Article | Received 25 February 2025; Accepted 8 July 2025; Published Day Mon Year
https://doi.org/10.55092/rl20250005

Design of a household robot with autonomous navigation for object detection and sorting†

Bingjie Xu[1], Yangzesheng Lu[2], Jingxiang Wang[2], Qinglei Bu[2,*], Mark Leach[2] and Jie Sun[2,*]

[1] School of AI, Suzhou Industrial Park Institute of Vocational Technology, Suzhou, China
[2] Department of Mechatronics and Robotics, Xi'an Jiaotong-Liverpool University, Suzhou, China

† This work is an extended version of the paper presented at the 2024 29th International Conference on Automation and Computing (ICAC), and copyright permission has been obtained from IEEE for its publication in this journal.

* Correspondence authors; E-mails: Qinglei.Bu02@xjtlu.edu.cn; Jie.Sun@xjtlu.edu.cn.

Highlights:

- Depth camera-based YOLOv11 object recognition.
- Enhancing autonomous navigation and grasping capabilities of domestic robots.
- Model-trained keywords-based speech recognition human-computer interaction.

Abstract: With the continuous advancements in robotics, household robots are increasingly becoming an integral part of daily life. To enhance robots' functionality and improve accessibility, we propose the development of an intelligent household robot designed to serve as a family assistant. The robot's automation capabilities enable it to independently perform various tasks, including object retrieval and interactive entertainment. By integrating a movable chassis, robotic arm, lifting platform, and flexible gripper, the robot is capable of grasping objects of varying sizes and types. The robot's vision system is built in conjunction with the YOLOv11 model, allowing it to detect target objects using a depth camera. Additionally, the robot employs 2D LiDAR and the Navigation2 framework in ROS2 to generate a 2D radar map of its environment. Through this pre-generated map, the robot can autonomously navigate indoor spaces. In addition, a speech recognition system was used to achieve efficient human-robot interaction. A functional prototype has been tested in an indoor setting, demonstrating the effectiveness and feasibility of the proposed design.

Keywords: object detection; YOLO algorithm; movable household robot; depth camera

1. Introduction

In recent years, there has been significant progress in artificial intelligence technology, and the field of robotics has emerged as a vital technology driving the future development of artificial intelligence [1,2]. Based on their specific application scenarios, the International Federation of Robotics (IFR) categorizes

robots into two main types: industrial robots and service robots. The latter can be categorized into household and professional service robots. Household service robots primarily prioritize domestic tasks, assisting users with leisure and entertainment, which is in contrast to professional service robots designed for public service purposes [3–5]. On October 8, 2024, the International Federation of Robotics (IFR) released the 2024 World Robotics Report: Service Robots, which reported a 30% increase in global sales of professional service robots. In 2024, the service robotics industry experienced explosive growth, with consumer service robots reaching 4.1 million units and professional service robot sales exceeding 200,000 units for the first time. The service robot market continues to expand, and the household service robot market is also steadily growing. On January 22, 2025, IFR predicted that in Germany humanoid robots would continue to attract widespread attention in 2025. This forecast indicates that there is still room for growth in the humanoid household robot market.

Currently, some household robots can provide companionship and simple functions such as desk cleaning, object transportation, and video playback. The HSR robot, developed by Jia Yin *et al.* [6], was designed to perform cleanliness checks and clean food waste from tables using deep learning technology. The robot has a robotic arm and a mobile chassis, allowing it to grasp a rag using its arm and wipe waste from table surfaces. Although the HSR robot demonstrates high accuracy in detecting desktop waste, its functionality is highly specialized and limited to desktop cleaning tasks. Similarly, the CHARMIE robot, developed by Tiago Ribeiro *et al.* [7], incorporates multiple hardware components, including a kinematic platform, a robotic arm, a lifting mechanism with a torso, and a robotic head, enabling it to perform more complex tasks. It has omnidirectional wheels and an independent suspension system to enhance operational stability. However, CHARMIE is constrained by its single gripper, which can only handle objects of specific sizes, such as cans, and cannot manage slightly larger objects. In contrast, the APR-02 robot, developed by Jordi Palacín *et al.* [8], represents a second-generation design to increase the robot's anthropomorphic characteristics to improve user affinity and social acceptance. This robot features two robotic arms and hands, but these components are primarily aesthetic and lack practical utility. Nevertheless, APR-02 has made notable advancements in naturalness and personalization during user interactions compared to other robots. However, the design does not adequately address safety considerations, particularly preventing potential damage to people or objects when grasping objects. The object detection algorithm also exhibits limitations, requiring further enhancement to improve its robustness and adaptability for handling diverse tasks.

To sum up, the three robots discussed above have each demonstrated their respective strengths in human-robot interaction, achieving varying degrees of success in physical execution, basic decision-making, and user interaction. In terms of physical execution, the HSR robot is capable of performing preset basic tasks such as wiping tables. Regarding basic decision-making, the CHARMIE robot can make fundamental judgments based on sensor feedback, such as avoiding obstacles. Regarding user interaction, the APR-02 robot communicates with users through its display screen to provide information and engage in simple interactions. However, despite their strengths, all three robots have significant limitations. They cannot perform tasks in complex situations and fail to meet users' expectations for performing diverse and challenging tasks.

Currently, the latest version of household robots are capable of providing emotion recognition functions and offering companionship services to family members. The functions of emotion recognition

and companionship provision require robots to read and analyze human facial expressions, vocal tones, and verbal content through cameras, thereby assessing human emotions and responding appropriately [9]. Additionally, certain companion robots are equipped with an early-stage education function for children. However, in order to transform household robots into family members with a higher level of human-robot interaction, a significant amount of effort is still required, as opposed to using a simple dialogue function. In conclusion, the household robots previously mentioned are primarily designed to fulfil a single function, which is insufficient to satisfy the objective of using household robots to perform multiple tasks.

In summary, we focuses on enhancing autonomous navigation and object detection and grasping capabilities of domestic robots, and proposes a complete design solution. In hardware, we independently designed a lightweight chassis and flexible gripper, and optimized the camera position to expand the field of view. In software, we compared the performance of the YOLOv8 and YOLOv11 algorithms, and combined depth camera data to improve navigation and grasping accuracy. Through the integration of hardware and software and experiments, the robot achieved full functionality, from speech recognition to autonomous navigation, object recognition, and grasping, offering new ideas for domestic robot applications. This article is an extended version of a conference paper [10]. It provides more detailed descriptions of the design, fabrication, and grip finger.

2. Hardware design

Usually, moveable robots rely on motor-driven wheels for their movement [11]. The kinematics of robots are completely different depending on the chassis structure. To facilitate the autonomous movement of the robot, a two-wheel differential kinematics configuration, which is flexible and simple, has been chosen as the basis. As is shown in Figure 1, the two power wheels of the two-wheel differential robot are set on the left, and right sides of the chassis, and the speeds of the two wheels can be controlled independently so that the chassis can move in a straight line or steered by controlling the speed of each wheels. Simultaneously, the chassis is equipped with an auxiliary support universal wheel to ensure balance, resulting in a three-wheeled wheel system structure. The lower sides of the circular robot chassis, which has a diameter of approximately 45 cm, have been powered by active wheels. The robot is able to freely navigate passageways that are wider than 50 cm due to the vehicle's center-symmetric design.

The proposed household robot comprises six parts: a depth camera, a digital crystal display, a robot arm, a flexible gripper, a lifting platform, and a movable chassis. In the middle layer of the chassis, the robot is equipped with environmental perception and positioning capabilities through the arrangement of a Raspberry Pi, LiDAR, Inertial Measurement Unit (IMU), speakers, motor speed controller, battery, and power management module. Key components with their parameters are shown in Table 1.

Table 1. Components and main parameters.

