ECE 445

SENIOR DESIGN LABORATORY

FINAL REPORT

Final Report for ECE 445

Team #36

CHENGHAN LI
(cli104@illinois.edu)
YIPU LIAO
(yipul2@illinois.edu)
YIMING LI
(yiming20@illinois.edu)
HAORAN YANG
(haorany8@illinois.edu)

TA: Leixin Chang

May 9, 2024

Contents

1 Introduction

1.1 Problem and Solution Overview

China's aging population is facing a growing challenge with the increasing prevalence of geriatric diseases, notably Parkinson's disease (PD), a debilitating neurological condition predominantly affecting the elderly. The prevalence of PD among individuals over 65 years old in China is approximately 1.7%, with a projected surge to 5 million patients by 2030, representing nearly half of the global PD population. And the prevalence increases further with age, exceeding 4% in people over 80 years of age. This demographic shift underscores the urgent need for innovative solutions to alleviate the burden of PD and enhance the quality of life for affected individuals.

Parkinson's disease manifests through symptoms such as tremors, rigidity, bradykinesia, and postural instability, severely impeding daily activities and independence. Elderly PD patients often struggle with simple tasks due to motor impairments, necessitating accessible and intelligent assistive technologies to support their daily living needs.

In response to this pressing need, our project proposes a novel solution in the form of an intelligent robotic arm system. This system is designed to empower Parkinson's patients by providing assistance with various tasks through advanced speech and visual recognition capabilities, the process in shown in Figure 1. By leveraging cutting-edge technology, our solution aims to revolutionize the way individuals interact with their environment, enabling seamless task execution and fostering greater independence and autonomy.

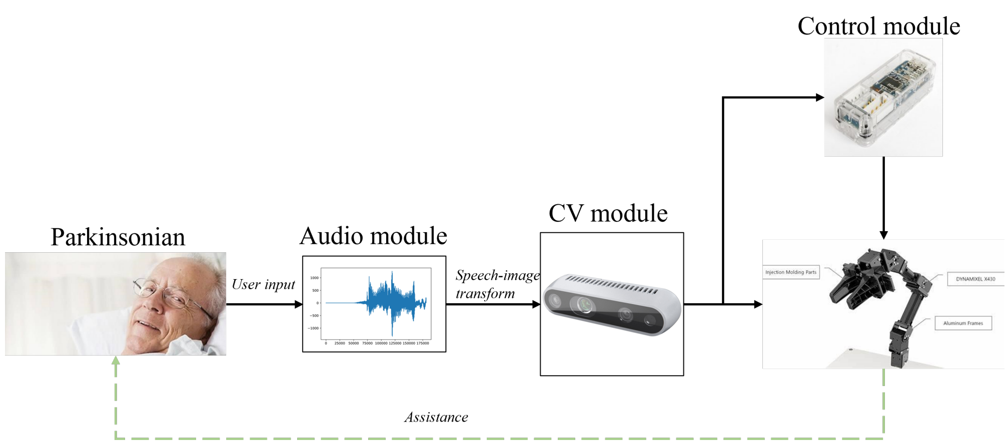


Figure 1: The whole system overview. The Parkinsonian will speak to the microphone, then the audio module will extract the object interested by the patient and then the CV module will find the location of the object and use the robotic arm to grab the object.

The specific system can be divided into three subsystems: language model, visual recognition, and robotic arm. These systems interact with each other and cooperate with each other in the process of completing the ordered tasks. First, the language model recognizes the operator's language instructions and converts them into computer signal instructions for transmission to the visual recognition system. The visual recognition system automatically recognizes the target object according to the instruction and the visual information from the camera as input, and generates the position information of the target object as output. Finally, after the robot arm obtains the position information from the visual recognition system, the corresponding motor moves initially, reaches the predetermined position and completes the command task.

The key innovation lies in the integration of a sophisticated language model, visual recognition system, and robotic arm subsystems, which work in tandem to interpret user commands, identify target objects, and execute precise actions. Through this cohesive integration, our intelligent robotic arm offers a comprehensive solution tailored to the specific needs of Parkinson's patients, enhancing their ability to navigate daily challenges with ease and confidence.

In summary, our project represents a significant advancement in assistive technology for Parkinson's disease, offering a practical and innovative solution to address the evolving needs of China's aging population. By harnessing the power of robotics and artificial intelligence, we aim to mitigate the impact of PD and empower individuals to lead fulfilling and independent lives.

1.2 High-level requirements list

Reliability: The system maintains a high level of reliability and recognition accuracy, and our goal is to maintain the accuracy of speech recognition and visual recognition above 90%.

Intelligence: The robot arm should be designed to recognize and execute enough commands to complete them. It can understand human natural language, and carry out relational extraction to extract the object of the sentence accurately.

payload: For this robotics arm, the max load on the end-effector is limited to 3 kg.

Operating voltage: power supply should be capable of supplying 24 V and 15 A or higher.

Motor type: Brushless DC Servo(H54P Series), Coreless DC Motor(H42P Series)

Communication Baudrate: 1000000 bps

1.3 Subsystem Overview

The block diagram of our whole system is shown in Figure 2.

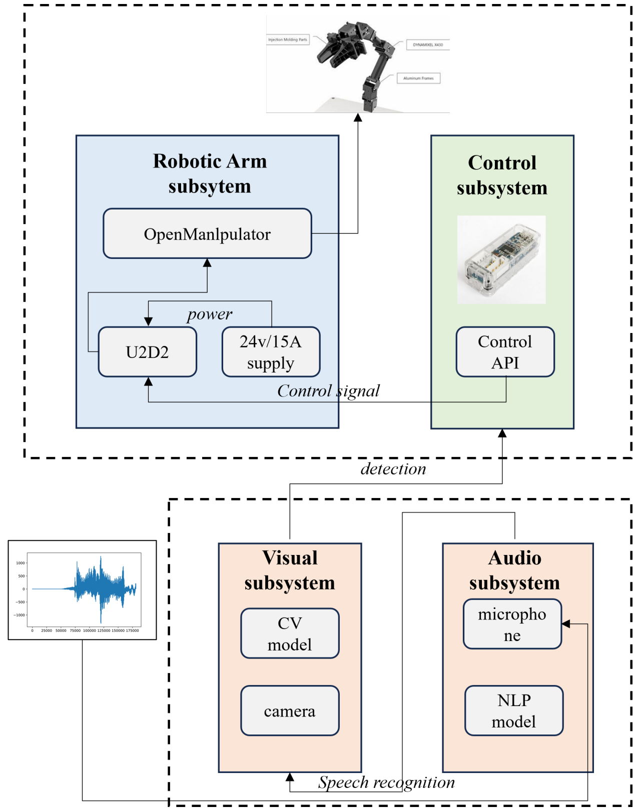


Figure 2: The block diagram of the whole system overview. It contains three subsystems.

The intelligent robotic arm system comprises three main subsystems that collaborate to provide daily living assistance to Parkinson's disease patients. First, the audio recognition subsystem captures the user's voice commands through a microphone and uses a natural language processing (NLP) model to parse these into computer signals. This process allows users to control the system through simple voice interactions, which is particularly beneficial for Parkinson's patients with limited mobility, significantly enhancing their convenience and independence.

Next, the visual recognition subsystem identifies and locates the target objects specified by users using a camera and a computer vision (CV) model. The system processes the visual information captured by the camera to accurately determine the position of objects, providing the necessary spatial information for the operation of the robotic arm. This enables the system to recognize and locate various everyday items such as utensils and books, assisting users in completing tasks that require fine motor skills.

Finally, the robotic arm control subsystem receives position information from the visual recognition subsystem and precisely executes the user's task commands using the Open-

Manipulator robotic arm and control unit. The high flexibility and precise control of the robotic arm make complex hand movements possible, significantly improving the quality of life and independence of Parkinson's patients. The close cooperation of these three subsystems not only responds to the real-time needs of users but also enhances their self-care capabilities through advanced technology, allowing them to maintain relative independence at home.

2 Design

2.1 Mechanical Control System

2.1.1 Design procedure

In the whole system, the main function of the mechanical control subsystem is to accept the information output by the visual recognition system as input according to the concepts and codes such as motion planning, operation control and kinematics, and finally control the robot arm through reasonable calls and operations to realize the command such as grasping proposed by the operator.

Considering that in the final version of this project, we will use the mechanical gripper with more complex functions designed and produced by the team independently, we will control the mechanical control subsystem for the mechanical arm and the gripper respectively. First, the United Robotics Description Format (URDF) is built under the framework of XML syntax to correctly describe the robot model. Configure MoveIt to complete the Motion Planning, 3D Perception, Kinematics, Collision Detection and other configurations, and generate the move_group core node. Next, C++ controllor packages for the robot arm and gripper were written respectively to complete the correct communication and control of the robot arm. Next, based on Python interface, the robot arm and gripper are controlled by reasonably writing the required instructions. Finally accept the coordinate and pose output from the visual control system to complete the user's instructions.

