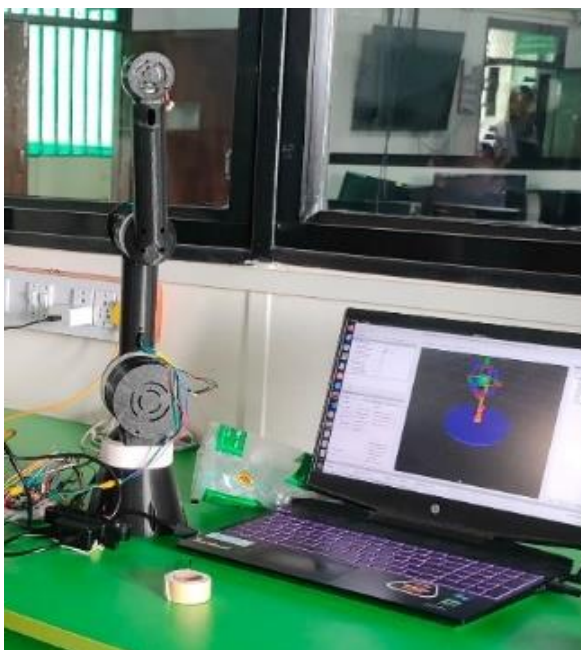


Figure 2. Arm integration

Program System

The program design was centered on ROS Noetic, chosen for its measured quality, adaptability, and strong library support. The workflow included:

1. *URDF Integration*: The mechanical arm plan was exported from SolidWorks into the Robot Description Format URDF to characterize its kinematic structure.
2. *MoveIt*: System Utilized for movement arrangement empowering collision avoidance and inverse kinematics computation for complex tasks.
3. *RViz*: Utilized for real-time visualization of the automated arm developments, guaranteeing precision in direction execution.

System Workflow

The implementation followed a structured workflow:

1. *Design and fabrication*: The automated arm was designed using SolidWorks and manufactured using a 3D printer. It was assembled with exact tolerances to guarantee smooth operation.
2. *Software setup*: The URDF record was imported into MoveIt for kinematic arrangement and movement, and ROS nodes were created in Python and C for hardware-software communication.
3. *Vision integration*: The OAK-D Camera was designed using the Profundity AI SDK to supply protest location and depth data, empowering energetic errand.
4. *Testing and validation*: The framework was tested in Gazebo for initial approval, followed by real-world testing to assess its execution in pick-and-place sorting and gathering tasks. Alterations were made based on test results to optimize framework execution (Figure 3).

