Development of an Environmentally Independent Mobile Manipulation System for Product Disposal in Retail Stores

Ryogo Kai *Student Member, IEEE*, Kenta Ohashi, Hikaru Fujita, Takuya Kojima, Yuma Sasaki
Mihoko Niitsuma, *Member, IEEE*, Kazunori Umeda, *Member, IEEE*

***Abstract*— Robotic systems are being used to automate convenience stores and retail stores, including competitions such as the World Robot Summit. Many of the systems that have been proposed so far are environmentally dependent, such as putting markers on product packages or using motorized shelves modified from existing shelves and are not easy to implement in stores. Therefore, we proposed a system to dispose of products with as few changes to the environment as possible. To realize the system, we proposed three approaches: markerless product disposal, autonomous movement with navigation, and customer detection with sensors that can be placed at any position. We conducted verification based on real convenience store scenarios and discussed the insights and challenges obtained from it.**

I. INTRODUCTION

This paper presents the proposal and technical challenge for the stock and disposal task by Team HARChuo at the World Robot Summit (WRS) [1] Future Convenience Store Challenge (FCSC 2023) [2] held in July 2023.

The Stock and Disposal Task is one of the competitive tasks of the FCSC. The rulebook for the Stock and Disposal Task [3] describes the characteristics of the task as follows: ‘‘The proposed system must contribute to energy saving and work efficiency.’’ The accuracy of the developed system will be competed by demonstrating the stock and disposal process in a near-future convenience store using robotic technology. The specific roles of the system include autonomous movement to the product shelf, product disposal, interruption of the task when approaching customers, and avoidance.

In research related to the Stock and Disposal task, many systems are environmentally dependent. For example, the products themselves need to be marked to recognize the type and posture of the products. Moreover, the shelves themself need to be motorized or automated to facilitate product removal, and the shelves need to have many sensors to recognize the products and approaching customers [4-9]. There are concerns that the lack of product visibility will have a negative impact on customers' willingness to purchase. In addition, if motorized shelves are installed and many sensors are fixed to the environment, it will take time to recover from a system failure, and the layout of the store cannot be easily changed.

To deal with these issues, we are developing a mobile manipulation system that disposes of product items as environmentally independent as possible. This system is applicable not only in convenience stores but also in retail stores. The concept of the proposed system is to use the original packaging without markers on the products, to use simple sliding shelves without motorization or automation, to allow mobile manipulators to move without being controlled by markers, and to enable customer detection with only one sensor that can be placed freely in the store.

To realize this concept, we propose a mobile manipulation system consisting of three elements: robotic arm control for product disposal, mobile manipulator navigation, and customer approaching detection. For product disposal, image processing is performed using an RGB-D camera attached to the robotic arm. For object detection, YOLACT [10], which is a Realtime Instance-Segmentation robust to overlapping objects, is used. Principal component analysis is performed on the YOLACT detection results to calculate the grasped position of the object. For navigation, SLAM is performed using point cloud information acquired from a 2D LiDAR. For customer detection, the Single 3D LiDAR-based Moving Object Extraction system (S3L-MOE system) [11] is used. The S3L-MOE system is designed to detect static information such as object placement and dynamic information such as human activity by projecting 3D point clouds obtained from the 3D-LiDAR onto a two-dimensional cell. By using these three approaches, the system can be made independent of the environment. In this paper, we propose a system according to the following items and contribute by publishing the results.

- Disposal by a robotic arm without attaching markers to the products and without motorizing or automating the shelf itself.
- Navigation without attaching markers to the environment for self-location estimation.
- Detection and avoidance of customers using a system that can be freely installed at any position in the environment.

II. SYSTEM OVERVIEW

A. Hardware

The hardware constraints of the mobile manipulator and the product shelf are defined in the rulebook [3]. To meet those constraints, we prepared a mobile manipulator and product shelf as shown in Fig. 1. The mobile robot shown in Fig. 1(a) is equipped with a robotic arm and a two-fingered parallel robotic hand. Image processing is performed using the RGB-D camera mounted on the robotic arm to control the robotic arm and hand. The mobile robot navigates using information obtained from a 2D LiDAR installed. Additionally, a 3D

* Research partially supported by Chuo University Joint Research Project.
R. Kai, K. Ohashi, H. Fujita, T. Kojima, Y. Sasaki, M. Niitsuma, and K. Umeda are with Chuo University, Bunkyo-ku, Tokyo, 112-8551. (e-mail: {niitsuma, umeda}@mech.chuo-u.ac.jp)

LiDAR is installed in the environment for customer detection mentioned in Section III (Fig. 1(b)).