2.1.2 Design Details

URDF

The URDF is a language format used to describe robots under the XML syntax framework, such as the freedom of chassis, camera, lidar, robot arm and different joints. A robot is mainly composed of a link and a joint in a tree structure. In this project, the robotic arm consists of seven links and six joints. The link label is used to describe the appearance and physical properties of each rigid body part of the robot. At the same time, we set a series of required attributes of the part in the link label, such as shape, size, color, inertia matrix, collision parameters, etc. The joint label is used to describe the kinematic and dynamic properties of the robot joint, and connects two adjacent links in the form of

”parent-child.” Each link is connected one by one in a series way through the joint, and each joint connects the upper and lower links through the parent-child relationship. At the same time, in order to ensure the operation of collision detection, we simplified the URDF of gripper into a cuboid that can exactly enclose gripper, including a joint whose motion type is fixed, that is, a special joint that is not allowed to move, and a Gripper Link. The Gripper is connected in series with the End Link of the mechanical arm, let the End Link as the parent link. The complete parent-child tree is shown in Figure 3 below. And the final URDF visualization of the robot arm and gripper in this project is shown in Figure 4.

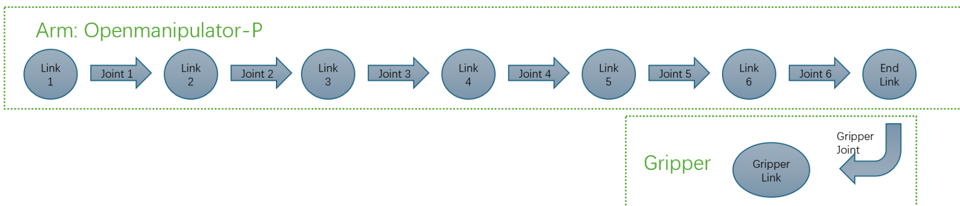


Figure 3: Parent-child Tree of the Project.

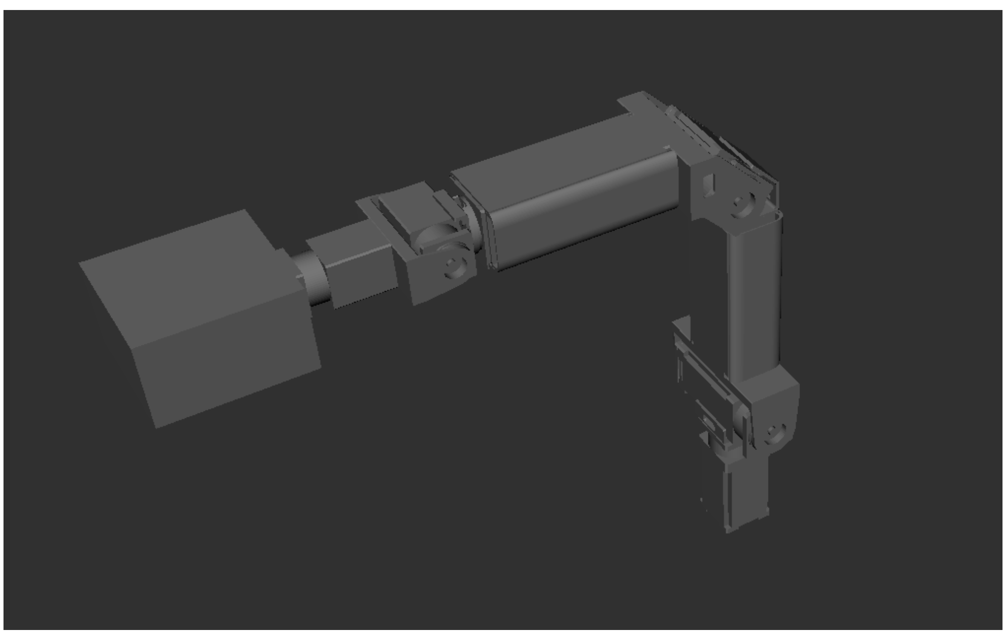


Figure 4: The URDF Visualization of the Robot Arm and Gripper.

MoveIt Configuration

MoveIt is the most widely used operating software that provides an easy-to-use robotics platform for developing advanced applications, evaluating new designs and building integrated products in industrial, commercial, R&D and other fields. MoveIt is a mobile manipulation related toolset software that integrates various SOTA libraries, including: motion planning, 3D sensing, kinematics, collision detection, Control (navigation), etc. The core nodes to realize the above functions is the “move group,” Figure 5 below is the schematic diagram of move group node.

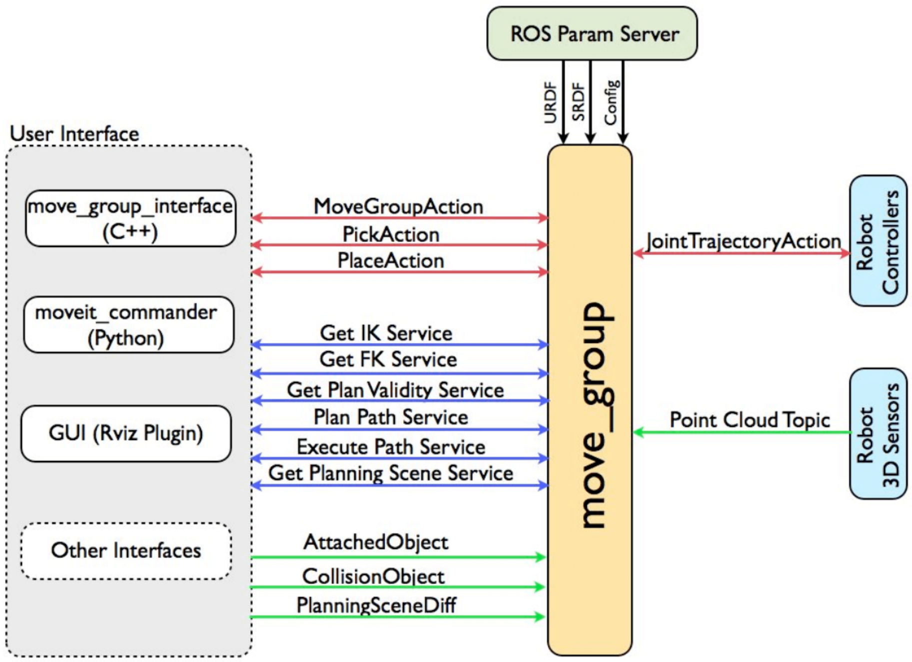


Figure 5: The schematic diagram of move group node.

In the MoveIt configuration of this project, we perform the following configuration: Optimize self-collision checking, which searches for robot joint pairs that can be safely disabled for collision detection, thereby reducing motion planning time. Define planning groups to create and edit "joint model" groups for the robot. Define the "home" pose. Setup ROS controllers, it's going to configure MoveIt to work with ROS Control to control the robot's physical hardware. In the final step, generate configuration files package needed to run our robot with MoveIt[1].

Hand-eye Calibration and Coordinate Transformation

In order to unify the camera coordinate system and the manipulator coordinate system in the visual recognition subsystem, hand-eye calibration is carried out. In this project, the Eye-to-hand method was used for hand-eye calibration, that is, the orientation of the camera coordinate system relative to the robot base coordinate system was calibrated, rather than following the robot. In this project, we used ArUco tag for hand-eye calibration.[2] ArUco markup is a two-dimensional binary-coded markup pattern designed to be quickly located by computer vision systems. Each tag consists of a wide black border and an internal binary matrix to determine its identifier (ID). Figure 6 shows how we perform hand-eye calibration.

In this project we used a tag with an ID of 26 and set the size to 40 mm*40 mm. Our final

Figure 6: hand-eye calibration.

calibration results are as follows:

$$
\begin{aligned}
\text{translation:}\quad & \\
& x: 0.634748018657 \\
& y: -0.195510960127 \\
& z: 0.530579004586 \\
\text{rotation:}\quad & \\
& x: -0.804823734182 \\
& y: -0.501931055961 \\
& z: 0.096718789851 \\
& w: 0.301611418301
\end{aligned}
$$

By converting the quaternions in rotation to a rotation matrix we get the transformation matrix we need. The transformation matrix is as follows:

$$
T = \begin{bmatrix} 0.74958907 & -0.45845943 & 0.63474802 & 0.47742138 \\ -0.31419153 & 0.38839573 & -0.19551096 & 0.86627504 \\ -0.58258038 & -0.79935206 & 0.53057900 & 0.14709312 \\ 0 & 0 & 0 & 1 \end{bmatrix}
$$

Figure 7 below shows the tf tree after hand-eye calibration. From Figure 7, we can see that the hand-eye calibration successfully unified the two coordinate systems.