Components	Main Parameters
LiDAR (LS-M10P)	Scan frequency 12 Hz
Inertial measurement unit (GY-95T)	Refresh rate 1 kHz
Wheel Odometer	Integrated in ESC
Battery	Rated Voltage 22.2 V (6S)
Depth camera	Range 0.6–8 m

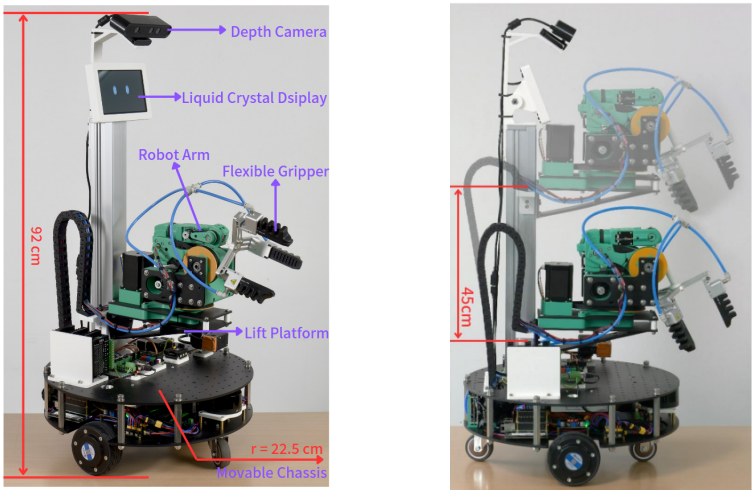


Figure 1. Hardware design of the proposed household robot.

This research designed and proposed a comprehensive object-picking strategy that included a pneumatic flexible gripper, a lifting platform, and a mechanical arm to address the necessity of organizing toys dispersed across the ground in Figure 2. This arm utilizes large-area 3D printing to reduce the use of metal materials, resulting in a lighter overall weight and allowing for more flexible gripping movements compared to traditional robotic arms. The drive scheme is based on the basic open-source stepper motor SmallRobotArm project and consists of two 57-stepper motors, one 42-stepper motor, two 28-stepper motors, and one 20-stepper motor. It uses a synchronous belt drive, which transmits motion through the mesh of equally spaced transverse teeth on the inner surface of the belt and the corresponding tooth grooves on the pulleys. This design ensures a strict transmission ratio with higher precision than traditional friction-type belt drives, as it eliminates relative sliding between the pulley and the drive belt.

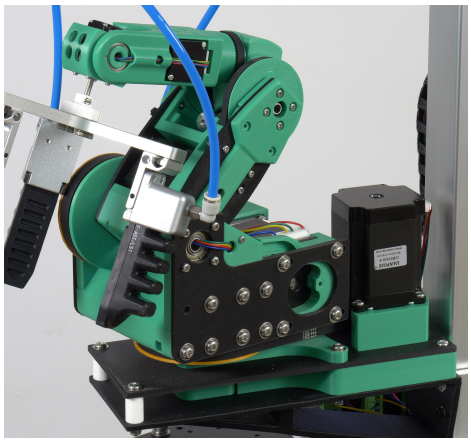


Figure 2. 3D printed foldable robot arm with pneumatic gripper.

Unlike traditional mechanical grippers made from rigid materials, this soft gripper is flexible and agile, as illustrated in Figure 3. It features three lightweight, easy-to-dissemble flexible fingers made from soft material, which adds to its safety. The gripper can adjust its size by modifying the distance between the clips on the flexible fingers. When a toy falls to the ground, the robot utilizes a camera to identify the object and then moves the gripper closer by adjusting the chassis and arm. The air valve is controlled by an electrical signal from a relay through the Raspberry Pi's GPIO, which opens and closes to manipulate

the internal air pressure. This air valve allows the flexible fingers to bend inward or outward to grasp the toy effectively.

Due to the need for significant vertical range lifting, the use of a platform is incorporated in the robot's design. This upgraded platform enhances the robot arm's movement using a ball screw slide, enabling movement of approximately 45 cm upwards and downwards. The lifting platform makes retrieving objects from the ground easier and placing them on higher surfaces, such as sofas and coffee tables. The robot's upper platform surface has a depth camera and a touch-sensitive liquid crystal display to improve visual recognition and child interaction.

Figure 3. 3-pneumatic fingers-based gripper.

In the early design stages, the first generation of movable household robots featured an important design element: the distance between the camera and the ground was 92 centimeters. This height allowed the robot to capture objects in its environment from a high angle. The working distance of the robot arm is approximately 22.7 cm. The common mobile platform for item retrieval using a mechanical arm typically involves directly mounting the arm onto an Automatic Guided Vehicle (AGV). However, the combined volume of the robot arm and chassis is relatively large, making it less practical for confined spaces commonly found in home environments. Additionally, the structural expansion of the arm body poses challenges in terms of flexibility and adaptability. Furthermore, this design introduces potential safety hazards, particularly concerning domestic settings. For these reasons, such configurations are not well-suited for residential environments and require further refinement to ensure functionality and safety in home-based applications. Figure 4 shows the entire process involved on the robot grasping the object.

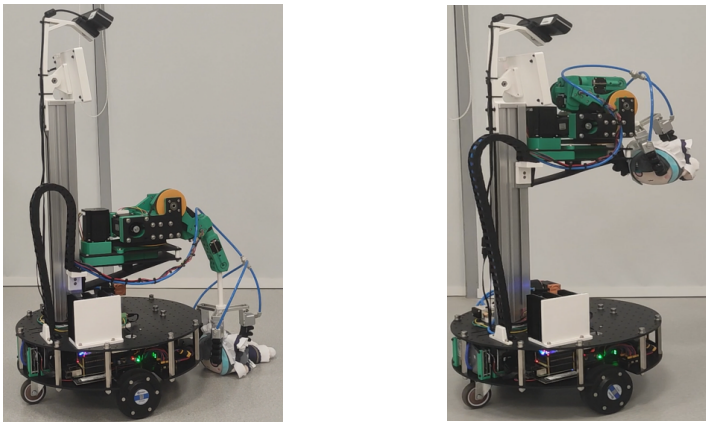


Figure 4. Objects gripping procedure.

To address the robot's operational requirements, which involve a greater demand for movement in the vertical direction and a relatively smaller demand for the horizontal plane range, we have introduced a lifting platform based on a ball screw sliding table along the vertical Z-axis. This modification enhances the longitudinal working range of the manipulator, allowing for an upper and lower movable range of approximately 45 cm. This adjustment enables the robotic arm to effectively reach items on sofas, tables, and other surfaces of similar height. Furthermore, the expanded vertical mobility enhances the robot's ability to interact with users more effectively, adapting to various scenarios in household environments.

To enhance target detection accuracy, it was crucial to prevent the robot arm's movement from obstructing the depth camera. Therefore, the depth camera was placed on the robot's chassis, specifically at a height of approximately 23 centimeters. This position provides a better viewing angle for a specific tilt. This adjustment also simplifies the subsequent process of recognizing the target object through coordinate transformation. The design concept of the second generation of moveable household robot has changed significantly as technology has advanced and user needs have changed. From the initial intelligent mobile tool, it becomes a more intelligent and interactive family member. In Figure 5, we can see the progress of this generation of robots.

(a) Different viewing angle ranges of the camera (b) The location of the updated camera

Figure 5. Hardware design of the proposed household robot.

3. Software implementation

3.1. Environment configuration

The robot uses highly reliable buses such as RS485 and CAN to control the robot's mechanical structure, reducing the risk of mechanical damage due to bus communication errors. The Raspberry Pi is a low-power and low-cost computer for the robot, as it operates on the Linux operating system and provides programmable GPIOs for hardware expansion. The GPIO interface is used to control each drive and sensor individually. At the same time, the robot communicates with the rest of the various modules using various methods, such as USB and GPIO switches. The Raspberry Pi is a single master controller for all devices, reducing system complexity. The Raspberry Pi also makes deploying subsequent upgrades and new features on the robot more straightforward and efficient. Furthermore, the Raspberry Pi can support deep learning algorithms compatible with object detection algorithms. ROS2, an open-source software development kit for robot applications, has been selected as the development environment. This kit offers a standardized software platform for robot applications [12].

