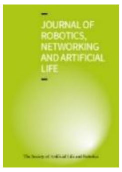
Journal of Robotics, Networking and Artificial Life
Vol. 11(3); March (2026), pp. 185-189
ISSN (Online): 2352-6386; ISSN (Print): 2405-9021
https://alife-robotics.org/jrnal.html



Research Article

The Research of an Augmented Reality System for the Implementation of Industrial Robots

Takuma Aiko, Eiji Hayashi, Gamolped Prem
Kyushu Institute of Technology, 680-4 Kawazu, Iizuka-city, Fukuoka, 820-8502, Japan
Email: aiko.takuma723@mail.kyutech.jp, haya@ics.kuytech.ac.jp, gamolped.prem129@mail.kyutech.jp
*Corresponding Author

ARTICLE INFO

Article History

Received 29 November 2024
Accepted 28 November 2025

Keywords

AR
Factory Automation Robots
Unity
ROS

ABSTRACT

The implementation of industrial robots is a critical strategy for mitigating the severe labor shortages affecting all industries in Japan. A significant barrier to their widespread adoption is the substantial capital investment required for the hardware and system integration. This study addresses this challenge through the development of a mobile Augmented Reality (AR) application designed to facilitate the robot adoption process. We have engineered and validated a system that enables the visualization and verification of a robot's operational trajectory prior to its physical installation, confirming its efficacy.

1. Introduction

In recent years, Japan has been facing a serious labor shortage in all industries. The main reason for this is the aging population combined with the diminishing number of children, and the consequent decrease in the working population. Progressive labor shortages will lead to a deteriorating working environment and a decline in product quality.

Given this background, the introduction of robots is attracting attention to achieve labor savings, manpower reduction, and productivity improvement. Indeed, the implementation of industrial robots offers numerous benefits, including task automation, quality stabilization, and improved safety. Our laboratory is also developing autonomous Japanese food packaging for bento box aimed at addressing the labor shortage (Fig. 1). On the other hand, many companies are cautious about introducing robots because the initial investment and operation costs are extremely high. These costs include the robot itself, related equipment, installation, and maintenance. Especially for small and medium-sized Enterprises (SMEs), the prospect of high costs and a lack of expert human resources are major barriers.

In addition, from the perspective of the Sustainable Development Goals (SDGs), there is an international need to improve productivity through the introduction of robotic technology in the prepared food industry.

In response to these challenges, our laboratory is developing autonomous Japanese food packaging technology for bento box.[1] [2] We are also utilizing augmented reality (AR) technology to help reduce the cost of introducing robots.[3] [4]

AR is a technology that superimposes virtual information onto real-world spaces. Beyond entertainment, it is being used in a wide range of fields, including education and manufacturing. In this research, we developed a mobile AR system that links Unity and ROS. This system allows us to display a 3Dmodel of a robot arm in real-world space and simulate its movements.

While Unity is generally known as a game development platform, it is also attracting attention as a simulation environment due to its high-precision physics simulation, real-time rendering, rich assets and ease of GUI development. The Robot Operating System (ROS) also dominates the market for robot software development.

This system makes it possible to check the flow line and placement of the robot and verify its operation without physically bringing the robot into the factory. This is expected to significantly reduce the cost and time required for system integration and lower the hurdles to introduction.

This system also enables visualization of the operation during teaching, which is effective for maintenance and adjustment after installation. Furthermore, the application developed in this research is cross-platform, meaning it can be operated on Android and iOS devices, which enhances its practicality. In the future, we expect the application to become an even more versatile and convenient AR support tool. This will be achieved by supporting multiple types of robot models and improving the user interface.

This system makes it possible to check the flow line and placement of the robot and verify its operation without physically bringing the robot into the factory. This is expected to significantly reduce the cost and time required for system integration and lower the hurdles to introduction.

This system also enables visualization of the operation during teaching, which is effective for maintenance and adjustment after installation. Furthermore, the application developed in this research is cross-platform, meaning it can be operated on Android and iOS devices, which enhances its practicality. In the future, we expect the application to become an even more versatile and convenient AR support tool. This will be achieved by supporting multiple types of robot models and improving the user interface.