The shelf is designed with sliding shelf boards (Fig. 1(c)). This is designed to achieve a similar purpose as pulling out shelves to improve operational efficiency when arranging products in an actual convenience store while also ensuring workspace allocation and enhancing efficiency for the robotic arm. One ArUco marker is placed on the shelf and used to measure the distance when pulling out the shelf.

B. Software

The mobile robot is required to operate autonomously after the start of the competition, according to the rule [3]. To propose a system that fulfills the concepts outlined in subsection I, we performed tasks following the flow shown in Fig. 2. Furthermore, to achieve autonomous operation, we developed the software system depicted in Fig. 3. We will explain the system in Fig. 3 using Fig. 2 as a reference.

In this system, System Manager shown in Fig. 3 utilizes Robot Status and Move Flag to control the robotic arm and the mobile robot independently. This enables separate control of these two components. When one of them is in motion, its status is transmitted as Robot Status, and Move Flag is sent to the other component to indicate whether it is allowed to move.

First, as shown in Fig. 2(a), the mobile manipulator moves to the front of the shelf. At this point, in Fig. 3, System Manager sends a Move Flag to the mobile robot, which is highlighted in orange, initiating its movement.

After reaching the front of the shelf, information indicating the mobile robot's stoppage is transmitted to System Manager. Subsequently, System Manager sends a Move Flag to the robotic arm depicted in blue in Fig. 3. As a result, as shown in Fig. 2(b), the robotic arm performs tasks such as pulling out the shelf, detecting objects, and disposal the products.

During the disposal process, there is a possibility that customers may approach the shelf for shopping (Fig. 2(c)). In such cases, it is necessary for the mobile manipulator to detect customer and moving away from the front of the shelf, ensuring that customers can freely continue shopping. Here, we continuously detect customers using the 3D LiDAR sensor. The detected result is transmitted to System Manager via the green block shown in Fig. 3. If customers are detected, System Manager sends a flag to halt the robotic arm's movements. Once the robotic arm comes to a halt, the robotic arm informs the robotic arm's status, System Manager sends a flag to the mobile robot to initiate avoidance actions. When the customer moves away from the front of the shelf, System Manager sends a flag to the mobile robot to return to the front of the shelf. After receiving the flag of the mobile robot's arrival, System Manager sends a flag to the robotic arm to restart the task.

Once the disposal task is completed, as shown in Fig. 2(d), the mobile manipulator returns to the stock room. If the product disposal is succeeded, System Manager sends a Move Flag to the mobile robot to go back to the stock room.

For the system operation of robotic arms and mobile robots, Robot Operating System (ROS) is used. The 3D LiDAR used for customer detection is integrated with RT middleware.

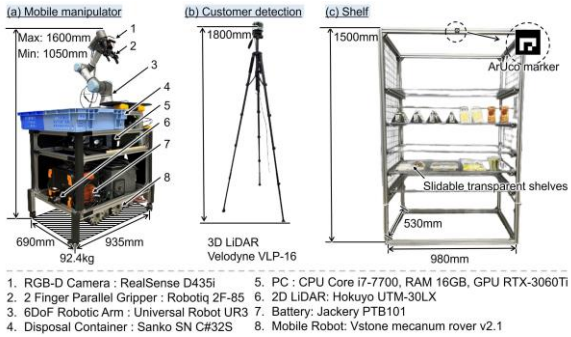

Figure 1. Hardware components.