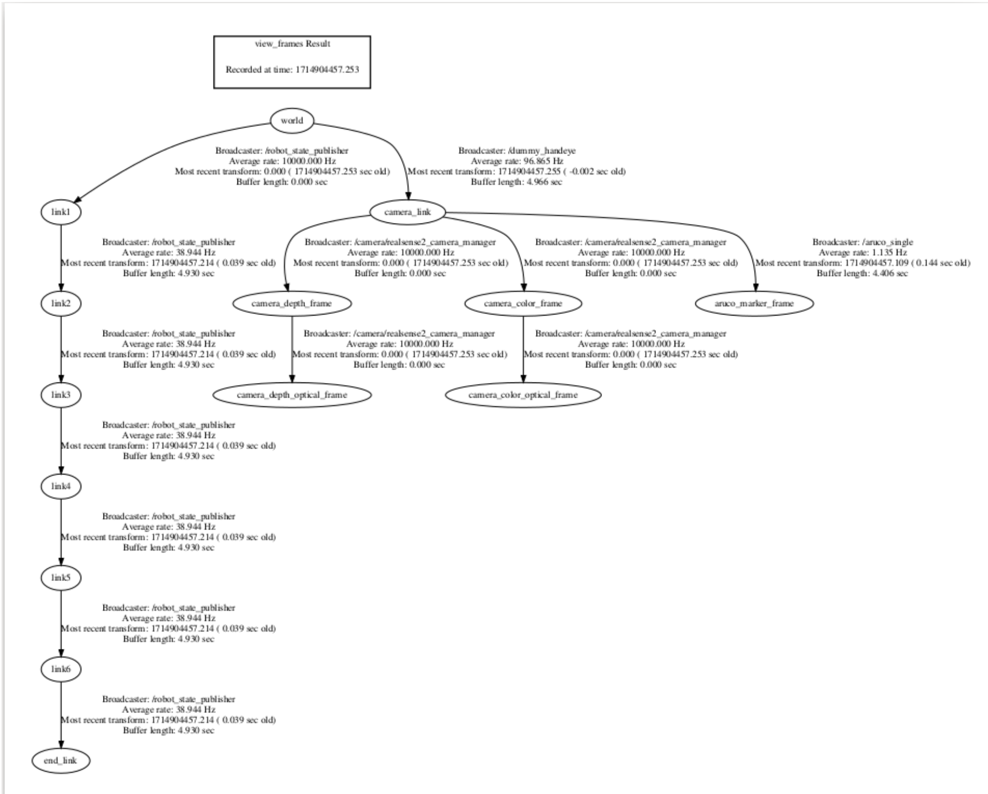


Figure 7: The tf tree after hand-eye calibration.

Python Interface and Robot Control To work with the speech recognition and visual recognition subsystems, we chose to set up the Python (moveit commander) interface in our project. We define two classes and several functions to control the movement of the robot arm and the opening and closing of the gripper. Figure **??** shows the control code of the project.[3]

2.2 Speech recognition subsystem

2.2.1 Design procedure

Firstly, through microphone input, the subsystem achieves real-time conversion, providing users with an efficient and convenient means of interaction. Following the voice input, leveraging the OpenAI-Whisper module, the system converts speech into natural language, achieving precise conversion from sound to text. After the conversion of speech into natural language, the system further processes the text data using the Spacy module to extract named entities. Through the efficient operation of the Spacy model, the system accurately identifies and categorizes various entities, including personal names, locations, and organizations.Figure 9 shows the procedures of speech recognition subsystem.[4]

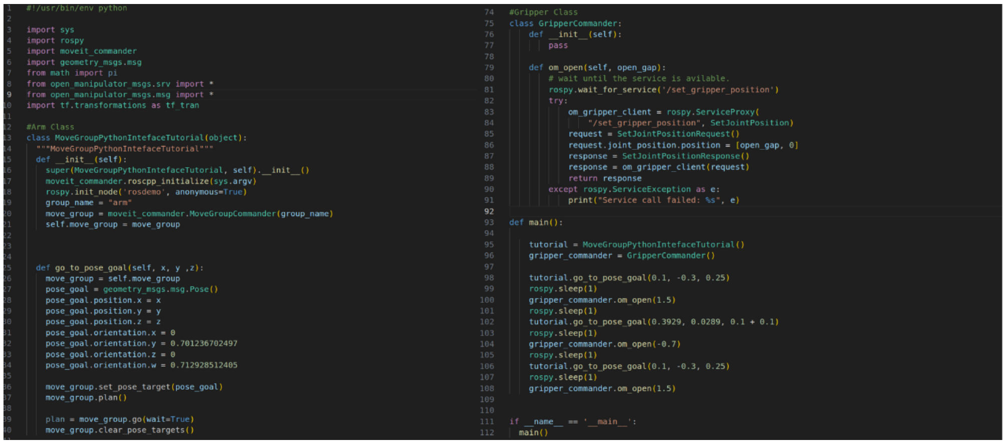


Figure 8: The control code.

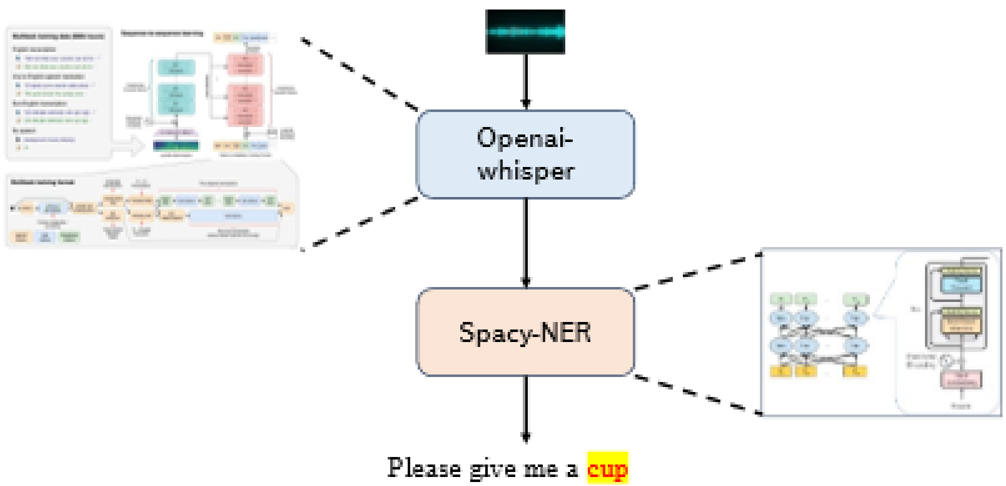


Figure 9: Speech recognition subsystem.

2.2.2 Design Details

Openai-whisper

Whisper is a universal speech recognition model that leverages extensive multilingual and multitask supervised data for training. It achieves near-human-level robustness and accuracy in English speech recognition. Whisper is capable of performing various tasks including multilingual speech recognition, speech translation, and language identification. The architecture of Whisper is a simple end-to-end approach, (show in Figure 10) employing an encoder-decoder Transformer model to convert input audio into corresponding text sequences. The encoder initially processes input representations with a stem consisting of two convolutional layers (with a filter width of 3) using the GELU activation function. The second convolutional layer has a stride of 2. Sinusoidal positional embedding is then added to the stem's output, followed by the application of encoder Transformer blocks. Transformers utilize pre-activated residual blocks, and the output of the encoder is normalized using normalization layers. In the decoder, learned positional embedding and tied input-output token representations are utilized. The encoder and decoder possess the same width and number of Transformer blocks.

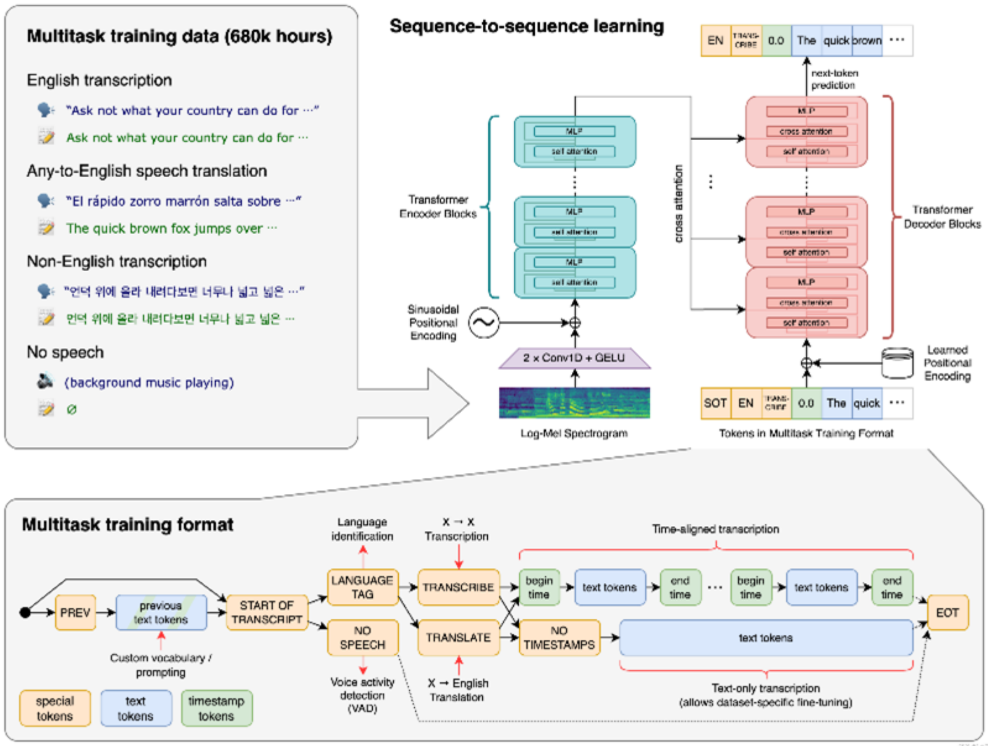


Figure 10: Openai-whisper model.