Robots frequently employ vision sensors to detect information in the context of perception [13].

Traditional RGB color cameras and RGB-D depth cameras are the most frequently used devices for object detection. The distance between the object and the camera is unknown, as 2D cameras are limited to capturing objects at the viewing angle. Additionally, the intrinsic data of the model is the sole means by which the distance from the object can be determined. Consequently, a depth-integrated 3D sensing camera with high accuracy and reliability has been selected. This camera can calculate the X, Y, and Z coordinates of each point from the object to the camera, which is necessary for object detection, robot navigation, and grasping using robotic arms. Selecting distinct versions makes it straightforward to switch between short-range, long-range, and high-resolution RGB cameras to accommodate individual requirements.

3.2. Model configuration

Object detection is the core of computer vision in intelligent systems [14–16]. Its primary objective is to identify a particular target and its location within a real scene or input image and to assign a pre-labeled category to each detected object. Deep learning object detection algorithms are classified into two series: RCNN (Region-based Convolutional Neural Network) and YOLO (You Only Look Once). RCNN is a deep learning model primarily designed for target detection tasks. It combines CNNs (Convolutional Neural Networks) with region proposal methods to detect objects in images effectively [17]. The introduction of RCNN represents a significant breakthrough in deep learning for target detection, substantially improving detection accuracy.

The YOLO algorithm has been a revolutionary technology in computer vision since its inception. It is well-known for its fast, accurate, and efficient target detection capabilities. Regarding RCNN and YOLO, the former introduces a cyclic structure, which requires computation at each time step. Consequently, the computational complexity of the model is high, leading to relatively slow training and inference speeds. In contrast, the algorithms of the YOLO series excel in processing speed and are capable of real-time target detection. Additionally, the YOLO algorithm boasts a simple structure and efficient computational performance, making it highly suitable for deployment on mobile devices or embedded systems. Thus, we selected the YOLO for the proposed household robot.

Over time, YOLO has gradually evolved from YOLOv1 to YOLOv12 [18], with each version making significant improvements and innovations to the original. These iterations not only improved detection accuracy but also broadened the application scenarios. Among the two versions, YOLOv5 and YOLOv11, the former has received widespread attention for its rich educational resources. These resources provide researchers with valuable data for training more powerful models to solve complex image recognition problems. The main advantage of YOLOv5 is its significant improvement in target detection. It achieved an average accuracy of 50.5% on the COCO dataset, marking a massive leap in recognition accuracy. Compared to its predecessor, YOLOv5 performs even better in handling small objects.

YOLO updated its newest version, v11, in September 2024, marking another significant advancement in image recognition technology. After conducting a careful comparative analysis of YOLOv5, YOLOv8, and YOLOv11, we arrived at a key finding: YOLOv11 reduces the number of parameters in the model while maintaining an excellent balance between accuracy and performance. Among the versions of YOLOv11, the YOLOv11m variant stands out for its higher mean average precision (mAP) score on the COCO dataset. Notably, YOLOv11m uses 22% fewer parameters than YOLOv8s while demonstrating a substantial improvement in computational efficiency. This improvement makes YOLOv11m highly suitable for both real-time applications and resource-constrained environments. Considering these

advantages, we decided to replace YOLOV8s with YOLOv11m to achieve better results. This decision results from an in-depth evaluation of model performance and a forward-looking consideration of future application scenarios.

3.3. Object detection

The control structure has been divided into different functional modules with specific steps to achieve object detection and follow up with a more organized structure for different functions. The complete detection framework is shown in Figure 6.

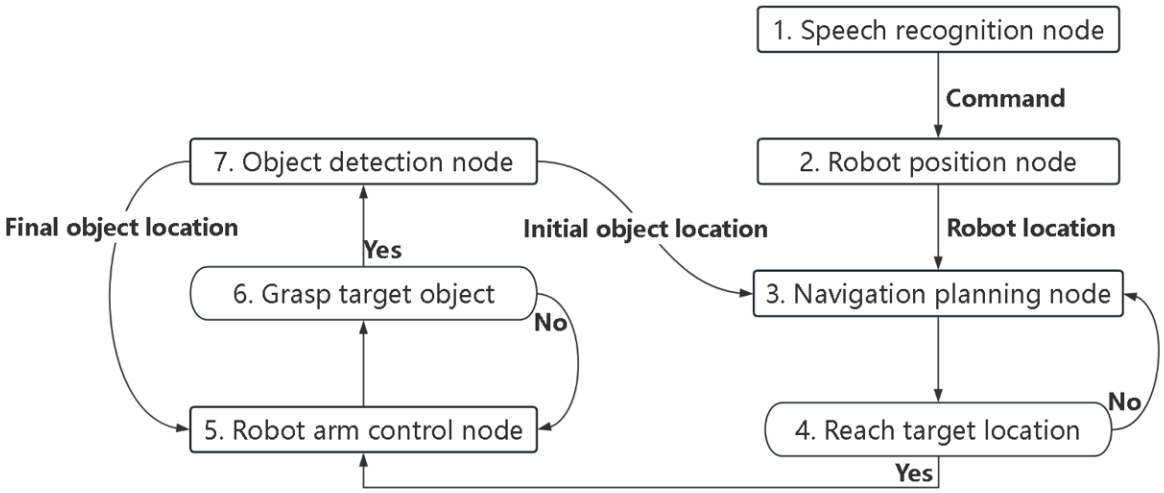


Figure 6. Overall control framework.

Step 1: The robot position node publishes its location into the robot system through radar.

Step 2: The object detection node publishes the initial object location to the robot system through the depth camera.

Step 3: The navigation node receives the information Steps 1 and 2 sent. The robot starts autonomous navigation.

Step 4: The robot determines whether it has arrived at the intended location. It advances to the subsequent task upon reaching the designated location. Otherwise, the navigation is repeated.

Step 5: After reaching the target location, the robot controls the robot arm. The robot arm control node accepts the latest location information of the target object from the object detection node and grasps it.

Step 6: The robot determines whether it has successfully grasped the object. Otherwise, the task should be re-executed. If it is successfully grasped, this pickup will enter a new cycle.

The visual recognition process in Figure 7 begins with the depth camera, which captures the depth information of the scene. The depth camera acquires both the original image and uses the intrinsic parameters of the camera, such as focal length and optical center, which are critical for subsequent image processing and coordinate conversion. Since the camera lens may introduce distortion, the raw image is processed at this stage to eliminate or reduce distortion and improve image quality. After distortion correction, the processed depth image is made available for visualization or further analysis. Object Detection is performed on the CPU using the YOLOv11 algorithm, an advanced target detection model capable of recognizing multiple objects in an image and determining their locations and classes.

A critical step in this process is converting the detected object coordinates from the image coordinate system to the 3D spatial coordinate system. This step is essential for applications requiring precise spatial information, such as robotic manipulation and augmented reality. The converted object position and category information are then published through the ROS2 system, enabling user integration with other systems or access. Finally, the object detection results are visualized, typically in an image in which detected objects are framed and labeled with their corresponding categories.