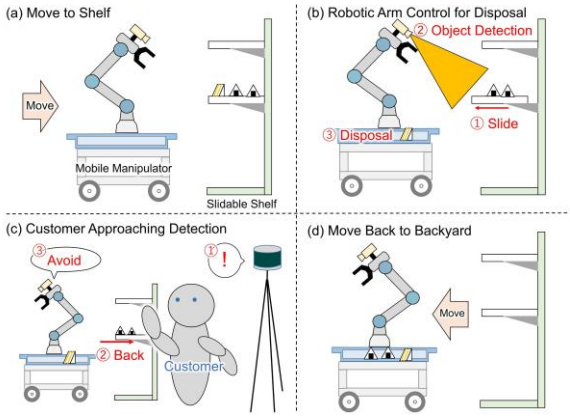

Figure 2. Flow of the proposed system.

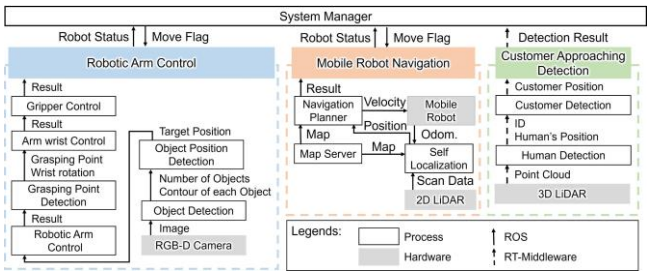

Figure 3. Software system.

Communication between ROS and RT middleware is established through TCP/IP communication.

III. METOHDS

A. Robotic Arm Control

The control of the robotic arm and robot hand is performed using the RGB-D camera mounted on the robotic arm. Here, the processes in Fig. 3 are explained using Fig. 4.

First, shelf extraction is performed (Fig. 4(a)). According to the competition regulations, the maximum gap between the shelf boards is 250 mm, posing a challenge for robotic arm operations. Therefore, to ensure sufficient workspace for the robotic arm, the robotic arm pulls out the shelves. To pull out shelves, the distance to the ArUco marker installed on the shelf is measured using an RGB-D camera mounted on the robotic arm. Based on the measured distance, the shelves are pulled out as shown in Fig. 4(a2) to (a4).

Next, object detection is performed (Fig. 4(b)). For object detection, the robotic arm is positioned in a Birds Eye Pose

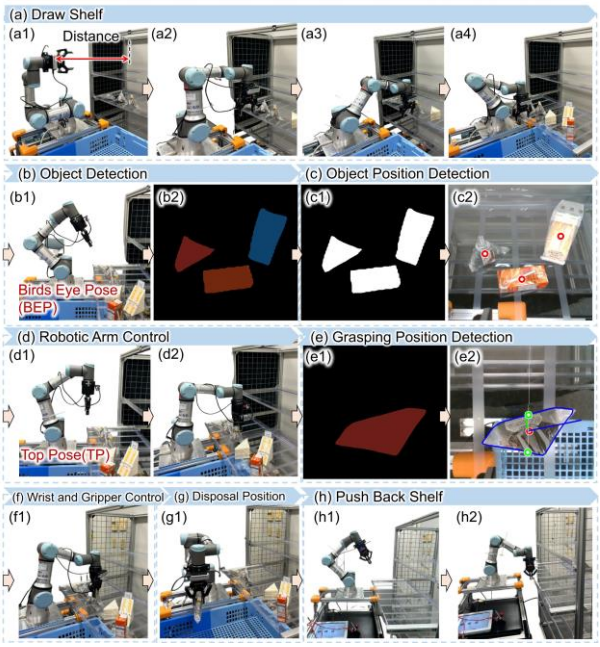

Figure 4. The process of product disposal using a robotic arm.

(BEP), as shown in Fig. 4(b1), allowing the RGB-D camera to capture an overhead view of the objects. When performing object recognition in BEP, YOLACT [10], which is one of the Real-time Instance Segmentation methods, is used. YOLACT is suitable for product disposal due to its ability to achieve real-time detection and its robustness in detecting overlapping objects. Running YOLACT in BEP results in object detection as shown in Fig. 4(b2).

Using the results of object detection, the positions of the objects are calculated. By applying RGB thresholding to the image from Fig. 4(b2), a binary image is obtained where object regions are displayed in white (Fig. 4(c1)). From this binary image, the contours of each object are extracted, and the pixel coordinates of the image centroid, indicated by the red circles in Fig. 4(c2), are calculated. This process allows us to determine the number of *N* objects centroids on the shelf.