Spacy-NER
Spacy first tokenizes the text to generate a Doc object. Subsequently, the Doc object undergoes processing through several stages. A trained processing pipeline typically includes components such as part-of-speech taggers, dependency parsers, and entity recognizers. Named entity recognition (NER) is achieved by loading pre-trained models.

2.3 Visual Subsystem

2.3.1 Design procedure

Figure 11 shows the procedures of visual subsystem . In details, The visual subsystem initiates by leveraging the RealSense D435i camera to capture comprehensive imagery, encompassing both RGB and depth data of the environment. These raw images undergo a series of preprocessing steps to enhance quality and extract pertinent features. Following this, they are seamlessly integrated with the entity recognition outputs from the speech system, establishing a holistic understanding of the scene. Concurrently, YOLOv5 meticulously identifies and localizes objects within the images. Upon successful detection, the identified objects are precisely cropped from the surrounding context, streamlining subsequent processing. Finally, employing the grasp regional convolutional neural network (GRCNN), the system meticulously analyzes the cropped objects, determining optimal coordinates for effective grasping actions.[5]

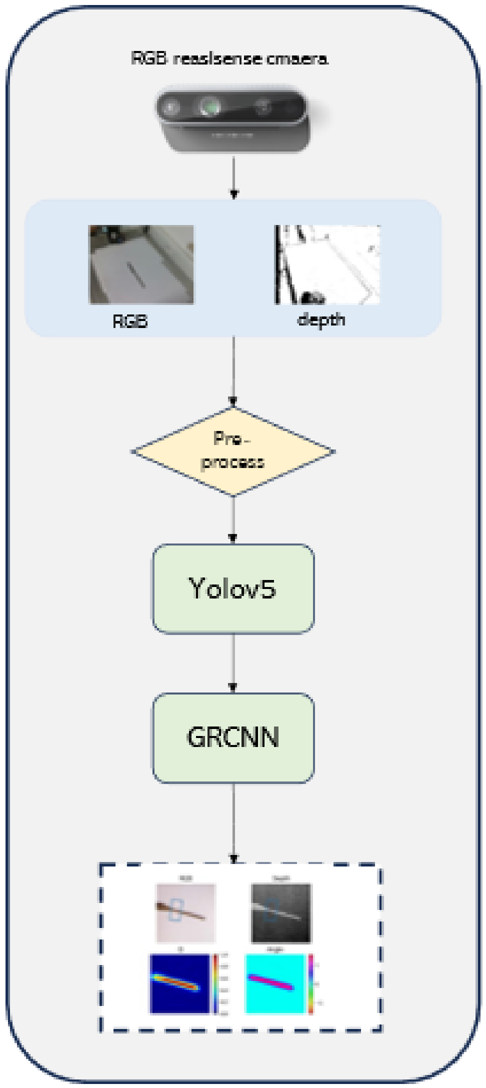


Figure 11: Visual subsystem.

2.3.2 Design details

Camera
The RealSense D435 camera provides a global shutter sensor and a larger lens to achieve better low-light performance compared to the cheaper D415 camera. Additionally, the D435 features a more powerful RealSense module, the D430. This RealSense camera series can capture distances up to 10 meters and can be used outdoors in sunlight. They

support outputting depth images at a resolution of 1280x720, and for regular video transmission, they can achieve up to 90fps.[6]
cam_intrinsics : 607.607,0,315.862,0,607.806,252.004,0,0,1
depth scale: 0.0010000000474974513

Yolov5

The YOLOv5 architecture comprises Input, Backbone, Neck, and Prediction modules (shown in Figure 12). Mosaic data augmentation, adaptive anchor calculation, and adaptive image scaling are included in Input. Mosaic augmentation, similar to YOLOv4, scales, crops, and concatenates images randomly, enhancing detection, especially for small objects. YOLOv5 adjusts initial anchor boxes during training, updating parameters through back propagation based on predicted versus ground truth boxes. Unlike resizing all images to a standard size, YOLOv5's adaptive letter boxing minimizes redundant information and inference speed impact caused by varying aspect ratios, adding minimal black border adaptive to the original image.

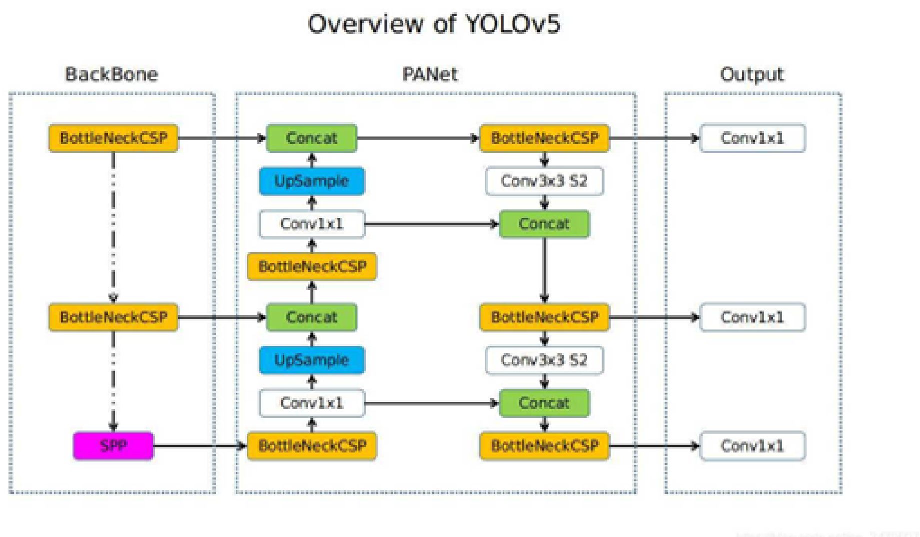


Figure 12: The structure of YoloV5.

GRCNN

The system receives a single reference image as one of the inputs to the visual CNN. The currently captured image from the camera acts as both the second input to the network and the input to the grasping CNN. Both networks operate continuously: the grasping CNN provides real-time rectangle predictions for monitoring, while the visual (VS) network regulates the robot's pose in real time. The VS CNN predicts a velocity signal, which is then multiplied by a proportional gain and applied to adjust the camera. The robot's internal controller adjusts joint velocities based on the predicted camera velocity. At each iteration, the current image updates according to the robot's position, and this loop continues until the control signal converges. Upon meeting a stopping condition, the grasping network's prediction is mapped to the world coordinate system. The robot reaches the predicted point using inverse kinematics and subsequently closes the gripper.[7]

2.4 Robotics Arm

For the robotics arm, we will use OpenMANIPULATORP that has six joints. The OpenMANIPULATORP is versatile and can be used for various tasks, including pick-and-place operations, manipulation of objects, and simple assembly tasks. Its multiple degrees of freedom (DOF) and configurable end-effectors make it suitable for a wide range of applications. We will modify it so that it can achieve our goal of helping people. The specific parameters for the robotics arm is listed in Tables and Figures in Appendix A.

2.5 End-effector

2.5.1 Design procedure

In our project, the adaptive end-effector is a sophisticated component designed to interact seamlessly with varied objects during operation, enhancing the robotic arm's versatility and effectiveness.

In our final design, we will use the five-bar linkage to deisgn our end-effector. The five-bar linkage can generate more complex paths or trajectories for the end effector. This makes it suitable for tasks that involve following a specific path or contour, such as painting, welding, or assembly operations that require following a complex outline. And the configuration of the five-bar linkage can be designed to achieve a higher mechanical advantage, allowing the system to exert greater forces with less input. This is particularly useful in applications where heavy lifting or exertion of large forces is required. Due the 2 DOF of five-bar linkage, we can design a end-effector that can capture different shapes and sizes of targets.

2.5.2 Deisgn detail

Our requirement for the end-effector is to grasp a variety of targets with different shape and weight. To achieve this goal, we designed a five-bar linkage end-effector shown in Figure 13.

Since the five-bar linkage system inherently provides two degrees of freedom (DOF), the motors in our design are strategically placed at points A and B. This setup ensures that torque is applied only at these points, and the primary control point for the system is located at point 3. It is essential to understand that for successful grasping functionality, the orientation of both the outer and inner platforms is crucial, rather than their specific positions.