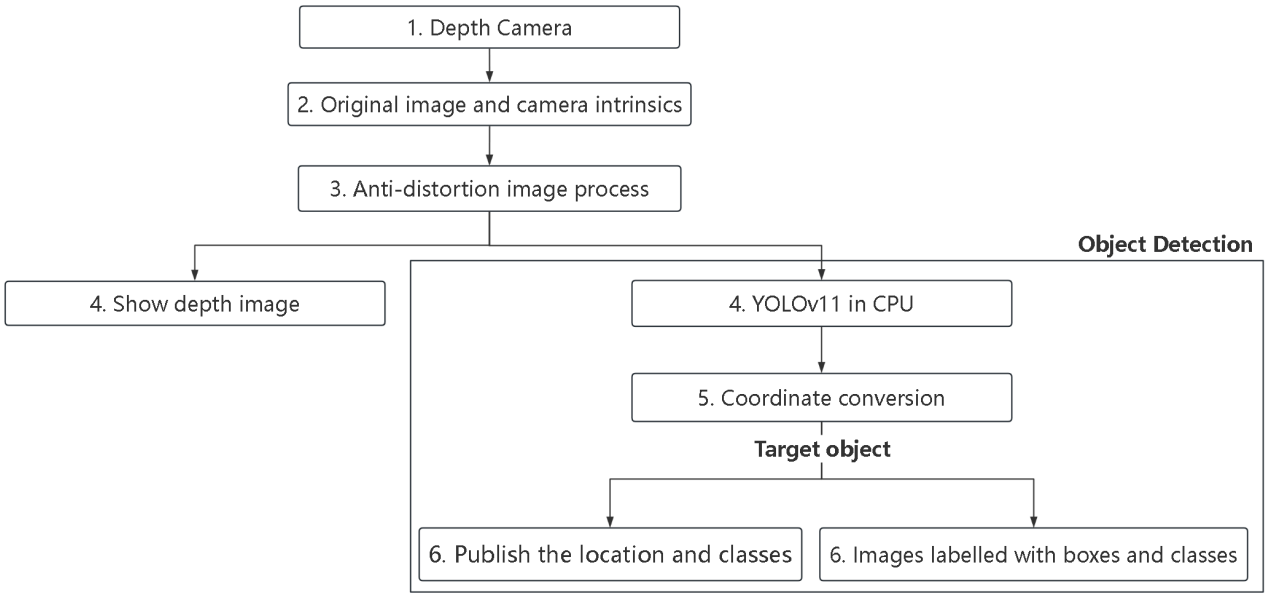


Figure 7. Framework of depth camera-based computer vision for this robot.

The depth camera captures high-precision images and utilizes the YOLOv11 object detection algorithm. This algorithm significantly enhances the accuracy of target object detection through advanced deep-learning techniques. In practice, YOLOv11 enables detailed labeling within bounding boxes (b-boxes). Measurements become intuitive by combining the depth camera with the YOLOv11 algorithm. During the testing process, we present preliminary results, as shown in Figure 8. A crucial aspect of this process is extracting useful information from the raw image data. We calculate the "bias" to achieve this by determining the depth relationship between the center point coordinates and the pixels in each bounding box. This bias is not fixed. Instead, a randomly selected location serves as the starting point and gradually expands in all directions, forming a list with multiple depth values. The goal is to enable the system to analyze each detected data point better. This entire process illustrates the refinement and intelligence of data processing, ensuring both the efficiency of image processing and the accuracy of the final results.

Bubble sort is the primary algorithm for object sorting. Median filtering is essential for image processing, as it enhances image smoothing by selecting the intermediate value as the new pixel value. Median filtering reduces noise while preserving the image's edge information. Median filtering is more effective at preserving the edge information in the image while smoothing it than linear filtering methods. As a result of the bubble sort, the pixel values' size order is altered without introducing new grey levels. Additionally, the median filtering algorithm is computationally efficient and straightforward. Even though the bubble sort necessitates many comparisons and exchanges during the sorting process, it remains real-time when the number of pixels in the window is low. Therefore, the depth values in this list are sorted using bubble sort and subsequently subjected to median filtering.

(a) Original image (b) YOLO labelled results (c) Depth image

Figure 8. Vision result testing.

3.4. Coordinate conversion

Coordinate system conversion in STEP4 is converting a camera coordinate system to an image coordinate system. Image processing typically involves four coordinate systems: the real-world coordinate system, the camera coordinate system, the image coordinate system, and the pixel coordinate system [19]. Mathematical transformations are implemented in each of these coordinate system conversions. Using the principle of similar triangles to establish the relationship function [20], the camera coordinate system to pixel coordinate system is a perspective projection relationship. The conversion process from 3D to 2D is represented by coordinates, as illustrated in Figure 9. This information can be further processed using an algorithm.

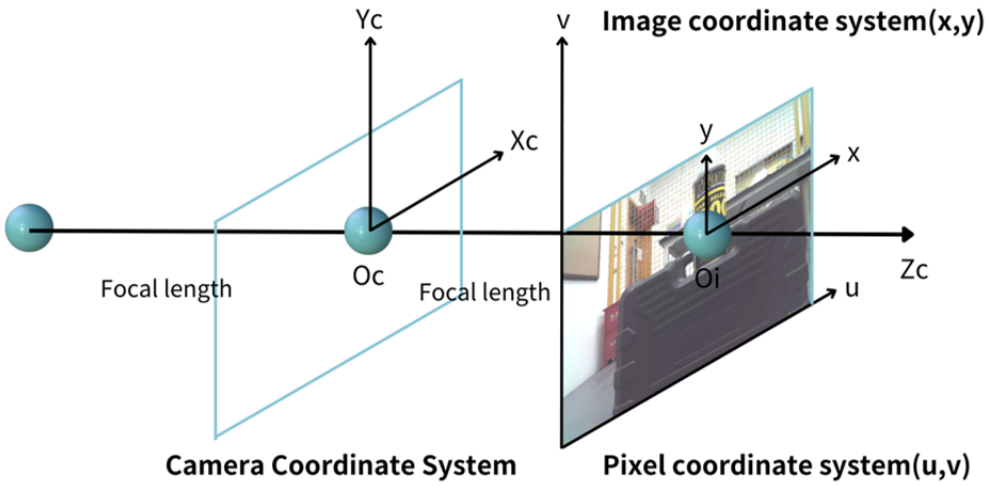


Figure 9. Coordinate conversion process.

In the field of vision, the camera plays a crucial role. Its primary function is to capture and process images through a computer to produce results. The optical center, which is the geometric center of the camera lens, is essential for the camera's imaging capabilities. We need a matrix of internal parameters to describe the relationship between the scene observed by the camera and its internal structure. This matrix comprises a specific set of elements representing the camera's interior's geometric parameters and physical properties. The internal parameter matrix not only reflects the camera's characteristics but can also be used to enhance the image processing workflow. Different cameras possess unique internal parameter matrices based on their design. Therefore, the right camera is essential to ensure image quality and accuracy.

When analyzing the connection between the camera and image coordinate systems, we identify some fundamental axes common to both. For instance, the X_c and Y_c axes in the camera coordinate system are parallel to the x and y axes in the image coordinate system. The camera's optical axis is also designated as the Z axis, which serves as a bridge between the two coordinate systems. The mapping from the camera coordinate system to the image coordinate system is typically performed using the right-hand rule based on the focal length's magnitude. Points are still measured in the camera coordinate system regarding pixels, where each pixel corresponds to a position in the image. The size and location of these pixel points are quantified in pixel units. Since the image coordinate system is measured in millimeters and the pixel coordinate system is expressed in pixel points, a proper conversion of the central point is essential.

Combined with the depth camera, we can further simplify the above process. We can express the transformation from the camera coordinate system to the image coordinate system in matrix multiplication. This form can intuitively represent the interactions among various components in the camera system, making image processing more accurate and efficient. Specifically, this transformation relationship can be formulated as the following Equation (1):

$$Z_C \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = [K|0] \begin{bmatrix} X_C \\ Y_C \\ Z_C \\ 1 \end{bmatrix} \tag{1}$$

Where K is the camera's internal reference matrix, it is inherent to the depth camera.

3.5. Mapping

The robot control system includes navigation algorithms and simultaneous localization and mapping (SLAM). LiDAR is the primary method for constructing a 2D planar map of the environment. The robot autonomously moves to the destination point after utilizing Cartographer [21,22] to achieve its localization and path planning within the map's range. Furthermore, when moving, the robot automatically avoids obstacles such as scattered toys and nearby children.

LiDAR is an essential sensor in robotic applications. It measures the distance between an object and a receiver by emitting a laser beam. LiDAR technology can primarily be categorized into two main types, which are further classified based on their functions and application scenarios in Table 2.

Table 2. Comparison of different kinds of LiDAR.