Using the calculated results, the order to manipulate each object is determined by horizontally traversing from the top left of the image. Each object is processed individually based on the established order. For the ith object, its position is calculated in 3D coordinates ($x_{i_{BEP}}^{camera}, y_{i_{BEP}}^{camera}, z_{i_{BEP}}^{camera}$) using the pixel value of the calculated image centroid for the ith object, the distance to the image centroid obtained from the RGB-D camera, and the camera's intrinsic parameters. Here, the superscripts denote the coordinate systems, and subscripts indicate the ith object. And the subscript i represents the calculated robotic arm pose. These calculated coordinates serve as the position of the object, which is set as the target position for controlling the arm.

Next, based on the target positions, the robotic arm is controlled (Fig. 4(d)). Attempting to control the arm from BEP to the target position using the shortest path might lead to collisions with objects along the path. Therefore, control is executed based on the Cartesian coordinate system. MoveIt[12] is used for robotic arm control and coordinate transformations. Firstly, the detected object position in BEP is transformed from the camera coordinate system to the robotic arm's wrist coordinate system ($x_{i_{BEP}}^{wrist}, y_{i_{BEP}}^{wrist}, z_{i_{BEP}}^{wrist}$). Subsequently, it is further transformed from the wrist coordinate system to the robotic arm base coordinate system ($x_{i_{BEP}}^{base}, y_{i_{BEP}}^{base}, z_{i_{BEP}}^{base}$). After the calculations, the robotic arm is positioned into the Top Pose (TP) as shown in Fig. 4(d1). In this TP pose, the coordinates are transformed from the base coordinate system($x_{i_{BEP}}^{base}, y_{i_{BEP}}^{base}, z_{i_{BEP}}^{base}$) to the wrist coordinate system ($x_{i_{TP}}^{wrist}, y_{i_{TP}}^{wrist}, z_{i_{TP}}^{wrist}$). This coordinate transformation enables control to guide the robotic arm to the target position of the object. Using this coordinate transformation, the robotic arm's control is performed to move to the target position as shown in Fig. 4(d2).

Once the target position is reached, the grasping position is calculated. After performing object recognition again using YOLACT in the configuration of Fig. 4(d2), Fig. 4(e1) is obtained. Following the same procedure as with Fig. 4(c1), a binary image is obtained, and the object contours are extracted. Performing principal component analysis (PCA) based on the object contours with respect to the centroid of the object results. In Fig. 4(e2), the direction of the second principal component's eigenvector, represented by the green line, is calculated. Here, the second component of the principal component analysis and the plane perpendicular to the direction are assumed to have a larger graspable area for the robotic hand. The intersection point of the eigenvector's extension line and the object's contour determines the grasping position as indicated by the green circle. Upon determining the grasping position, the output specifies how much the robotic arm's wrist needs to rotate and how much the robotic hand should close.

Using the calculated results, the robotic arm's wrist and hand are controlled as shown in Fig. 4(f). Subsequently, the robotic arm is controlled to dispose of the object into the container (Fig. 4(g)). Once the disposal is completed, the robotic arm is controlled to return the shelf back to its original position, as shown in Fig. 4(h1) and (h2). This movement is controlled based on the path determined using the distance to the shelf calculated in Fig. 4(a1). By following this series of processes, the robotic arm performs the disposal of products.

B. Mobile Robot Navigation

To navigate our robot, we used navigation package [13], released as a ROS package. The robot builds an environmental map using gmapping [14], which performs SLAM based on odometry and point cloud information from a 2D LiDAR. The size of competition field is shown in Fig. 5(a), and an environmental map constructed at the competition is shown in Fig. 5(b). Autonomous navigation is performed based on Navfn [15] for global path planning and Dynamic Window Approach [16] for local path planning. As the robot moves, it follows the generated path. There is no need for local alignment, such as referencing markers as the robot moves. For localization, we use amcl [17], which estimates self-location based on scan data from the 2D LiDAR and odometry information. The position coordinates of the robot's destination, such as the front of the shelf, are set as ①~④ in Fig. 5(b) in advance. In the competition, the robot must first move to a workable position in front of the shelf where it is to task. When the robot moves to the front of the shelf, it goes