Within the mechanics of the five-bar linkage, the motor at point B is responsible solely

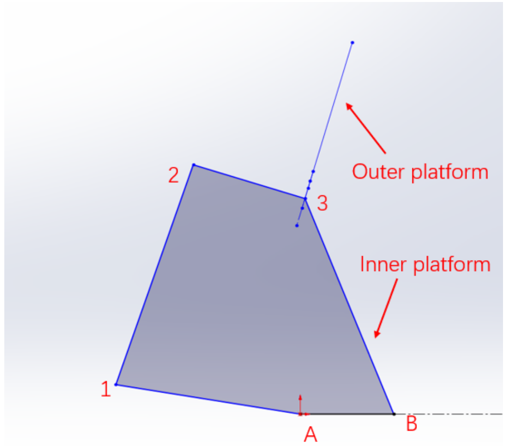
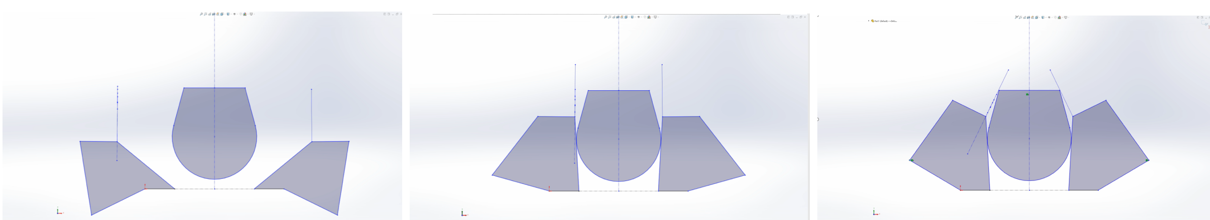


Figure 13: Five-bar linkage

for controlling the orientation of the outer platform. In contrast, the motor at point A is crucial as it influences not only the orientation but also the positioning of both the outer and inner platforms. Based on this understanding, we have developed a specific grasping strategy.

The initial phase of our strategy involves simultaneously activating motors A and B. This coordinated action maintains the orientation of the outer platform while dynamically adjusting the positions of both the outer and inner platforms to approach the target. Once the inner platform makes contact with the target, the strategy shifts; we then operate only motor A. This adjustment specifically alters the orientation of the outer platform to ensure it also makes contact with the target, effectively achieving the grasp. This process and the mechanical interactions involved are illustrated in Figure 14.



Start position State 1 State 2

Figure 14: Grasping strategy

Due to the angle relationship in the five-bar linkage and Law of cosines and sines, we can get the following equation.

$$\eta = \pi + \beta - \theta - \gamma$$
$$k = \sqrt{l_4^2 + l_5^2 - 2l_4l_5\cos\eta}$$
$$\delta = \sin^{-1}\frac{l_4\sin\eta}{k}$$

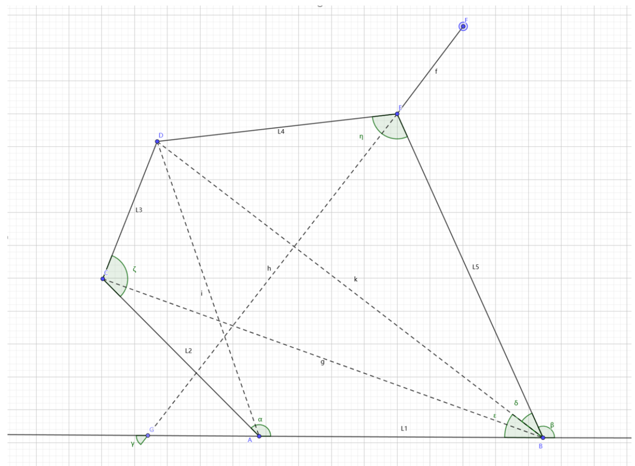


Figure 15: Diagram of five-bar linkage

$$\varepsilon = \pi - \beta - \delta$$

$$i = \sqrt{l_1^2 + k^2 - 2l_1 k \cos \varepsilon}$$

$$\zeta = \cos^{-1} \frac{l_2^2 l_3^2 - i^2}{2l_2 l_3}$$

Then we can get

$$l_1^2 + l_2^2 - 2l_1 l_2 \cos \alpha = l_3^2 + k^2 - 2l_3 k \cos(2\pi - \varepsilon - \alpha - \zeta)$$

So α have to satisfy the above equation to remain the orientation of outer platform unchanged. Every time when β is changed, α need to be updated using this equation.
For now our determined parameters are:

$$l_1 = 19.5\ mm$$

$$l_2 = 31.74\ mm$$

$$l_3 = 86.5\ mm$$

$$l_4 = 19.5\ cm$$

$$l5 = 100\ mm$$

$$r = 71\ mm$$

The CAD details for the principal components are delineated in Appendix A. The majority of these components will be fabricated using 3D printing technology. This method facilitates rapid prototyping, thereby accelerating the design process substantially. However, for component AppendixA Figure22, laser cutting will be employed due to the critical nature of the holes on this part, which are integral to the orientation of the outer platform. Laser cutting is selected over 3D printing in this instance due to its superior manufacturing accuracy, ensuring greater precision in the production of essential features.

3 Cost and Schedule

3.1 Cost Analysis

3.1.1 Labor

In this project, the labor cost is calculated by the total number of hours to complete the task and the hourly wage per team member. With reference to ECE industry average hourly wage, we see that the labor cost per team member per hour is determined to be 100 RMB. At the same time, considering that we have about 8 weeks to complete the project and we plan to We plan to spend 20 hours a week on the project, we can calculate:

$$20hr/week \times 8weeks \times 100RMB/hr = 16000RMB$$

Therefore, the labor cost per team member is 16000 RMB.

3.1.2 Parts

The parts list and estimated costs are shown in Table 1.

Table 1: Parts List and Estimated Costs

Description	Manufacturer	Part Number	Quantity	Cost
Robotic Arm	Robotis	OpenMANI PULATOR-P	1	97000 RMB
Serial Module: Steering Engine Controller	Robotis	U2D2	1	500 RMB
3D Camera	Intel	RealSense D435i	1	2100 RMB
Notebook	Xitong	954	4	40RMB
Student power supply			1	150RMB

3.1.3 Total Cost

According to the above statistics, we know that the labor cost of our project is 16000RMB and the cost of Parts is 99790RMB. Therefore, the total cost of our project is 115,790RMB.

3.2 Schedule

3.2.1 Timeline

March:

1. Complete the configuration of the robotic arm environment and download pre-training models. **Done**

2. Accomplish basic manipulator motion control. **Done**

April:

1. Procure all sensors, devices, and control boards required for the project. Assemble and connect all components. **Done**

2. Investigate the YOLOv5 Detection model and integrate it into the system. **Done**

3. Finalize the audio model. **Done**

4. Complete the preliminary assembly of each module. **Done**

May:

1. Finalize the overall design of the robot arm function. **Failed on designing our own end effector**

2. Conduct tests for assigned tasks. **Done**

3. Prepare project-related reports. **On going**

3.2.2 Per-person

Our personal weekly schedule is shown as Table 2.

4 Requirements and Verification

4.1 Mechanical Control System

After many rounds of testing and demos, we can prove that the mechanical control subsystem performs well both in isolation and in conjunction with other systems. The mechanical control subsystem can accurately locate the position after receiving the coordinate and pose input from the visual recognition subsystem. At the same time, the center position of the grab frame output by the visual recognition subsystem completely coincides. In the actual test, the target position offset caused by the mechanical control subsystem is less than ±0.05 mm, which can be ignored. In summary, we conclude that the mechanical control subsystem has the characteristics of high precision, high sensitivity, high anti-interference and high stability, and performs well in various application scenarios. Figure 16 is a picture of the whole process of grasping the mouse test.

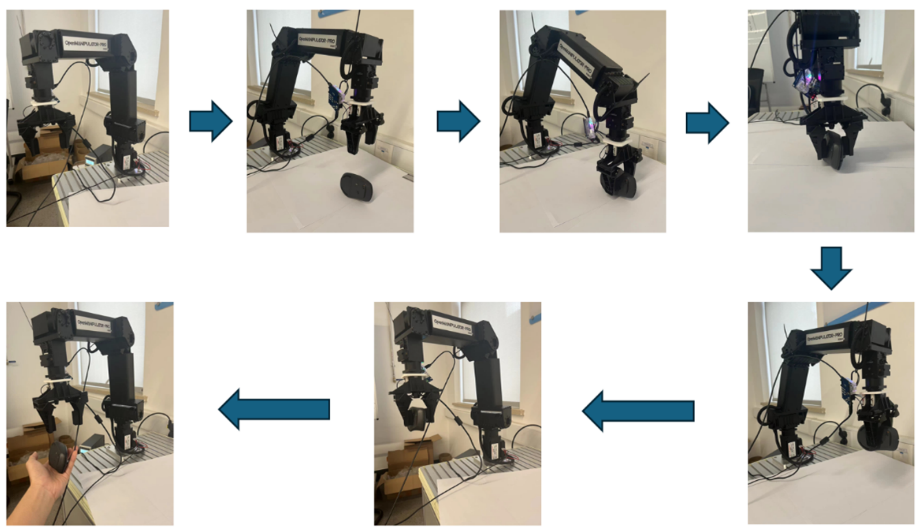


Figure 16: The Whole Process of Grasping a Mouse.