Categories	Effects	Usage Scenarios	Prices
2D LiDAR	Scanning plane	Commonly used in SLAM	Hundreds to thousands RMB
3D LiDAR	In 3D coordinate system	Commonly used outdoors	Generally over ten thousand RMB

The layout of an indoor environment is typically straightforward, making 2D LiDAR sufficient for effective navigation and obstacle detection in fixed indoor settings. Additionally, 2D LiDAR is generally more cost-effective compared to 3D LiDAR. Given that our robot's operational environment is within a home, we concluded that the functionality of 2D LiDAR is adequate for indoor applications. Considering its capabilities and cost advantages, we selected 2D LiDAR for the household robot.

However, when a robot relies solely on LiDAR data for navigation, it encounters a significant

challenge: accurately determining its precise location. This difficulty arises because the data provided by LiDAR is not always reliable. Ambient noise can disturb the data, and terrain changes can introduce bias. As a result, the map's construction and the robot's localization may drift a little, causing them to appear stationary while they are actually moving. Consequently, the accuracy of the localization could be influenced.

Additional data is needed to accurately determine a robot's location and prevent localization drift to enhance LiDAR localization. The robot uses a data fusion strategy incorporating LiDAR, a wheeled odometer, and an inertial measurement unit (IMU) to mitigate cumulative errors. Using sensor data beyond LiDAR, the odometer can improve localization accuracy by tracking the robot's relative displacement, orientation, and trajectory. Precisely, the robot measures its speed, distance, and direction by collecting readings from the encoder of the wheeled odometer, which allows it to plot its relative displacement on a plane. Additionally, to reduce data jitter, the direction data from the wheel odometer is supplemented with data from the IMU. After implementing filtering algorithms, this approach yields more stable and accurate results.

After conducting a thorough technical comparison and analysis, we chose the Cartographer algorithm [23] for our mapping needs. This algorithm is designed for 2D LiDAR measurement and localization, making it a mature option with stable and reliable performance. The algorithm has been successfully integrated into ROS2's Navigation2 framework, a widely used software library in the driverless vehicle field. By adjusting the necessary parameter files, the Cartographer algorithm can seamlessly embed into the proposed robot's driving systems.

The navigation algorithm can generate a predetermined behavioral tree that determines the expected path from the vehicle's current position to the destination on the pre-generated map. It then issues a continuous movement command to the motor controller. This path circumvents impassable areas, including walls, unknown areas, user-marked no-go zones, and other obstacles detected on the map in real-time and updated. This component of the robot functions as a sequence of ROS2 nodes. Upon receiving the most recent coordinates of the goal point from the visual recognition module, this navigation module updates the destination position and autonomously advances toward the goal. The navigation module broadcasts the successful arrival message to the other nodes in the system that are awaiting the following command once it has successfully reached the target.

3.6. Robot movement control

The robot performs self-localization at startup through a meticulously coordinated calibration between the LiDAR and depth camera. This process involves aligning the robot's sensor data with a pre-constructed map to determine its position within the environment accurately. Additionally, SLAM allows the robot to navigate through unfamiliar environments while continuously updating its map in real-time.

After completing its initial localization, the robot utilizes the depth camera to identify and estimate the precise position of the target object relative to itself. The resolution is chosen as 640×480, and the accuracy of the robot's estimation of the target object's position is approximately between 5 millimeters and 2 centimeters. This step is crucial for subsequent path planning, ensuring the robot can accurately approach the target object's location. During the path planning phase, the robot system ensures that the target object is positioned at a specific distance directly in front of the robot for efficient interaction. The path planning algorithm incorporates considerations such as obstacle avoidance and other environmental

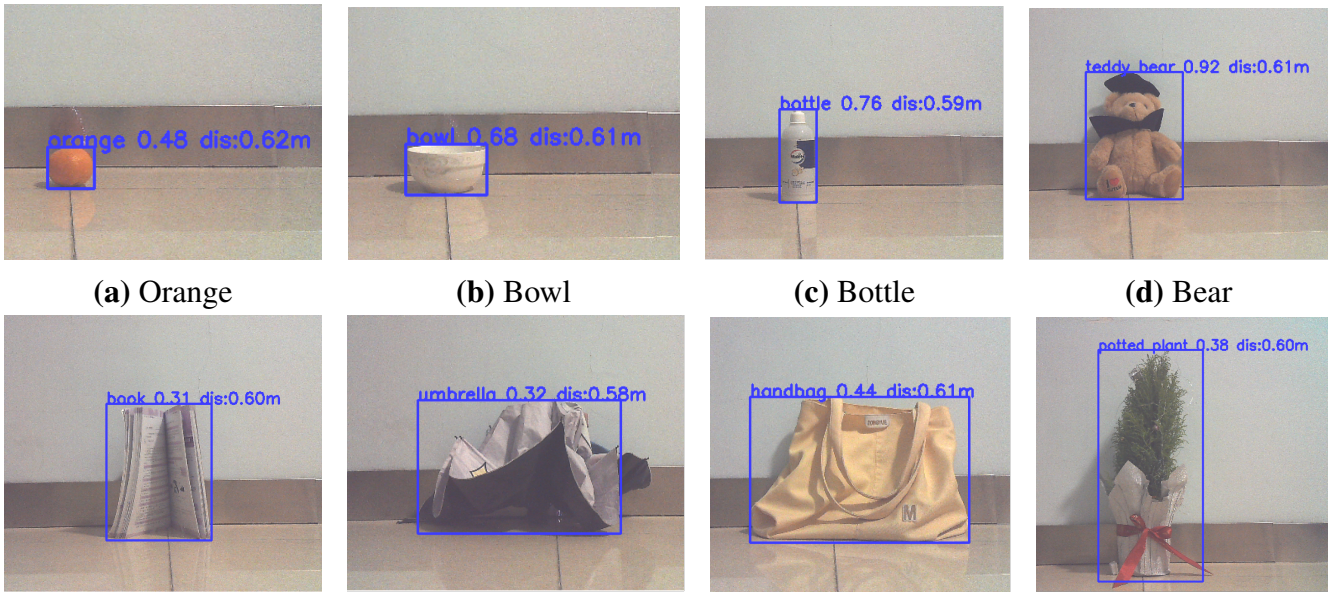
factors to guarantee both the safety and efficiency of the robot's movements.

When the robot reaches the target location along the planned path, the lifting platform and robotic arm are activated and ready for grasping. The robot arm uses open-loop control. By controlling the rotation of the 2-axis, 3-axis and 4-axis motors of the robot arm, the end of the arm is extended beyond the range of the robot chassis and accurately positioned directly above the target object. The elevated platform begins to descend, placing the gripper portion of the robotic arm into the target object. The gripper control node turns on the air pump to inflate and close the gripper to securely grasp the target object. Throughout the process, the robot's sensors and control system work closely to ensure every step. From localization to path planning to gripping, the robot is executed precisely.

4. Result and analysis

4.1. YOLO algorithm recognition capability

Qualitative analysis has a vital place in the process of detecting objects. This analysis is usually based on experiments and relies on our subjective judgment to determine which objects are successfully identified. Observing the detection results gives us access to a large amount of valuable data. This data is essential for the training and optimization of machine learning models. In order to better simulate scenarios typical of home life, we selected a relatively closed and spacious indoor environment for testing. We conducted a series of object detection and grasping-related experiments in this environment on eight everyday household objects using the YOLOV11 algorithm. Thus, these objects included an orange, a bowl, a bottle, a bear, a book, an umbrella, a handbag, and a potted plant. To thoroughly evaluate the robot's grasping ability, we carefully measured and recorded the size and weight of each object. Based on the data presented in Figure 10 and Table 3, it is evident that the robot demonstrates the capability to grasp objects of varying sizes and weights effectively. This capability highlights the adaptability of the robotic system in handling diverse household items.



(a) Orange **(b)** Bowl **(c)** Bottle **(d)** Bear

(e) Book **(f)** Umbrella **(g)** Handbag **(h)** Potted plant

Figure 10. Experimental tests of YOLOv11.

Table 3. Dimensions and weights of objects for detection.