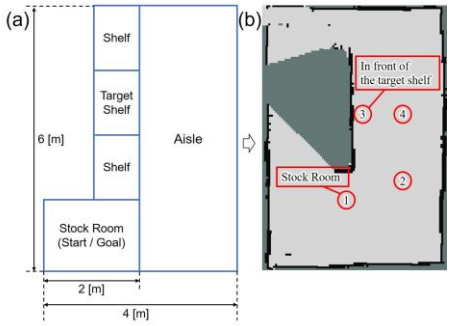

Figure 5. An environmental map constructed at the competition.

TABLE I. THE VALUES OF THE NAVIGATION PARAMETERS

Parameter name	Value
foot_print [m]	[0.468, 0.345] [0.468, -0.345] [-0.468, 0.345] [-0.468, -0.345]
infration_radius [m]	0.05
xy goal tolerance [rad]	0.05
yaw_goal_tolerance[m]	0.05
fake_foot_print [m]	[0.468, 0.22] [0.468, -0.22] [-0.468, 0.22] [-0.468, -0.22]

from the initial position ① through ② to the front of the target shelf ③. ② is a transit point, established to make it easier to plan the path to the front of the target shelf. When the System Manager receives a flag indicating that the task of the robotic arm is finished, the mobile manipulator returns from ③ to ① via ②. Table I shows the values of the navigation parameters. These parameters are related to path planning and robot motion. The parameter values are determined empirically, as it is difficult to reach the destination with the default values of the navigation package.

C. Customer Approaching Detection

In the stock and disposal task, it is necessary to do the avoidance action that considers the direction from which customers are coming because the width of the aisle is narrow. In this paper, we use the Single 3D LiDAR-based Moving Object Extraction system (S3L-MOE system) [11] to detect humans. This system achieves real-time human detection without the need for a learning process. Three-dimensional point clouds obtained every 0.1 seconds from a single 3D-LiDAR unit are projected onto cells of a two-dimensional grid map, and moving objects are extracted by background subtraction. By storing the positions of all objects in the environment, the relative positions of objects can be acquired in real time. It is also possible to track people by assigning an ID to each person, thereby enabling recognition of customers and detection of the direction of approach within a store.

The following procedure is used for human detection in this system. First, the system detects the position of a stationary object that serves as a background for an arbitrary time in an unoccupied environment. In this study, the background was detected in 2 minutes of measurement. Next, the background is subtracted from the scan data, and the point cloud is divided by clustering to extract moving objects. Finally, a moving object is detected as a person if the number of cells occupied by the moving object $N_{MO_{cell}}$ satisfies

$$2 \leq N_{MO_{cell}} \leq 50 \quad (1)$$

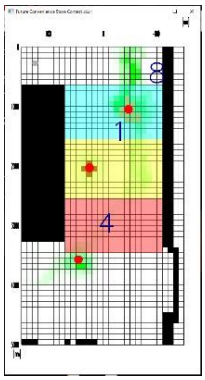

Figure 6. The map drawn by using S3L-MOE system: The red rectangle shows a customer detection right side of the shelf. The yellow rectangle shows a customer detection in front of the shelf. The blue rectangle shows a customer detection left side of the shelf. The red circles show the person's positions in the shop. The number in the upper right of the red circle shows the ID of each person.

and the average position of the point cloud is calculated as the position of the person. The person is identified by the ID assigned to each moving object.

The following procedure is used for customer detection. In this system, the mobile manipulator is large and is likely to be detected as a human. Since the mobile manipulator and clerks start their work from the stock room, the system recognizes a person who accesses the stock room shown in Fig. 6 as clerks or the mobile manipulator. By storing the IDs of the clerks and the mobile manipulator ID_{clerk}, the system detects customers.

$$ID_{clerk} = \begin{cases} ID_{person_i}, & \text{if } person_i \text{ is in the stock room,} \\ 0, & \text{otherwise.} \end{cases} \quad (2)$$