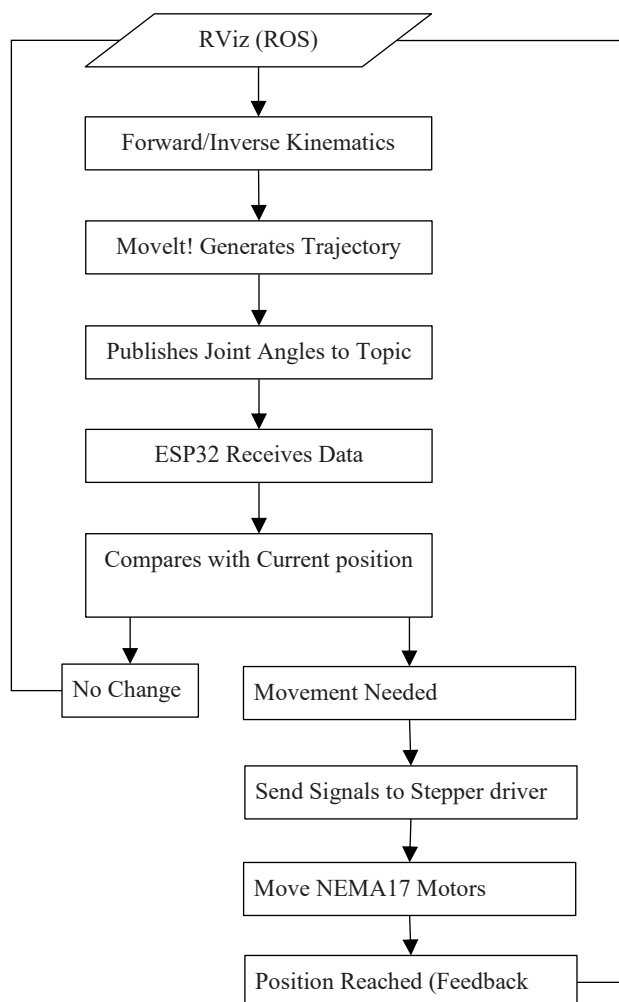


Figure 3. Workflow

Key Development

1. *Dynamic adaptability*: The integration of the OAK-D Camera permitted the framework to adjust to changes in question estimate introduction and area in real time, based on measured quality.
2. *Modularity*: The equipment and computer program plan guarantee simple adaptability, empowering the expansion of unused sensors or end effectors for expanded applications.
3. *Productivity*: ROS-based movement arrangement and real-time communication minimized computational delays, improving the framework's responsiveness and precision.

This comprehensive technique guarantees the fruitful advancement of a strong, versatile, and effective mechanical arm that can address the challenges of present-day mechanical mechanization.

RESULT AND DISCUSSION

Simulation Result

The simulation helped confirm the proposed strategy with the mechanical arm, marked the robotic arm movement in RViz, checked the kinematic model and planned trajectory, and verified the arm motion in dynamic environments through RViz visualization. The system repeatedly attained a collision-free path using the OAK-D Camera, and object recognition was performed efficiently. The data sent to the ROS accurately allowed movement alterations to be executed. The simulations validated the proposed conditions of the workspace for the mechanical arm, proving that it could achieve precision and efficiency within the defined parameters (Figures 4 and 5).

Real-World Results

Within the constraints of accuracy and reproducibility, the robotic system performed admirably in the real world. The system demonstrated the ability to reliably control the hardware modules under normal operating conditions after the control directives recorded from the simulation were translated into execution. Although several movement precision issues were found, they were caused by real-world factors such as mechanical constraints, external friction, and sensor noise interference.

However, the system's performance remained satisfactory (Figure 6).

Model Testing

Prototype testing was conducted to assess the framework's execution in real-world scenarios such as:

1. *Pick and place operation*: The mechanical arm was tested on tasks such as pick-and-place operations. Objects of diverse sizes, weights, and textures were effectively identified and handled. The OAK-D Camera provided precise depth and position information, enabling the consistent execution of these operations.
2. *Object sorting tasks*: Sorting Errands. The arm classifies and sorts objects based on predefined criteria, illustrating its versatility in warehousing and mechanical situations.
3. *Real-time adaptability*: The arm reacts to changes in object positions in real time, displaying its adaptability in dynamic conditions (Figure 7).

Test Result

Table 3. System result.

Metric	Result	Description
Positioning accuracy	95% success rate	Objects placed within required tolerance
Task time	3.2 seconds per task	Includes moving from one place to another
Cycle rate	~18.75 tasks per minute	Calculated from average task time (1 / 3.2s)
System adaptability	High	Modular design allows quick reconfiguration for new tasks