Categories	Dimensions			Weight
	L	W	H	
Orange	0.06 m	0.06 m	0.04 m	107.5 g
Bear	0.19 m	0.14 m	0.27 m	117.7 g
Potted Plant	0.17 m	0.17 m	0.52 m	1054.6 g
Bowl	0.12 m	0.12 m	0.06 m	186.4 g
Umbrella	0.32 m	0.07 m	0.24 m	304.2 g
Handbag	0.41 m	0.09 m	0.28 m	305.5 g
Book	0.18 m	0.25 m	0.01 m	515.5 g
Bottle	0.06 m	0.06 m	0.26 m	112.4 g

Table 4 presents the measured distances and accuracy of different object categories at three distinct distances (0.6 m, 0.8 m, and 1 m). In the YOLO algorithm, accuracy is a comprehensive probabilistic metric. It refers to the precision with which the YOLO model predicts the category of target objects rather than indicating the distance or size of the objects. The YOLO algorithm generates many prediction bounding boxes during object detection when processing an image. Accuracy can exclude those bounding boxes less likely to contain the target objects. For instance, hundreds of prediction bounding boxes may be generated in a complex image detection scenario. By setting an accuracy threshold, bounding boxes with an accuracy below this threshold are preliminary filtered out. This approach reduces the computational load for subsequent processing and lowers the false positive rate. Among the tested objects, the "Bear" achieved the highest accuracy, which maintained consistently high accuracy across all distances. Similarly, the "Bottle" demonstrated relatively high accuracy at 0.6 meters. These findings suggest that the accuracy of bears and bottles is relatively stable across different distances, while the accuracy of other categories exhibits fluctuation.

Table 4. Measured distance and accuracy of objects.

Categories	Distance: 0.6 m		Distance: 0.8 m		Distance: 1 m	
	Measured distance	Accuracy	Measured distance	Accuracy	Measured distance	Accuracy
Orange	0.62 m	0.47	0.82 m	0.27	0.98 m	0.31
Bear	0.61 m	0.92	0.82 m	0.85	0.98 m	0.74
Potted plant	0.6 m	0.38	0.77 m	0.37	0.99 m	0.28
Bowl	0.61 m	0.68	0.82 m	0.67	0.98 m	0.54
Umbrella	0.58 m	0.32	0.78 m	0.41	0.98 m	0.39
Handbag	0.61 m	0.44	0.79 m	0.43	0.92 m	0.3
Book	0.6 m	0.31	0.8 m	0.26	0.98 m	0.34
Bottle	0.59 m	0.76	0.83 m	0.44	0.98 m	0.44

The "Book," on the other hand, was relatively inaccurate at all distances. This reason may be attributed to its shape, which complicates accurate measurement by the depth camera. Books can be placed in various orientations in different scenarios, such as lying flat, standing upright, or leaning at an angle. When we conducted our tests, the books were either laid open or stood upright. We think that these different orientations can alter the shape and contour of the book as perceived by the depth camera, thereby increasing the difficulty of recognition and measurement. Despite these variations, the measured distances for all categories were generally close to the actual distances, albeit with some deviation.

A common trend observed was a decrease in accuracy for most categories as the distance increased, likely due to the inherent challenges in precise measurements over longer distances. Because the robot detects the objects in real-time, as the robot approaches the objects, the accuracy improves, ensuring that the object recognition and grasping functions run successfully.

In Table 5, we compare the accuracy and measured distances of two object detection models, YOLOv8 and YOLOv11, across four object categories (Bowl, Bottle, Umbrella, Handbag) and three measured distances (0.6 m, 0.8 m, 1 m). Notably, YOLOv8's accuracy in the Bottle category decreases at distances of 0.8 m and 1 m, indicating possible challenges in detecting this object at longer ranges. In contrast, YOLOv11 demonstrates higher accuracy in the Bowl and Handbag categories (0.68 and 0.44, respectively) than YOLOv8 (0.63 and 0.33). Based on the test results for these four objects, YOLOv11 consistently outperforms YOLOv8 or achieves comparable results in nearly all scenarios. To further analyze the models' overall performance, we calculated the average accuracy of each model across all categories and distances. The average accuracy for YOLOv8 is 0.487, while for YOLOv11, it is 0.495. These results indicate that YOLOv11 performs slightly better in these testing environments. Given these findings, we conclude that YOLOv11 performs superior object detection and localization under the tested conditions. As a result, we selected YOLOv11 for our robotic system.

Table 5. Comparison of YOLOv8 and YOLOv11.

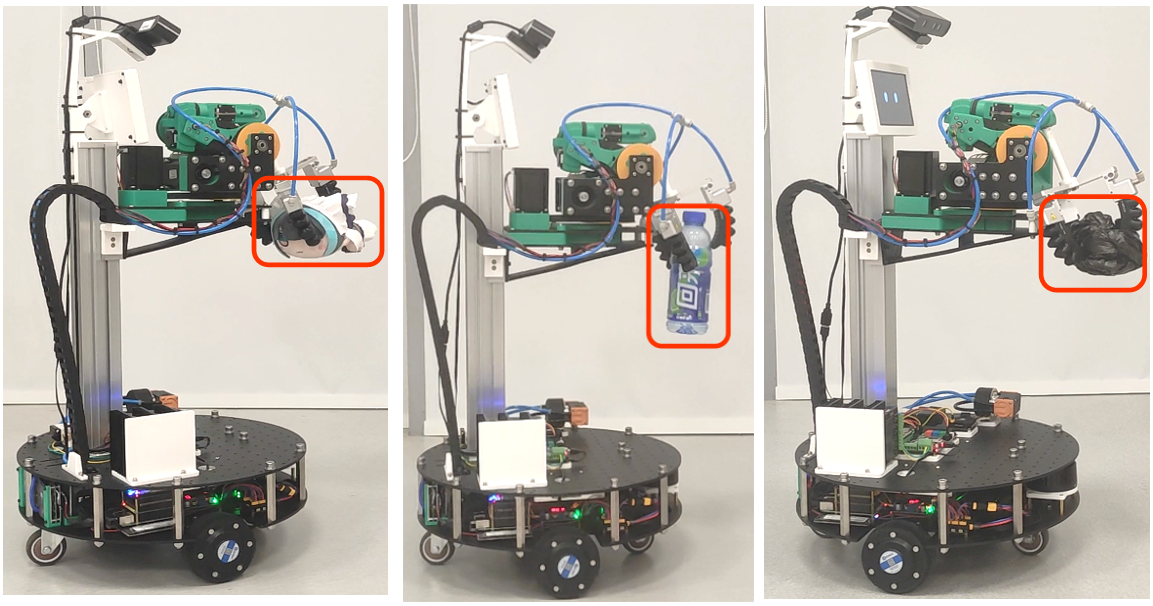
Categories		Bowl	Bottle	Umbrella	Handbag
Distance: 0.6 m ModelYOLOv8	Measured distance	0.59 m	0.59 m	0.64 m	0.61 m
	Accuracy	0.63	0.77	0.32	0.33
Distance: 0.6 m ModelYOLOv11	Measured distance	0.61 m	0.59 m	0.58 m	0.61 m
	Accuracy	0.68	0.76	0.32	0.44
Distance: 0.8 m ModelYOLOv8	Measured distance	0.75 m	0.75 m	0.8 m	0.72 m
	Accuracy	0.39	0.5	0.43	0.55
Distance: 0.8 m ModelYOLOv11	Measured distance	0.82 m	0.83 m	0.78 m	0.79 m
	Accuracy	0.67	0.44	0.41	0.43
Distance: 1 m ModelYOLOv8	Measured distance	0.98 m	0.98 m	0.91 m	0.86 m
	Accuracy	0.27	0.47	0.34	0.32
Distance: 1 m ModelYOLOv11	Measured distance	0.98 m	0.98 m	0.98 m	0.98 m
	Accuracy	0.54	0.44	0.39	0.3

4.2. Robot grasping capability

The initial generation of mobile home robots has already demonstrated the ability to grasp objects at a fixed point, as illustrated in Figure 11.