ID_{person_i} represents the ID assigned to any person $person_i$. The following procedure is used to detect the direction of approach. When a customer comes within the three areas shown in Fig. 6, which are determined based on the position of the target shelf, the direction of the customer's approach is sent to the mobile manipulator as a flag. The flag sent to the mobile manipulator $F_{direction_{customer}}$ is calculated by the following equations.

$$F_{right_{shelf}} = \begin{cases} 1, & \text{if } customer_i \text{ is in red area in Fig. 6,} \\ 0, & \text{otherwise.} \end{cases} \quad (3)$$

$$F_{center_{self}} = \begin{cases} 2, & \text{if } customer_i \text{ is in yellow area in Fig. 6,} \\ 0, & \text{otherwise.} \end{cases} \quad (4)$$

$$F_{left_{self}} = \begin{cases} 4, & \text{if } customer_i \text{ is in blue area in Fig. 6,} \\ 0, & \text{otherwise.} \end{cases} \quad (5)$$

$$F_{direction_{customer}} = F_{right_{shelf}} + F_{center_{self}} + F_{left_{self}} \quad (6)$$

The above procedure allows the mobile manipulator to know the direction in which a customer is coming, even when multiple customers are approaching. For example, when $F_{direction_{customer}} = 5$, it means that customers are approaching from both sides of the target shelf.

Based on the flag calculated by (6), the mobile manipulator performs the avoidance action. When avoiding a customer, the mobile manipulator provides information by voice and asks the customer to leave. After confirming that there is no customer in the yellow area in Fig. 6, the mobile manipulator

starts moving from ③ to ④ in Fig. 5, and then moves to ②. When no more customers are detected in front of the shelf, the mobile manipulator moves from ② to ③ and resumes its work.

IV. EXPERIMENTAL RESULTS

We have verified that each system works in an integrated manner via System Manager. This paper analytically verifies the performance of each system to clarify its performance.

A. Robotic Arm Control

Disposal experiments were performed using three items commonly used in the competition: rice balls, sandwiches, and orange juices. A total of 10 grasping experiments were performed for each object. The evaluation consisted of two aspects: object recognition success rate and disposal success rate. The object recognition success rate was determined as the percentage of successful cases where the centroid of the object was calculated. The disposal success rate was calculated as the percentage of cases where the robot successfully disposed the object. The results are presented in Table II.

B. Mobile Robot Navigation

We conducted simulations and real-world validations within an environment model resembling the competition arena (Fig. 5(a)). The objective was to verify whether the robot could successfully approach the target shelf and execute tasks. For our robot to comfortably pull out the shelf, it is necessary for the robot to position where the robot's center-to-shelf distance is within 0.5 meters. Throughout the validation, we conducted 10 trials and assessed the success rate by examining cases in which the robot's center-to-shelf distance was less than 0.5 meters upon reaching the forefront of the shelf.

The simulation validation indicated that the robot couldn't reach the front position of the shelf. As a result, we carried out additional validation by modifying the robot's footprint parameter, which influences path generation and tracking within the navigation system. We reduced the size of the robot's footprint to be smaller than that of the actual robot, equivalent to the *fake_foot_print* in Table I. Table III shows the results.

C. Customer Approaching Detection

To validate the effectiveness of the customer detection method, experiments were conducted in the same environment as *Mobile Robot Navigation* experiment. The evaluation consisted of the following three aspects: recognition rate of person detection, time taken from receiving the customer detection flag in the *System Manager* (Fig. 3) to playing the avoidance prompt audio ($\mathrm{Time}_{\mathrm{audio}}$), time taken from receiving the customer detection flag in the *System Manager* (Fig. 3) to the robotic arm being in a state capable of avoidance ($\mathrm{Time}_{\mathrm{avoid}}$). Table IV shows the experimental results.

V. DISCUSSION

A. Robotic Arm Control

The success rates of object detection exceeded 90% in all cases. In case the orange juice was not detected, the captured image from the camera was shown in Fig. 7(a). The cause of this could be attributed to insufficient training of YOLACT.