4.2 Speech recognition subsystem Verification

Through real-world testing scenarios, it has been observed that this subsystem effectively achieves speech entity recognition. Figure 17 shows the result of speech recognition subsystem.

4.3 Visual Subsystem

We verified the detection effect of YOLOv5 and GRCNN in the actual robotic arm scenario. The results show that YOLOv5 (shown in Figure 6) performs well in target recogni-

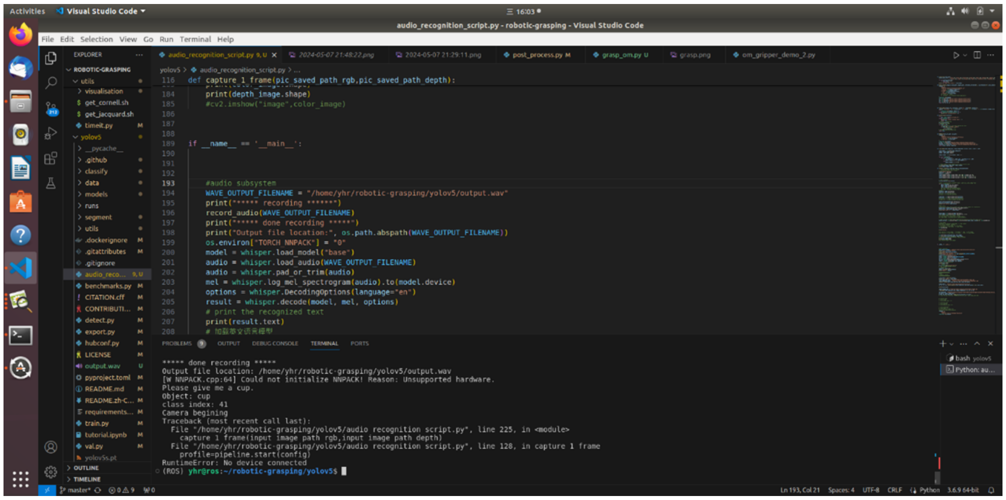


Figure 17: Verification of Speech recognition subsystem.

tion, while GRCNN can accurately detect and locate the grasp position (shown in Figure 7), providing reliable support for the perfect combination of target detection and grasp recognition.

Figure 18: Verification of Yolov5.

4.4 End Effector

4.4.1 Verification

Since the target for the robotic arm is nearly symmetric, we are able to simplify the design by reducing the number of motors used. We have constrained the angle between L1 and L2 to remain constant, ensuring that the outer platform maintains perpendicularity to line AB. Given that the lengths of L1 and L2 are equal, this configuration effectively transforms our initial five-bar linkage system into a simpler four-bar linkage system, comprised of line i, and links L3, L4, and L5, which form a parallelogram.

As a result of this design choice, only a single motor located at point B is required to drive the end-effector. To enhance the system's adaptability, we permit the angle between L2 and L3 to vary in response to L5 encountering an obstruction. This flexibility allows the linkage to adjust dynamically, ensuring that the arm can adapt its movement when L5 is

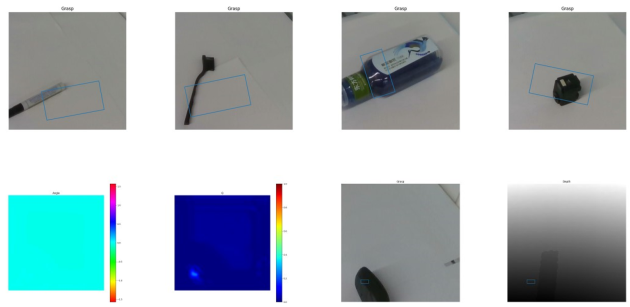
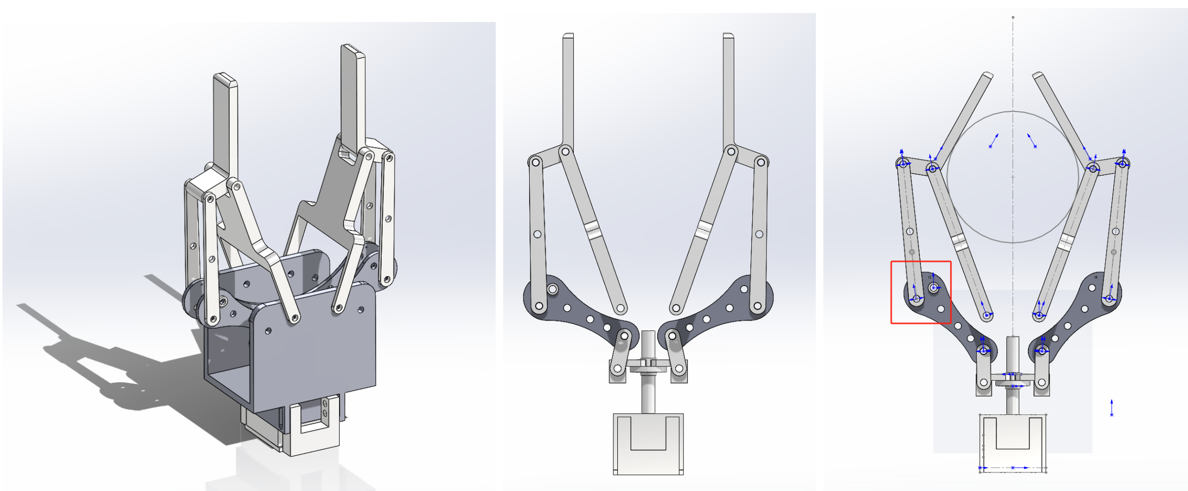


Figure 19: Verification of GRCNN.

restricted, thereby maintaining operational efficiency and effectiveness.



CAD for the end-effector CAD for inner structure Bend of outer platform

Figure 20: CAD of the end-effector

Figure 20 provides a detailed depiction of our design's structural configuration. In order to achieve the specified limit effect, a pin has been strategically placed between L2 and L3 to constrain their minimum angle of interaction, ensuring controlled movement within predefined parameters. Additionally, a rubber band is affixed to both L2 and L3 to further influence their dynamic relationship. This setup allows L2 and L3 to remain relatively stationary in relation to each other while permitting L5 to move freely.

The implementation of the rubber band not only reinforces the connection between L2 and L3 but also introduces the capability to increase the angle between these components. This flexibility is crucial for the inward bending of the outer platform when L5 ceases its movement, thereby achieving the desired mechanical functionality. The red box in Figure 20 distinctly marks the area where the angle between L2 and L3 increases, providing a visual reference that aids in understanding the mechanical interactions and limitations set forth in our design.

5 Conclusion

5.1 Accomplishment and Uncertainties

In this project, we achieved significant accomplishments and successfully implemented several key functionalities that showcase advanced control and intelligent interaction capabilities of the robotic arm. Below are the specific achievements and final functionalities:

1. **Robotic Arm Control and Hand-Eye Calibration:** We successfully implemented precise control of the robotic arm and completed the coordinate transformation from the camera's coordinate system to the robot's base coordinate system through hand-eye calibration. This progress provided accurate spatial localization for subsequent operations, ensuring the accuracy and efficiency of the robotic arm's maneuvers.

2. **Voice to Text Conversion:** By integrating the OpenAI Whisper network, we achieved the functionality of converting voice input into text. This allows users to interact directly with the system via voice, enhancing the system's usability and accessibility, especially important for users with limited mobility.

3. **Text Analysis and Entity Extraction:** Using the Spacy network, we performed named entity recognition and relationship extraction on the converted text, accurately identifying the objects that users wished to grasp. This step is crucial in the intelligent grasping process, ensuring that the robotic arm can understand the specific needs of the users.

4. **Object Detection and Pose Estimation:** We configured and successfully used the Realsense depth camera, in conjunction with the YOLOv5 network, to accurately detect the position of target objects on the desktop. Additionally, by inputting both the depth image and the object's RGB image into the GRCNN network, we successfully obtained the optimal grasping pose for the robotic arm's end effector and were able to perform accurate grasping tasks.

5. **Uncompleted Work:** Although we have designed the end effector, it has not yet been installed on the robotic arm due to time constraints, nor have we conducted practical grasping tests. This remains a key part of the project that is still to be completed.

Overall, this project has successfully implemented most of the functionalities envisioned at the beginning of the project, particularly in assisting patients with object grasping through voice input. We look forward to completing the installation and testing of the end effector in the future to enhance the system's capabilities and further improve the quality of life for Parkinson's patients.