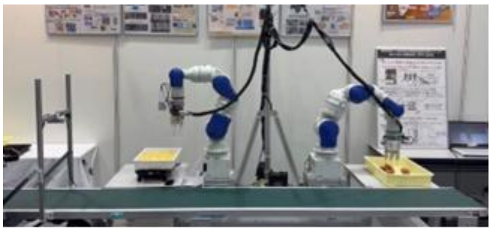
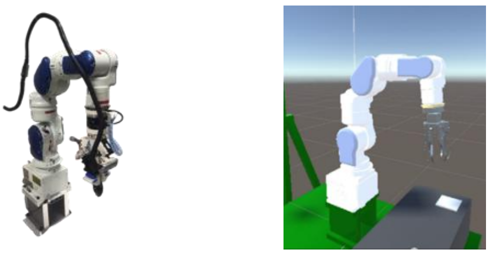


Fig.1. A self-learning robot for the food industry

2. Methodology

2.1. Robot

Fig. 2 shows the appearance of the robotic arm introduced in the developed mobile AR system. Specifically, the YASKAWA MOTOMAN-SIA5F, manufactured by Yaskawa Electric Corporation, was adopted. This robotic arm possesses seven rotational axes, enabling it to mimic the complex movement capabilities of a human arm. This significantly contributes to the realization of highly flexible tasks, such as precise operations while avoiding obstacles or dexterous manipulations in confined spaces. The robotic arm virtually displayed within our AR system adopts the URDF format, which eliminates physical constraints and allows for simulations in diverse scenarios.



a. Real model　　b. 3D model

Fig. 2. YASKAWA MOTOMAN-SIA5F

2.2. URDF Importer

A tool called URDF Importer was used to introduce the Unified Robot Description Format (URDF), an XML format file describing the robot's structure, into Unity. The URDF Importer analyses the URDF file and automatically generates and configures the links described therein as Unity Game Objects and joints as corresponding Joint components. It also imports the visual models (3D mesh files) and collision models specified in the URDF into Unity, giving each a mesh renderer, a collider and the appropriate rigid body components required for the physics simulation.

By unifying the definition of the robot models to be introduced into Unity and ROS, the high-performance physics simulation possessed by Unity can be used while maintaining compatibility with existing tools in ROS.

2.3. AR Foundation

In the developed mobile AR system, AR Foundation, provided by Unity, was adopted for the implementation of AR functionalities. While various frameworks exist for realizing AR features, including ARCore, ARKit, and HoloLens in addition to AR Foundation, the latter possesses a distinctive characteristic of absorbing the complex API differences of major mobile AR platforms, namely Apple ARKit (iOS) and Google ARCore (Android), to function as a unified abstraction layer. This property enables developers to efficiently deploy AR applications cross-platform across multiple mobile OS environments. Furthermore, support for Windows Mixed Reality platforms, including HoloLens, has also been realized. Moreover, the adoption of AR Foundation facilitates the integration with existing assets and external SDKs within Unity, thereby promoting the development of applications with complex AR capabilities. Table 1 lists the features available in each provider plug-in supported by Unity.

Table 1. Platform support

Feature	ARCore	ARKit	HoloLens
Session	Yes	Yes	Yes
Device tracking	Yes	Yes	Yes
Camera	Yes	Yes	
Plane detection	Yes	Yes	
Image tracking	Yes	Yes	
Object tracking		Yes	
Face tracking	Yes	Yes	
Body tracking		Yes	
Point clouds	Yes	Yes	
Ray casts	Yes	Yes	
Anchors	Yes	Yes	Yes
Meshing		Yes	Yes
Environment probes	Yes	Yes	
Occlusion	Yes	Yes	
Participants		Yes	

2.4. *Communications*

Data cannot be exchanged directly between Unity and ROS. Therefore, packages called ROS TCP Connector and ROS TCP Endpoint were introduced.

On the Unity side, we introduced ROS-TCP-Connector, a C# package embedded in the Unity project, which provides an API for sending and receiving messages to and from ROS topics (Publish/Subscribe), services and actions within a Unity application. On the ROS side, a Python package, ROS-TCP-Endpoint, was introduced to receive TCP connections from ROS-TCP-Connector on the Unity side and pass the messages to the topics.

The Transmission Control Protocol (TCP) is widely recognized as the basis for Internet communication and is a protocol that guarantees reliable transfer of data due to its high reliability. The strict data assurance mechanism possessed by TCP can be an extremely important communication method in fields such as robot control, where missing data or inconsistent order directly affects the safety and functionality of the entire system.

Fig. 3 Shows the communication architecture between Unity and ROS. The left side represents the Unity application, while the right side represents the ROS (Robot Operating System) environment. It can be seen that Unity and ROS exchange data by mutually utilizing services via the TCP communication protocol.