In Table 6, we have listed the dimensions and weights of the three objects shown in Figure 11. The doll measures 0.15 m in length, 0.15 m in width, and 0.23 m in height. The bottle measures 0.07 m in length, 0.07 m in width, and 0.16 m in height. The trash bag measures 0.18 m in length, 0.15 m in width, and 0.12 m in height. These objects vary in size. The robot's ability to grasp these objects indicates its adaptability to items with different lengths, widths, and heights and demonstrates its capability to handle

common household items within a certain size range.



Toys collection Bottles collection Waste collection

Figure 11. Objects grasping test.

Table 6. Dimensions and weights of objects for robot grasping.

Categories	Dimensions			Weight
	L	W	H	
Doll	0.15 m	0.15 m	0.23 m	150.3 g
Bottle	0.07 m	0.07 m	0.16 m	250 g
Garbage Bag	0.18 m	0.15 m	0.12 m	10 g

Among the three objects, the bottle is the heaviest at 250 g, followed by the doll at 150.3 g, while the trash bag is the lightest at only 10 g. The robot's ability to grasp these objects with varying weights shows that it has a certain weight-carrying capacity and can handle common lightweight and small household items.

The doll is relatively regular in shape, the bottle is slender, and the trash bag is flat. These objects have different shapes. The robot's ability to grasp them indicates that its grasping mechanism is compatible with handling objects of various shapes which may be found in a home environment.

4.3. Speech recognition capability

In addition, the robot system is equipped with a speech recognition module, which is integrated into the robot's control framework to enable the recognition and processing of user voice commands. The Automatic Speech Recognition (ASR) system is the core module that enables efficient interaction between users and the robot. To more clearly illustrate the system's workflow, we have supplemented Figure 12, which primarily includes stages such as voice input, preprocessing, feature extraction, recognition, and result output. In Figure 12, the speech is input into the system and then preprocessed to remove noise and other interfering factors. Subsequently, feature extraction is performed on the preprocessed speech to obtain parameters that represent the characteristics of the speech. The extracted features are used for

training, with a portion serving as reference templates in the template library. Meanwhile, test templates are generated based on the speech features of the recognition signals. After matching with the reference library, the template with the highest score is output as the recognition result.

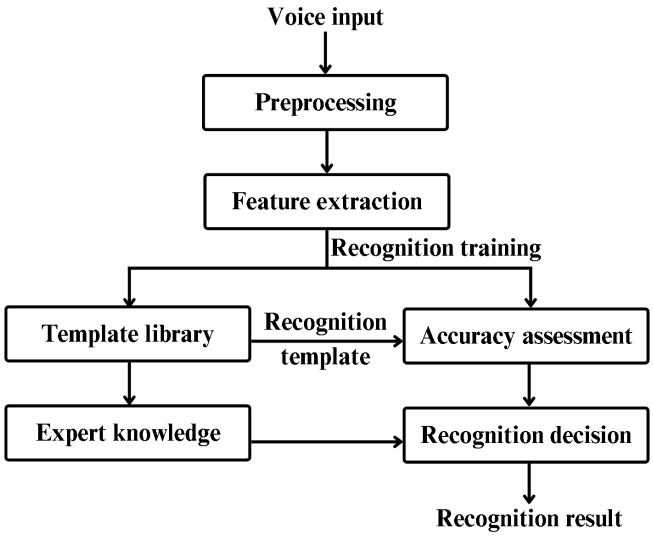


Figure 12. The process of speech recognition.

The robot's voice interaction flow design is tailored to meet specific requirements and functionalities, incorporating elements such as user commands, the robot's responses, and the execution of corresponding actions. This structured interaction ensures seamless communication between the user and the robot. The integration of the speech recognition module enhances the system's usability, allowing users to control the robot with a high degree of accuracy and efficiency.

To comprehensively evaluate the performance of the speech recognition system, we designed a series of experiments focusing on recognition accuracy, recall rate, and user experience, among other aspects. The experimental environment simulated daily home scenarios, including different areas such as the living room and kitchen, and considered factors such as varying background noise and speaking distances. The participants in the experiment comprised a small group of five individuals ranging in age from 16 to 55 years old, ensuring that the experimental results could reflect the impact of speech interaction across different age groups. During the experiment, participants were asked to repeat the same commands 30 times. We conducted experiments for each command and calculated the accuracy of speech recognition.

Each participant issued a series of predefined voice commands to the robot under different experimental conditions. The commands were related to common household tasks, such as "turn left" or "follow". Upon receiving the voice commands, the robot processed them through its built-in speech recognition system and provided feedback to the participants. The participants then judged the accuracy of the recognition and recorded the relevant data. Additionally, we collected subjective evaluations from the participants regarding their experience with the speech interaction, including aspects such as recognition speed, response accuracy, and interaction friendliness. We define recognition accuracy as "whether the robot accurately completes the commands issued by the user." The measurement method is as follows: "Each tester repeats the same command 30 times. We measure the recognition accuracy by repeating each command and calculating the proportion of accurate responses." User interest is "the

user's emotional state after the test and willingness to participate in future tests." We use a five-point scale (one representing very dissatisfied and five representing very satisfied) to measure user interest. Feedback is collected from participants at the end of the experiment.

We employed metrics such as accuracy, recall rate, and F1 score to assess the performance of the speech recognition system. Precision, Recall, and F1 are commonly used metrics when evaluating a speech recognition system's performance.

$$Precision = \frac{TP}{TP + FP} \tag{2}$$

where TP is the number of correctly recognized samples, and FP is the number of samples incorrectly identified as positive. In a quiet environment, the system's average recognition accuracy is 85%, which means that 85% of the recognized samples are correctly identified.

$$Recall = \frac{TP}{TP + FN} \tag{3}$$

where FN is the number of missed samples (false negatives).

$$F1 = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{4}$$

which provides a balanced measure of precision and recall.

Based on statistical analysis of the experimental data in Table 7, the system achieved an average recognition accuracy of 83.8%, a recall rate of 80%, and an F1 score of 81.8% in quiet environments. In environments with background noise, the accuracy slightly decreased but remained above 75%. In a quiet environment, the system's average recognition accuracy is 83.8%, indicating that 83.8% of the recognized samples are correct. In environments with background noise, the accuracy drops slightly but remains above 75%. This data suggests that while noise impacts system performance, the system still demonstrates a certain level of robustness. Initially, we considered that individuals of different ages might vary in terms of speaking rate and familiarity with commands. Therefore, we deemed it necessary to test whether age would have an impact on the performance of the speech recognition system. However, after further understanding the working principles of the speech recognition system, which primarily relies on analyzing acoustic features and semantic information of speech for recognition, we realized that age should not theoretically have a direct impact on these aspects of speech recognition. Based on the subjective evaluations provided by the participants, most users highly praised the convenience of the speech interaction. However, some issues were noted, such as occasional recognition errors in noisy environments and misunderstandings of specific commands.

Our robot is currently in the prototype stage. Experiments have been conducted primarily in controlled environments, such as laboratories, to ensure accurate evaluation and performance testing. However, due to the inherent complexity of the system and the potential risks, full-scale testing in real-home environments has not yet been undertaken. We recognize that such real-world testing will be critical in refining and perfecting the robot's various functions. Future research will focus on continuously improving and optimizing the robot's design.

Table 7. Test results of speech recognition.

Environmental Condition	Recognition Accuracy (%)	Recall Rate (%)	F1 Score (%)	User Feedback	Common Issues
Quiet Environment	83.8	80.0	81.8	Most users highly praised the convenience of speech interaction	Occasional misunderstandings of specific commands may be related to accent or speaking speed
Noisy Environment	75.0	70.0	72.4	Some users reported occasional recognition errors in noisy environments	Occasional recognition errors and misunderstandings of specific commands in noisy environments

4.4. Merits and limitations

We believe that robot development must align with the demands for personalization, intelligence, cost efficiency, and high reliability. In the future, intelligent household robots are poised to become an indispensable part of human life.