5.2 Future Work

As we continue to advance our project, there are several key areas we plan to focus on to enhance the capabilities and overall effectiveness of the robotic arm system:

1. **Installation and Testing of the End Effector:** Our immediate next step is to install the newly designed end effector onto the robotic arm. Following installation, comprehensive testing will be conducted to ensure the end effector operates effectively and safely, particularly in handling and manipulating objects. This will also include fine-tuning the interface between the end effector and the robotic control system to optimize performance.

2. **Improvement of Voice Interaction Capabilities:** To make the system more intuitive and user-friendly, we will enhance the voice recognition and processing capabilities. This includes expanding the vocabulary and improving the accuracy of the speech to text conversion, particularly under varying environmental conditions. Such enhancements will improve the system's responsiveness and accuracy in understanding and executing user commands.

3. **Enhanced Object Recognition and Manipulation:** We plan to upgrade the computer vision algorithms to enhance the system's ability to recognize and manipulate a wider range of objects with different shapes, sizes, and textures. This will involve training the system with more diverse datasets and possibly integrating more advanced neural networks that specialize in object detection and manipulation.

These efforts are geared towards ensuring that the robotic arm system not only meets but exceeds the expectations and needs of its users, particularly in aiding those with Parkinson's disease in their daily activities.

6 Discussion of Ethics and Safety

6.1 Ethics

Developing a project involving the use of robotic arms to provide assistance to Parkinson's patients raises ethical and safety concerns. Ethical issues to consider include privacy, the right to know, and potential harm to customers.

The IEEE Code [8] of Ethics emphasizes the importance of the safety and welfare of the public. Therefore, the project should prioritize the safety and comfort of the human clients. The ACM Code of Ethics [9] emphasizes the importance of avoiding harm, and the project should strive to avoid any potential harm to the clients.To avoid ethical violations, project teams should obtain informed consent from clients before providing services to them.

In the development process of robots, the project team should pay close attention to their potential interference with patients' normal lives, especially considering the functional limitations that may arise in certain situations. This means not only ensuring that the design and functionality of the robots can integrate seamlessly into patients' daily lives but also carefully considering the potential constraints that may arise in specific circumstances. Such attention not only contributes to ensuring the effectiveness and practicality of the robots but also minimizes the disruption and distress they may cause to patients' lives.

Furthermore, we should also pay particular attention to the issues that may arise from robots handling sensitive information and personal privacy. This includes but is not limited to how robots collect, store, and use personal data, as well as ensuring that this information is not accessed or abused without authorization. In the design and development process, strict adherence to best practices in privacy protection is essential, while also considering various potential threats and risks to ensure that users' data security and privacy are adequately protected.

6.2 Safety

According to the Occupational Safety and Health Administration (OSHA) safety standards, in section 1910.212, it is stated that all machinery should meet general requirements. Specifically, it is mandated that robotic systems operate in compliance with safety regulations. In this project, the primary concern revolves around the pressure exerted by the mechanical arm and the pressure applied by the massage head on the human body. This issue persists across all stages of our project, from design and testing to the final demonstration. To elaborate, we have the following safety concerns:

1.The mechanical arm must possess sufficient precision and sensitivity to ensure that no unintended harm is inflicted upon patients during task execution. Additionally, the design of the mechanical arm must take into account the physical condition and mobility of the patients to ensure that its operation does not exert additional pressure or discomfort.

2.The safety of the mechanical arm must also consider its contact with the patients' bodies, necessitating the implementation of appropriate protective measures to prevent injury from accidental contact. Furthermore, the control system of the mechanical arm should feature reliable safety functionalities, such as emergency stop buttons and limit protections, to address potential unforeseen circumstances and ensure patient safety under all conditions.

For addressing safety concerns regarding the mechanical arm assisting Parkinson's patients, the following solutions can be implemented:

1.Precision and Sensitivity: Ensure that the mechanical arm possesses high precision and sensitivity, which can be achieved by using high-quality sensors and precise control systems. Additionally, thorough testing and calibration should be conducted to ensure that the mechanical arm can accurately recognize and respond to the patients' movements during task execution.

2.Consideration of Patients' Physical Condition and Mobility: When designing the mechanical arm, it's essential to fully consider the patients' physical condition and mobility to ensure that its operation does not exert additional pressure or discomfort on them. This can be achieved through customized design or adjustable operating parameters.

3.Implementation of Safety Measures: The design of the mechanical arm should account for its contact with the patients' bodies and incorporate appropriate protective measures, such as soft protective covers or safety triggers, to prevent injuries from accidental contact.

4.Integration of Safety Features into the Control System: The control system of the mechanical arm should be equipped with reliable safety features, including emergency stop buttons, limit protections, and collision detection, to address potential unforeseen circumstances. Additionally, training and guidance should be provided to ensure that operators know how to respond correctly to emergency situations.

Table 2: Weekly Schedule

Week Number	Haoran Yang	Yipu Liao	Yiming Li	Chenghan Li
3.25-3.31	Inital set up of robotic arm	Inital set up of robotic arm	Run the yolov5 model	Run the yolov5 model
4.1-4.7	Install the manipulator motion algorithm library	Install the manipulator motion algorithm library	Install the Realsense camera driver	yolov5 was used to detect the images collected by the camera
4.7-4.14	Fix previous problems	Fix previous problems	Fix previous problems	Fix previous problems
4.14-4.21	Design and print the gripper	Design and print the gripper	Test the vision system	Implement audio system
4.21-4.28	Complete the control system and preliminary demo	Complete the control system and preliminary demo	Integrated voice system and visual system	Integrated voice system and visual system
4.28-5.5	Test mechanical reliability test	Do real safety and functionality test	Test visual-audio reliability test	Test visual-audio reliability test
5.5-5.12	Finish the whole project and prepare the demo	Finish the whole project and prepare the demo	Finish the whole project and prepare the demo	Finish the whole project and prepare the demo
5.12-5.19	Improve thr functionality and write final report	Improve the functionality and write final report	Improve the functionality and write final report	Improve the functionality and write final report

References

[1] J. Redmon and A. Angelova, "Real-time grasp detection using convolutional neural networks," in *2015 IEEE international conference on robotics and automation (ICRA)*, IEEE, 2015, pp. 1316–1322.

[2] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 779–788.

[3] A. Radford, J. W. Kim, T. Xu, G. Brockman, C. McLeavey, and I. Sutskever, "Robust speech recognition via large-scale weak supervision," in *International Conference on Machine Learning*, PMLR, 2023, pp. 28 492–28 518.

[4] X. Zhu, S. Lyu, X. Wang, and Q. Zhao, "Tph-yolov5: Improved yolov5 based on transformer prediction head for object detection on drone-captured scenarios," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 2778–2788.

[5] A. Depierre, E. Dellandréa, and L. Chen, "Jacquard: A large scale dataset for robotic grasp detection," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE, 2018, pp. 3511–3516.

[6] Intel. ""Intel RealSense Depth Camera D435i"." (), [Online]. Available: https://www.intelrealsense.com/depth-camera-d435i/ (visited on 03/10/2024).

[7] Z. Ge, S. Liu, F. Wang, Z. Li, and J. Sun, "Yolox: Exceeding yolo series in 2021," *arXiv preprint arXiv:2107.08430*, 2021.

[8] IEEE. ""IEEE Code of Ethics"." (2016), [Online]. Available: https://www.ieee.org/about/corporate/governance/p7-8.html (visited on 02/08/2020).

[9] ACM. ""ACM Code of Ethics"." (2018), [Online]. Available: https://www.acm.org/code-of-ethics (visited on 03/24/2024).