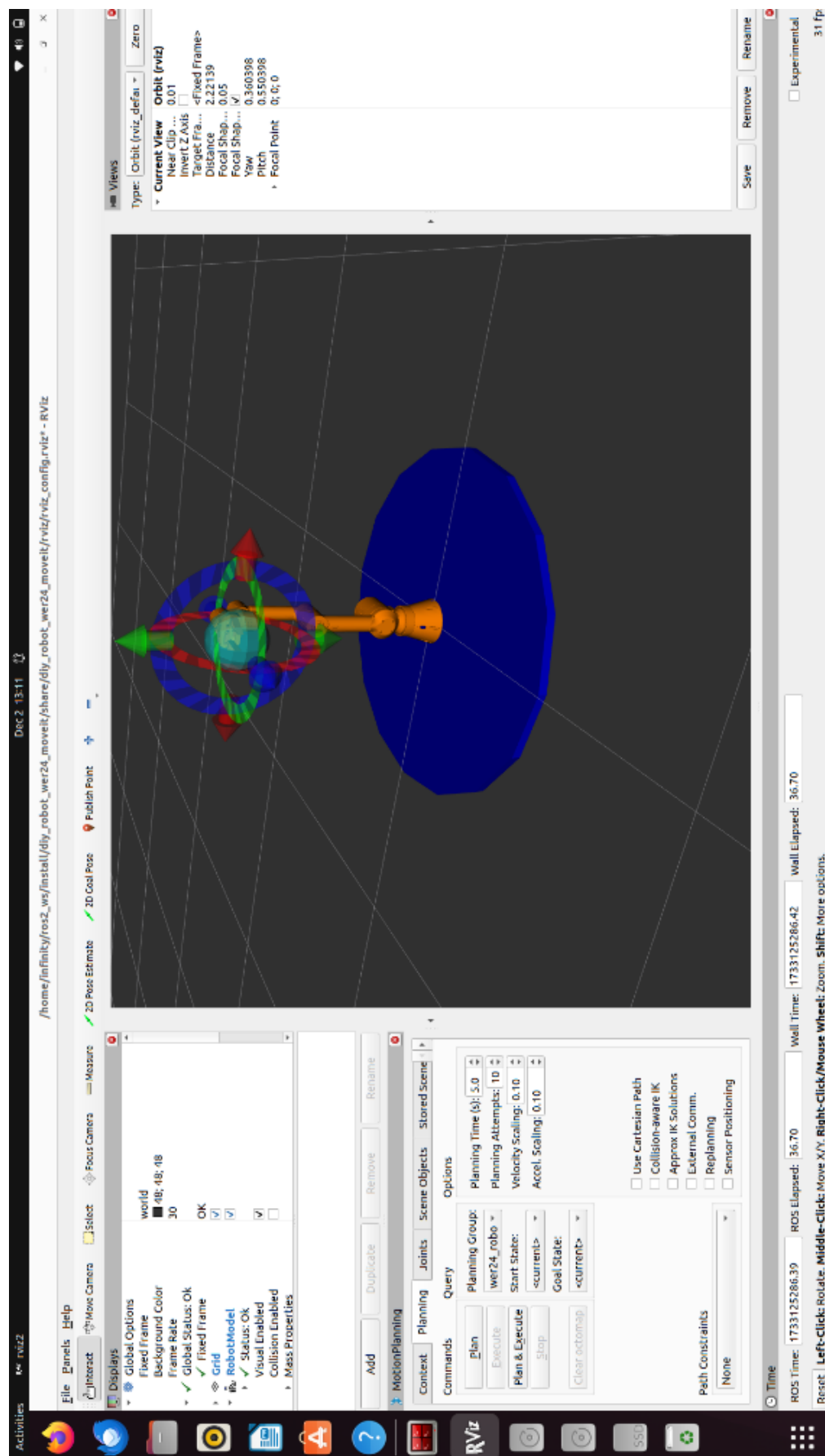


Figure 4. Path planning visualization 1

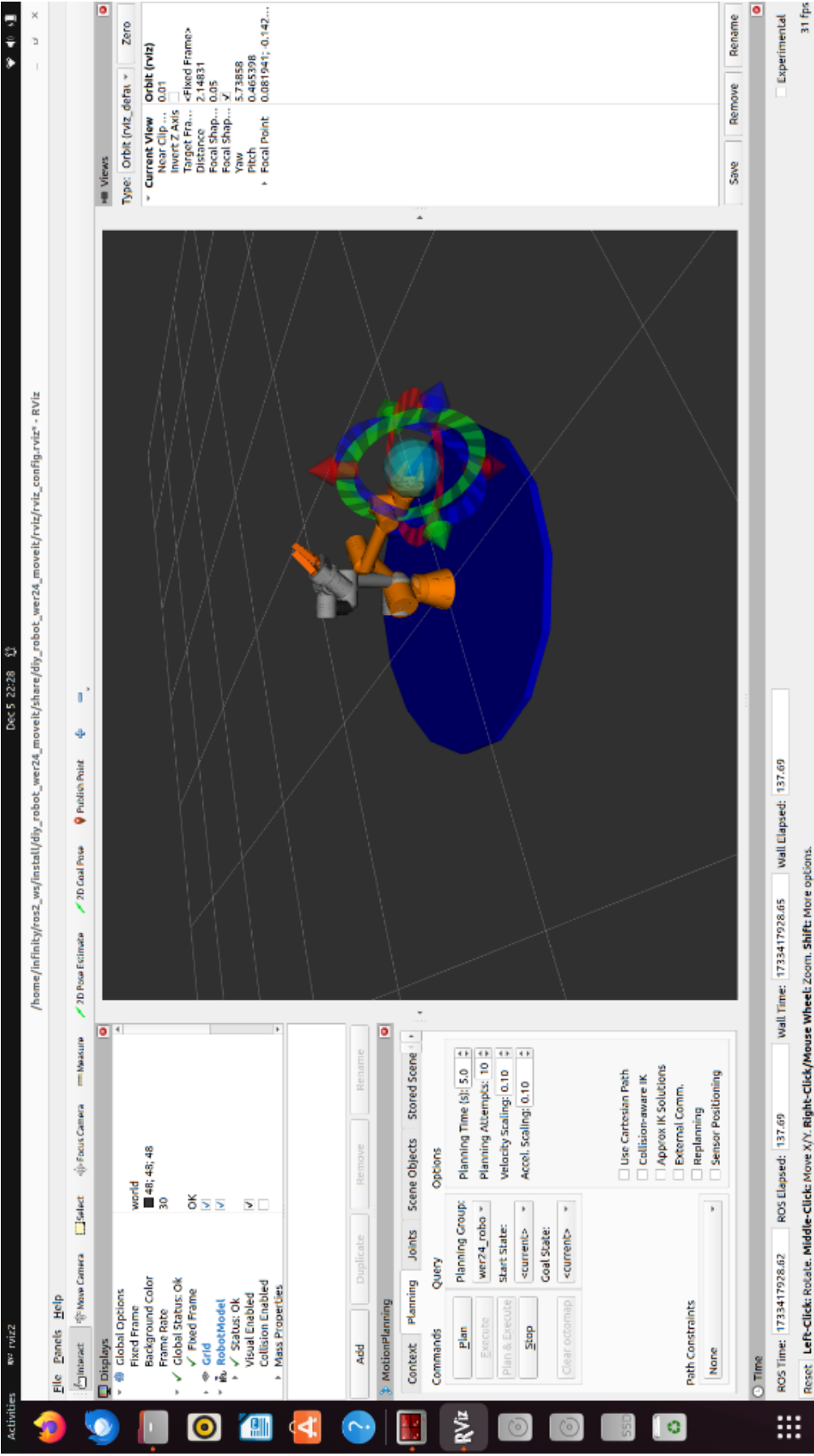


Figure 5. Path planning visualization 2



Figure 6. Real-world execution of path

1. *Positioning accuracy:* precision of the framework achieved a 95-victory rate in situating objects within the required tolerance.
2. *Operational cycle time:* Cycle time for each assignment counting pick-and-place and sorting, was completed within a normal of 3.2 s.
3. *Cycle rate:* Based on the task time, the system operated at approximately 18.75 tasks/min.
4. *System adaptability:* The modular plan permitted the expansion of unused end effectors, empowering the arm to handle different assignments with negligible reconfiguration (Table 3).

Comparative Analysis

The proposed framework was tested against previously implemented mechanical systems, and the use of ROS in conjunction with the OAK-D Camera provided remarkable benefits in terms of flexibility and precision. Unlike most conventional frameworks, which tend to depend on sequential command following and reactive modalities, this system demonstrated a remarkable ability to function in unstructured environments, enhancing its applicability to active robotic systems.

Limitation

Despite its qualities, the framework displayed certain limitations:

1. *Payload restrictions:* The arms' 3 kg payload capacity limits its application in heavy-duty assignments.
2. *Environmental dependencies:* The execution of OAK-D cameras is affected by lighting conditions, which may influence object discovery precision in poorly lit situations.
3. *Computational delays:* In scenarios with fast natural changes, slight delays were observed in handling and executing movement alterations.