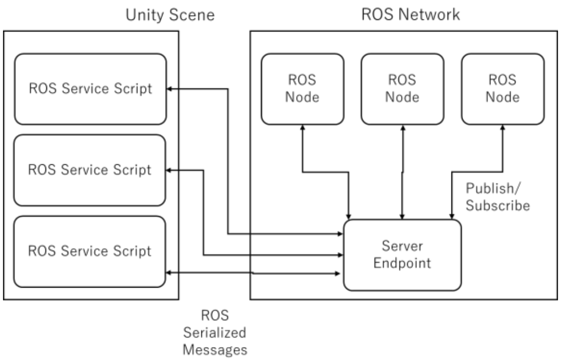


Fig. 3. ROS-Unity communication

2.5. *Communications*

An overview of the system is shown in Fig. 4.

On mobile devices, the project was developed in Unity and converted to an .apk file that can be run on mobile devices. The PC runs MoveIt, a ROS framework. When the camera is activated, a 3D model of robot arm is superimposed on the real environment through the mobile device's camera.

When the user presses a button on the screen indicating the start of the simulation, the current robot posture and the robot's target coordinates are passed to ROS via TCP communication, and ROS uses MoveIt to calculate the optimal route to reach the target coordinates based on the information received. The result is then returned to the mobile terminal, and the MoveIt motion plan is executed.

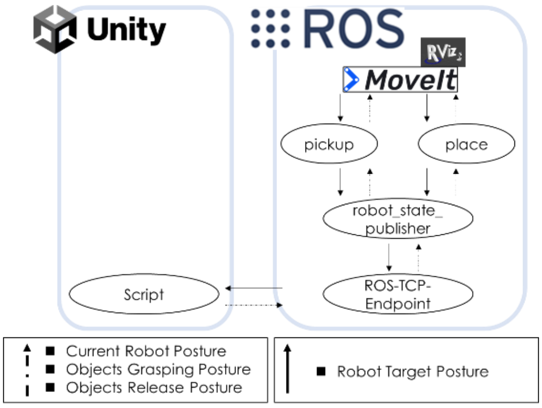
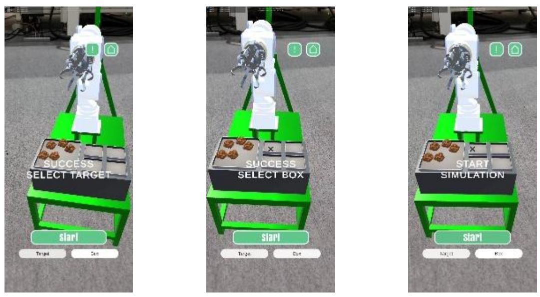


Fig. 4. System overview

3. Experiment

3.1. *Operation Check*

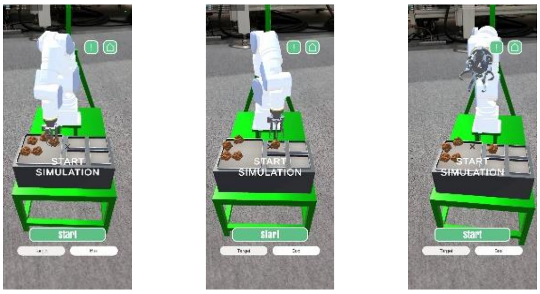
We confirmed that the mobile AR system we created works correctly. First, the flow of this system from application startup to the start of simulation is shown below. When the application is launched, the home screen is first displayed. After the screen transition, the camera of the mobile terminal is activated and the robot arm, grippers, and objects to be grasped (hereinafter referred to as "objects") are displayed on the screen. When the simulation start button is pressed, the message linkage with the ROS is initiated and the simulation using the 3D model of the robot arm begins. The results of the operation after the simulation start button is pressed on the mobile terminal are shown in Fig. 5. In the figure, "a" indicates the state in which the object to be grasped is selected, "b" indicates the state in which the box that the object is placed is selected, "c" indicates the initial state, "d" indicates that the object is grasped, "e" indicates that the object is placed, and "f" indicates that the operation is completed.



a. State of select object

b. State of select box

c. Initial state



d. State of grasping an object

e. State in which an object is place

f. State of operation completed

Fig. 5. Image of Mobile AR System

4. Consideration

Through experiments, we confirmed that the robotic arm was able to perform the desired motions. This indicates that the trajectory of the robot arm's end-effector, specifically for grasping an object, was appropriately calculated by MoveIt and accurately simulated.

However, a limitation of the current mobile AR system was identified: when the camera is activated, the robot arm appears fixed in front of it, preventing any adjustment of its position once displayed. Consequently, it is challenging to display the robot at an arbitrary location.

AR can be broadly categorized into two types: location-based AR and vision-based AR. Location-based AR utilizes positional information such as Wi-Fi, IMU (Inertial Measurement Unit), and GPS to display virtual content. As location-based AR does not require markers or specific visual features, it is well-suited for outdoor use.