The common cause of disposal failure across all objects was the discrepancy between the centroid calculated from the image and the actual centroid, which is taken as the centroid of the object's bottom surface. When using YOLACT, the contour of all sides composing a single object visible in the image is obtained (Fig. 7(b)). Consequently, due to variations in how the object is perceived, differences arise between the image processed centroid and the centroid of the object's bottom surface. This discrepancy causes, even if the robotic arm was controlled to the target position, the object would be off-center from the hand’s center, leading to unsuccessful grasping.

In the case of the orange juice, when the product was placed upright, as shown in Fig. 7(c), the object shifted from the gripper's center, causing inaccurate depth detection from the RGB-D camera. It caused the product to be crushed.

Furthermore, for sandwiches, the grasp position differed from the intended position due to the result of PCA. This occurred because the inclined part of the sandwich was projected onto a 2D plane when viewed from the top, causing it to be identified as the second principal component (Fig. 7(d)).

TABLE II. DISPOSAL EXPERIMENTS USING THE ROBOTIC ARM.

Object	Object detection success rate [%]	Disposal success rate [%]
Rice ball	100	70
Sandwich	100	40
Orange juice	90	50

TABLE III. COMPARISON EXPERIMENT OF PATH PLANNING USING SIMULATION AND ACTUAL ROBOT.

	Virtual Environment		Actual Environment	
Footprint	Actual Size	Fake Size	Actual Size	Fake Size
Number of Arrivals	0	10	3	10
Success Rate	0 [%]	100 [%]	10 [%]	60 %]

TABLE IV. EXPERIMENT ON CUSTOMER DETECTION IN A SIMULATED COMPETITION ENVIRONMENT.

	$\mathrm{Time}_{\mathrm{audio}}$		$\mathrm{Time}_{\mathrm{avoid}}$	
Customer Detection Success Rate	Average	Standard Deviation	Average	Standard Deviation
100[%]	2.71[s]	1.97[s]	26.95[s]	5.06[s]

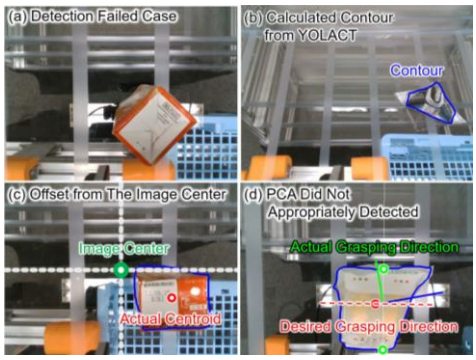

Figure .7 Example of Disposal Failure

In the future, we are considering using a 6 DoF pose estimation method to improve the accuracy of centroid calculation and enhance the success rate of grasping. By deriving the centroid from the 6D pose, more precise control target values for the robotic arm can be obtained, allowing for the pre-specification of grasp positions for each object. However, when we used the current proposed 6D pose estimation method as a test, such as Gen6D [18], achieving high accuracy has not been confirmed. Therefore, by pre-defining the robotic arm's posture for 6D pose estimation, the accuracy of 6D pose estimation can be improved, making it feasible for practical use in real-world environments.

B. Mobile Robot Navigation

In this verification, the number of times the robot reached the task position was improved when the foot_print parameter was reduced, regardless of whether the robot was used in simulation or on the actual machine. This is because a smaller foot print parameter allows more movements in the motion plan, even near obstacles such as shelves. Also, there were some cases in which the robot reached the task position in the case of the actual environment experiment but did not meet the requirements. This occurred because of odometry errors caused by slipping the mobile robot's wheels. The main reason for the wheel slip is thought to be due to the heavy load of the robot.

In the future, we expect to improve the success rate without adjusting the size of the footprint by navigating to the periphery of the work area and adjusting the position in detail by feeding back the information from the self-position estimation by the on-board 2D LiDAR and so on. Moreover, the mobile robot we used needs to be reconsidered, since slipping during movement makes it difficult to control.

C. Customer Approaching Detection

In this verification, the success rate of customer detection was 100%. This indicates that the S3L-MOE system was working properly. The mean time of $\text{Time}_{\text{audio}}$ took 2.714[s]. The standard deviation was 1.97[s], showing a large variation. This is because the customer approached the robot when it could not immediately halt its motion sequence due to the middle of the task execution.