Future Improvement

Future emphasis of the framework will be centered on:

1. *Increased payload capacity:* Improving the payload capacity by utilizing more grounded materials and actuators.
2. *Enhanced vision system reliability:* Moving forward, the strength of the vision framework is to perform dependably beneath shifting lighting conditions.
3. *Optimized computational performance:* Computational execution by leveraging quicker processors and more effective calculations.

This tells about, and the dialogue highlights the framework's capacity to address modern challenges in mechanical robotization while recognizing roads to encourage improvement and optimization.

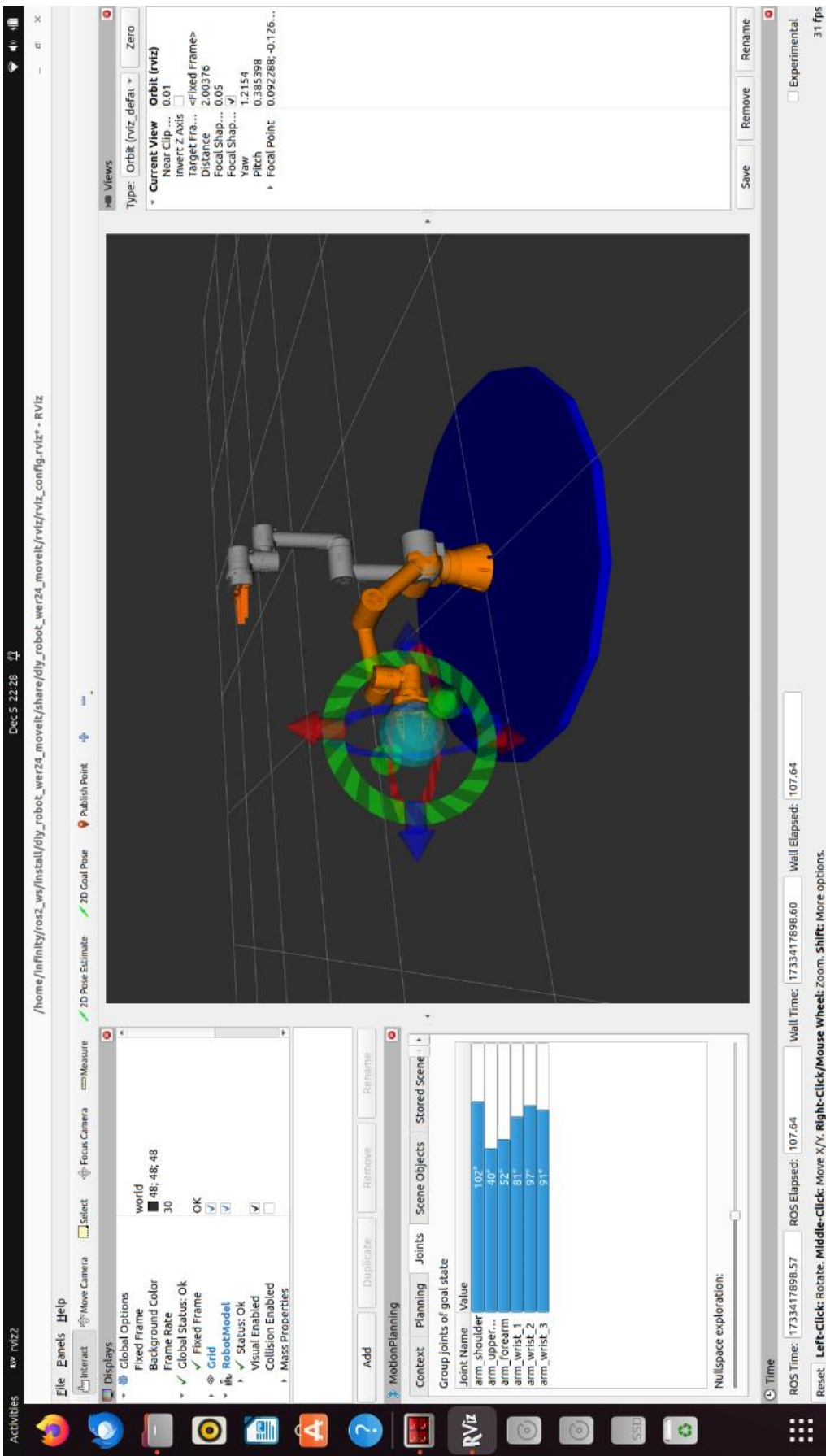


Figure 7. Arm testing by Rviz.