In contrast, vision-based AR displays virtual content by recognizing and tracking specific markers (e.g., printed images or 2D codes) or objects present in the real world. Furthermore, vision-based AR includes two subcategories: marker-based AR, which recognizes predefined patterns, and marker less AR, which does not rely on specific markers. While marker-based AR can display virtual content with high accuracy and speed if a marker is present, it cannot function in areas without markers. Marker less AR, on the other hand, utilizes SLAM (Simultaneous Localization and Mapping) technology to display virtual content, often by performing plane detection, among other methods.

Moving forward, it is essential to develop a more user-friendly mobile AR system by effectively leveraging these AR technologies.

5. Conclusion

In recent years, there has been a marked shortage of labor in all industries in Japan due to the decline in the working population. While the introduction of robots into production lines, etc. is an effective method to solve the manpower shortage, the huge cost (expense and time) for system development and setting can be a major barrier to introduction. The 3D model of a robot arm can be made to behave in the same way as an actual robot. And we have developed a mobile AR system that combines this with AR, making it possible to check the line of human movement without having to bring the robot into the factory.

Unity is a game development platform and ROS was used to calculate the trajectory of the robot's paw position.

In the experiment, we confirmed that the robot was able to perform the specified operation. On the other hand, there are areas that need to be improved, such as the difficulty in displaying the robot at an arbitrary position. In addition, as a future prospect, the company aims to support a wide range of robot simulations by introducing other robots to the mobile AR system. In addition, the user interface will be improved to make it easier for users to use.

Furthermore, the Unity and ROS integration technology adopted in this study enables the construction of a highly reliable simulation environment. As a future outlook, we aim to seamlessly apply control strategies developed in this virtual environment to real-world robotic systems. Specifically, by integrating this technology with Unity's AR capabilities, it becomes possible to achieve intuitive remote operation of robots, pre-simulation of tasks in hazardous environments, and feedback of those results to the actual machine. These advancements are expected to significantly contribute to the enhancement of safety and efficiency in robot operations.

As part of our future work, we are planning to connect the system with a real robot. This will allow us to evaluate the discrepancy between the AR-projected robot motion and the actual robot motion before and after operation. Additionally, we are planning the development of a user interface (UI) that is intuitive and easy to use for individuals who have limited exposure to ROS.

References

1. Prem Gamolped, Sakmongkon Chumkamon, Chanapol Piyavichyanon, Eiji Hayashi, Abbe Mowshowitz, Online Deep Reinforcement Learning on Assigned Weight Spaghetti Grasping in One Time using Soft Actor-Critic, Proceedings of International Conference on Artificial Life and Robotics 2022, pp554-558
2. Nattapat Koomklang, Sakmongkon Chumkamon, Prem Gamolped, Tomofumi Tsuji, Eiji Hayashi, Abbe Mowshowitz, Deep-Learning-Based Designed Weight Picking Noodle-like Object, Proceedings of International Conference on Artificial Life and Robotics 2023, pp301-304
3. Yoshihiro Koyama, Eiji Hayashi, Akira Kawaguchi, Development of AR System for Grasping String Foods on Introduction of Industrial Robot. Proceedings of International Conference on Artificial Life and Robotics 2024, pp450-452
4. Takuya Matsumoto, Eiji Hayashi, Sakmongkon Chumkamon, Ayumu Tominaga, Abbe Mowshowitz, Rese arch on AR system for industrial robot introduction. Proceedings of International Conference on Artificial Life and Robotics 2023, pp323-326

Authors Introduction

Authors IntroductionMr. Takuma Aiko

He received his Bachelor's degree in Engineering in 2021 from the Faculty of Engineering, Kyushu Institute of technology in Japan. He is currently a master student in Kyushu Institute of Technology, Japan.

Mr. Gamolped Prem

He received bachelor's degree in engineering in 2021 from Robotics and Automation Engineering, King Mongkut's University of Technology North Bangkok in Thailand and received masters in 2022 from Kyushu Institute of Technology at Hayashi Laboratory. He is currently a PhD student in Kyushu Institute of Technology at Hayashi Laboratory and conducts research about Robotics and AI.

Prof. Eiji Hayashi

Prof. Eiji Hayashi is a professor in the Department of Intelligent and Control Systems at Kyushu Institute of Technology. He received the Ph.D. (Dr. Eng.) degree from Waseda University in 1996. His research interests include Intelligent mechanics, Mechanical systems and Perceptual information processing. He is a member of The Institute of Electrical and Electronics Engineers (IEEE) and The Japan Society of Mechanical Engineers (JSME).