Our proposed robot is designed to navigate indoor environments autonomously and collect loose objects based on a safe soft gripper. By leveraging object detection technology, the robot can identify various items and sort them into appropriate locations using a robotic arm. While the robot has a display screen, there remains considerable potential for enhancing human-robot interaction.

To address this, we aim to maximize the display's functionality to improve the robot's communication capabilities. For instance, we plan to integrate large-scale language models into the robot's language interaction module. These models, trained on extensive textual data, can acquire rich linguistic knowledge and semantic understanding, enhancing the robot's ability to comprehend, generate, and interact through language. This improvement will further bridge the gap between humans and robots, fostering more seamless and intuitive interactions. Currently, the robot is still in the prototype stage, and there is a gap between the current version and mass production. Therefore, at this stage, our primary focus for the speech recognition function was to test its effectiveness. We plan to leverage large language models to further enhance the robot's speech recognition capabilities in future work.

5. Conclusion

The proposed household robot aims to assist the new generation of parents and children by independently performing tasks such as storage and companionship, as specified by pre-programmed rules and instructions. The main contributions of this research can be summarized in three aspects. First, we completed the innovative hardware design by independently creating a lightweight chassis and fitting a flexible gripper. Second, we enhanced the robot's adaptability to complex home environments by optimizing camera placement to expand the field of view. Third, we successfully implemented full-function capabilities in the robot by integrating soft and hardware systems, accomplishing speech recognition, autonomous navigation, object recognition to grasping. A series of experiments confirmed

the algorithm's viability, providing a strong reference for domestic robot development. Future work will address the evolving demands of household environments by combining practical applications with ongoing analysis. Efforts will be directed toward enhancing object detection accuracy and integrating large language models into the speech module to improve the robot's semantic understanding capabilities.

Acknowledgments

This work was supported in part by the Teaching Development Fund of XJTLU under Grant TDF20/21-R22-144 and TDF22/23-R25-198, and in part by the XJTLU Research Development Fund under Grant RDF-22-01-081, and in part by the XJTLU Research Enhancement Fund under Grant REF-21-02-001.

Authors' contribution

Conceptualization, methodology, validation, formal analysis, data curation, writing—original draft preparation: Bingjie Xu, Yangzesheng Lu, and Jingxiang Wang; writing—review and editing, supervision: Qinglei Bu, Mark Leach and Jie Sun. All authors have read and agreed to the published version of the manuscript.

Conflicts of interests

The authors declare no conflict of interest.

References

[1] Liu Y, Lu Y, Peng C, Bu Q, Liang YC, *et al.* Autonomous vehicle based on ROS2 for indoor package delivery. In *2023 28th International Conference on Automation and Computing (ICAC)*, Birmingham, UK, August 30–September 1, 2023, pp. 1–7.

[2] Lu Y, Liu Y, Bu Q, Lim EG, Devaraj R, *et al.* An autonomous vehicle platform for parcel delivery. In *2023 28th International Conference on Automation and Computing (ICAC)*, Birmingham, UK, August 30–September 1, 2023, pp. 1–7.

[3] Patruno C, Renò V, Nitti M, Mosca N, di Summa M, *et al.* Vision-based omnidirectional indoor robots for autonomous navigation and localization in manufacturing industry. *Heliyon* 2024, 10(4):e26042.

[4] Wang Y, Chen Z, Li H, Cao Z, Luo H, *et al.* Movevr: enabling multiform force feedback in virtual reality using household cleaning robot. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, Online, April 25–26, 2020, pp. 1–12.

[5] Newman BA, Paxton CJ, Kitani K, Admoni H. Towards online adaptation for autonomous household assistants. In *Companion of the 2023 ACM/IEEE International Conference on Human-Robot Interaction*, Stockholm, Sweden, March 13–16, 2023, pp. 506–510.

[6] Yin J, Apuroop KGS, Tamilselvam YK, Mohan RE, Ramalingam B, *et al.* Table cleaning task by human support robot using deep learning technique. *Sensors* 2020, 20(6):1698.

[7] Ribeiro T, Gonçalves F, Garcia IS, Lopes G, Ribeiro AF. CHARMIE: a collaborative healthcare and home service and assistant robot for elderly care. *Appl. Sci.* 2021, 11(16):7248.

[8] Palacín J, Rubies E, Clotet E. The assistant personal robot project: from the APR-01 to the APR-02 mobile robot prototypes. *Designs* 2022, 6(4):66.

[9] García GA, Pérez G, Laycock-Narayan RK, Levinson L, Amores JG, *et al.* Preliminary study on the feasibility of approximating children's engagement level from their emotions estimation by a picture-based, three-model AI in a family-robot cohabitation scenario. *Adv. Rob.* 2024, 38(23):1710–1728.

[10] Lu Y, Xu B, Wang J, Bu Q, Leach M, *et al.* An autonomous household robot for object detection and grasping. In *2024 29th International Conference on Automation and Computing (ICAC)*, Sunderland, UK, August 28–30, 2024, pp. 1–7.

[11] Li H, Huang K, Sun Y, Lei X, Yuan Q, *et al.* An autonomous navigation method for orchard mobile robots based on octree 3D point cloud optimization. *Front. Plant Sci.* 2025, 15:1510683.

[12] Maruyama Y, Kato S, Azumi T. Exploring the performance of ROS2. In *Proceedings of the 13th international conference on embedded software*, Pittsburgh, USA, October 1–7, 2016, pp. 1–10.

[13] Liu Z, Cai Y, Wang H, Chen L, Gao H, *et al.* Robust target recognition and tracking of self-driving cars with radar and camera information fusion under severe weather conditions. *IEEE Trans. Intell. Transp. Syst.* 2021, 23(7):6640–6653.

[14] Sun M, Xiao J, Lim EG, Zhang B, Zhao Y. Fast template matching and update for video object tracking and segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Seattle, USA, June 14–19, 2020, pp. 10791–10799.

[15] Sun M, Xiao J, Lim EG, Zhao Y. Starting point selection and multiple-standard matching for video object segmentation with language annotation. *IEEE Trans. Multimedia* 2022, 25:3354–3363.

[16] Sun M, Xiao J, Lim EG, Xie Y, Feng J. Adaptive ROI generation for video object segmentation using reinforcement learning. *Pattern Recognit.* 2020, 106:107465.

[17] Bhavya Sree B, Yashwanth Bharadwaj V, Neelima N. An inter-comparative survey on state-of-the-art detectors—R-CNN, YOLO, and SSD. In *Intelligent Manufacturing and Energy Sustainability: Proceedings of ICIMES 2020*, Hyderabad, India, August 21–22, 2021, pp. 475–483.

[18] Wang G, Chen Y, An P, Hong H, Hu J, *et al.* UAV-YOLOv8: a small-object-detection model based on improved YOLOv8 for UAV aerial photography scenarios. *Sensors* 2023, 23(16):7190.

[19] Kolin N, Chebotareva E. A comparative analysis of object detection methods for robotic grasping. In *2024 International Conference on Artificial Life and Robotics (ICAROB2024)*, Horuto Hall, Japan, February 22–25, 2024, pp. 304–307.

[20] Mei S, Liu C, Lv X. A falls recognition framework based on faster R-CNN and temporal action sequences that can be deployed on home service robots. *Meas. Sci. Technol.* 2024, 35(8):085005.

[21] Zhou L, Zhu C, Su X. SLAM algorithm and navigation for indoor mobile robot based on ROS. In *2022 IEEE 2nd International Conference on Software Engineering and Artificial Intelligence (SEAI)*, Xiamen, China, June 10–12, 2022, pp. 230–236.

[22] Jiang Y, Leach M, Yu L, Sun J. Mapping, navigation, dynamic collision avoidance and tracking with LiDAR and vision fusion for AGV systems. In *2023 28th International Conference on Automation and Computing (ICAC)*, Birmingham, UK, August 30–September 1, 2023, pp. 1–6.

[23] Documentation GCR. Compilation guide. 2025. Available: https://google-cartographer-ros.readthedocs.io/en/latest/compilation.html (accessed on 30 March 2025).