CONCLUSION

The Shrewdly ROS Powered Automated Arm marked a notable achievement in automation by integrating ROS control with advanced vision systems and solving fundamental problems in adaptability, flexibility, and precision. Strict methods for creating robots, system integration, detailed modelling, and model testing highlighted the system's ability to conduct sophisticated jobs, such as pick-and-place operations, sorting, and real-time modifications.

With the incorporation of the OAK-D Camera, depth perception and object recognition improved, allowing the robotic arm to perform real-time adaptations. Simulations established the system's capabilities for motion planning, while actual tests validated its usability for testing throughout diverse scenarios, even with some restrictions, such as payload constraints, sensitivity to background light, and other environmental factors. The modular and flexible architecture of the system offers a solid foundation for further modifications and broader applications. This study demonstrates that ROS and experienced vision systems can be coupled into multifunctional robotic platforms for logistics, manufacturing, and healthcare applications. Future research will focus on widening the advanced lasting utility range of operations, lowering the algorithmic complexity, and enhancing the resistance to environmental circumstances. This project aims to develop new methods for strong and versatile robotic systems suitable for practical use.

REFERENCES

1. Quigley M, Gerkey B, Conley K, Faust J, Foote T, Leibs J, et al. ROS: an open-source robot operating system. In: Proceedings of the IEEE International Conference on Robotics and Automation (ICRA) Workshop on Open Source Software; 2009 May 12–17; Kobe, Japan. p. 1–6.
2. Chitta S, Sucas I, Cousins S. MoveIt! An open-source software for kinematics, motion planning, and control. IEEE Robot Autom Mag. 2012;19(1):40–50. doi:10.1109/MRA.2011.2181748.
3. Audonnet FP, Hamilton A, Aragon-Camarasa G. A systematic comparison of simulation software for robotic arm manipulation using ROS2. In: Proceedings of the 22nd International Conference on Control, Automation and Systems (ICCAS); 2022 Nov 27–30; Busan, South Korea. Piscataway (NJ): IEEE; 2022. p. 755–762. doi:10.23919/ICCAS55662.2022.10003832.
4. Han Y, Mokhtarzadeh AA, Xiao S. Novel cartographer using an OAK-D smart camera for indoor robots location and navigation. J Phys Conf Ser. 2023;2467(1):012029. doi:10.1088/1742-6596/2467/1/012029.
5. Moreno HA, Carrera IG, Ramírez-Mendoza RA, Baca J, Banfield IA, editors. Advances in automation and robotics research: proceedings of the 3rd Latin American Congress on Automation and Robotics, Monterrey, Mexico 2021. 1st ed. Cham: Springer; 2022. (Lecture Notes in Networks and Systems; vol. 347). doi:10.1007/978-3-030-90033-5.
6. Müller A, Friedl L. Implementation of a UR3 robot for automated tasks in an industrial setup using ROS. IEEE Robot Autom Mag. 2016;23(3):50–57.
7. Redmon J, Divvala S, Girshick R, Farhadi A. You only look once: unified, real-time object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR); 2016 Jun 27–30; Las Vegas, NV, USA. Piscataway (NJ): IEEE; 2016. p. 779–788. doi:10.1109/CVPR.2016.91.
8. Kragic D, Christensen HI. Survey on visual servoing for manipulation. Stockholm (Sweden): Centre for Autonomous Systems, Numerical Analysis and Computer Science, Royal Institute of Technology (KTH); 2002.
9. Wurman PR, D'Andrea R, Mountz M. Coordinating hundreds of cooperative, autonomous vehicles in warehouses. AI Mag. 2008;29(1):9–20. doi:10.1609/aimag.v29i1.2082.
10. Tomić T, Petrovic P. Advances in collaborative robotics: challenges and opportunities for industrial applications. Procedia CIRP. 2020;91:32–37.