In the future, we believe that the average time to the audio presentation and avoidance can be shortened, and the variation in avoidance time can be reduced by subdividing the motion processing of the robot arm and shortening the cycle for receiving the customer detection flag.

VI. CONCLUSION

This paper presents a proposal and technical challenge for the Stock and Disposal task at the World Robot Summit. We proposed a mobile manipulation system that disposes of merchandise as environmentally independent as possible. We proposed a robot arm that disposes of merchandise on a simple sliding shelf without markers on the merchandise, an approach in which the mobile manipulator is not controlled by markers and moves to avoid customers even while traveling, and a customer detection method that includes the direction of approaching customers using only one sensor that can be freely placed inside the store. Each approach was verified in an environment that simulates the real environment. Future work will focus on product disposal by manipulation using 6-DoF posture estimation method, detailed position adjustment of mobile robots using feedback of self-position estimation information, and reduction of response time until customer avoidance behavior is achieved.

REFERENCES

[1] "World Robot Summit" https://wrs.nedo.go.jp/ (accessed Aug. 10, 2023).
[2] "FUTURE CONVENIENCE STORE CONTEST" https://f-csc.org/ja/wrs-fcsc-ifac-2023/ (accessed Aug. 10, 2023).
[3] "WRS Future Convenience Store Challenge 2023 Stock and Disposal Task Rulebook" https://f-csc.org/wp-content/uploads/2023/02/FCSC2023_stock_disposal_task_v001_Eng.pdf (accessed Aug. 10, 2023).
[4] "Future Convenience Store Challenge DAY2 (September 11,2021)" https://www.youtube.com/live/J3p6wuFzX40?feature=share (accessed Aug. 10, 2023).
[5] "Future Convenience Store Challenge DAY3 (September 11,2021)" https://www.youtube.com/live/aEeoPNXMIEE?feature=share (accessed Aug. 10, 2023).
[6] G. A. Garcia Ricardez *et al*, "Autonomous service robot for human-aware restock, straightening and disposal tasks in retail automation," Advanced Robotics, Vol. 36, No. 17-18, pp. 936-950, 2022.
[7] M. Seki *et al*, "Development of XYZ stage-type display robot system for stock and disposal tasks in convenience stores," Advanced Robotics, Vol. 36, No. 23, pp. 1252-1272, 2022.
[8] R. Tomikawav *et al*, "Development of display and disposal work system for convenience stores using dual-arm robot," Advanced Robotics, Vol. 36, No. 23, pp. 1273-1290, 2022.
[9] K. Mizutani *et al*, "Development of automatic opening and closing shelf for convenience stores in cooperation with collaborative robot," Advanced Robotics, Vol. 36 No. 23, pp. 1241-1251, 2022.
[10] D. Bolya *et al*, "YOLACT: Real-Time Instance Segmentation," IEEE/CVF International Conf. on Computer Vision (ICCV), pp. 9156-916, 2019.
[11] T. Nishio and M. Niitsuma "Moving objects extraction for building environmental maps describing human walking activity using a 3D LiDAR," Advanced Robotics, Vol. 36, No. 23, pp. 1291-1304, 2022.
[12] D. Coleman *et al*, "Reducing the Barrier to Entry of Complex Robotic Software: a MoveIt! Case Study.," arXiv:1404.3785, 2014.
[13] "ros-planning/ navigation", https://github.com/ros-planning/navigation (accessed Aug. 26, 2023).
[14] L. Valle *et al*., "Planning Algorithms," Cambridge University Press, pp. 371-381, 2006.
[15] D. Fox *et al*, "The dynamic window approach to collision avoidance," IEEE Robotics & Automation Magazine, Vol. 4, No. 1, pp. 23-33, 1997.
[16] S. Thrun *et al*, "Probabilistic Robotics (Intelligent Robotics and Autonomous Agents)", The MIT Press, 2005.
[17] G. Grisetti *et al*, "Improved Techniques for Grid Mapping with Rao-Blackwellized Particle Filters," IEEE Trans. on Robotics, Vol. 23, pp. 34-46, 2007.
[18] Y. Liu *et al*, "Gen6D: Generalizable Model-Free 6-DoF Object Pose Estimation from RGB Images," arXiv:2204.10776, 2022.