Appendix A Example Appendix

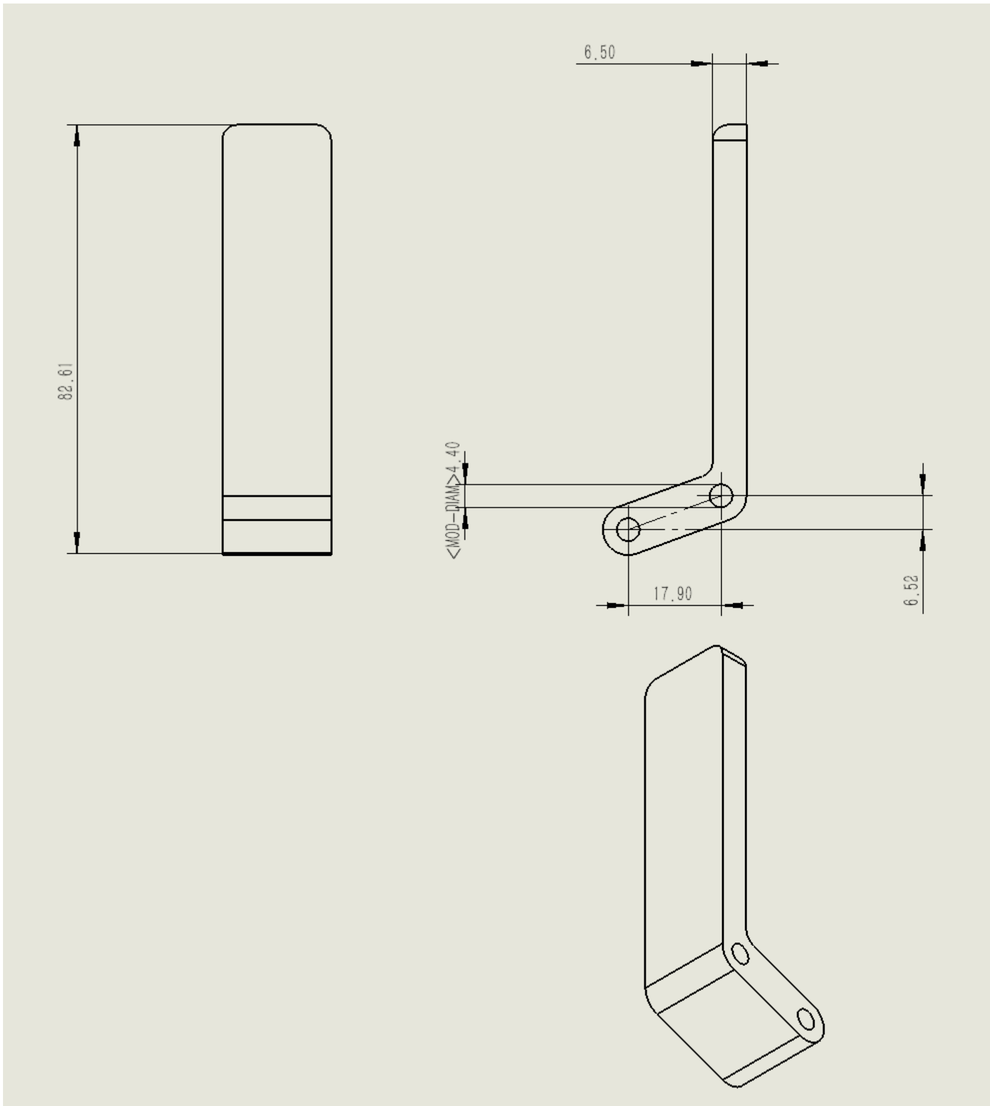


Figure 21: outer platform and L4

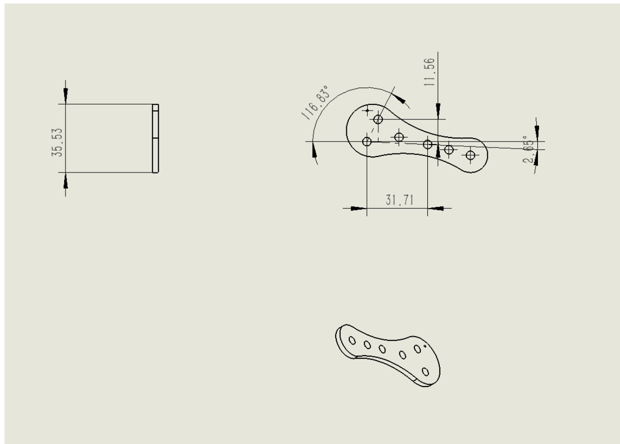


Figure 22: L1 and L2

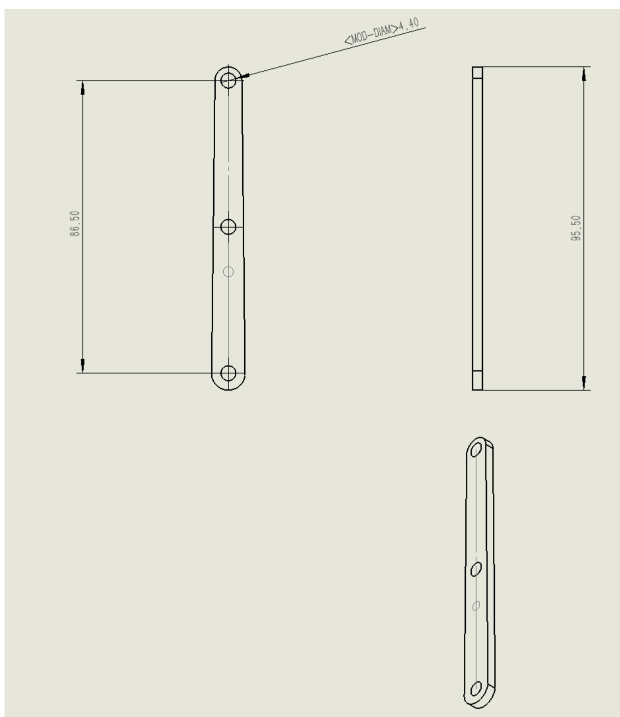


Figure 23: L3

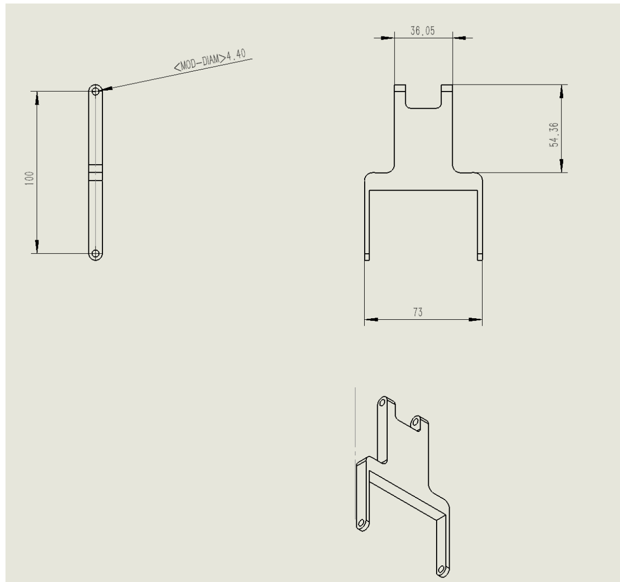


Figure 24: L5

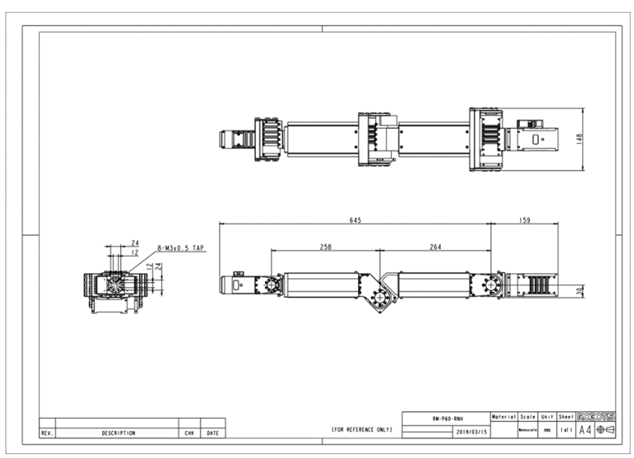


Figure 25: Parameter for robotics arm

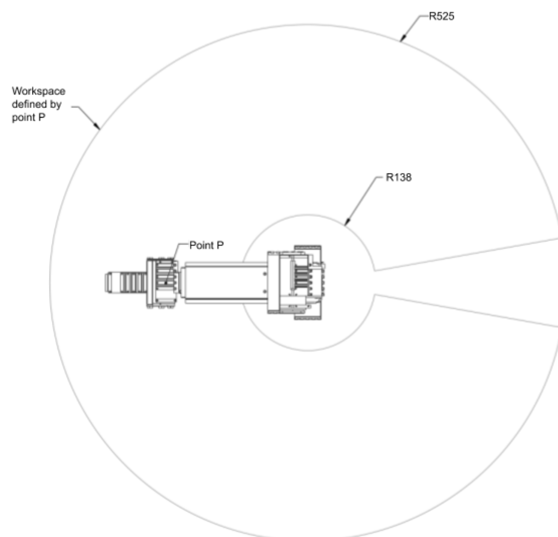


Figure 26: Work space of robotics arm

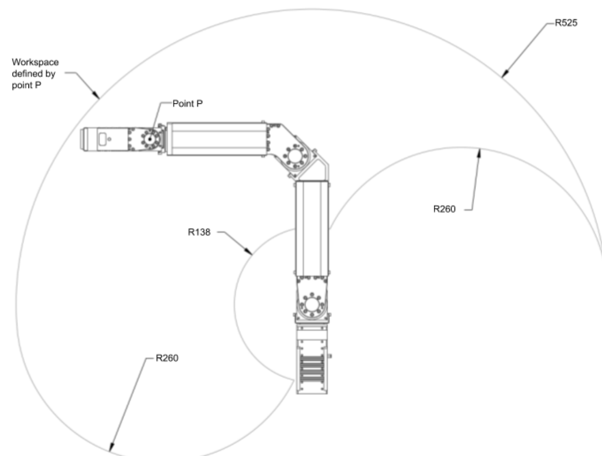


Figure 27: Work space of robotics arm

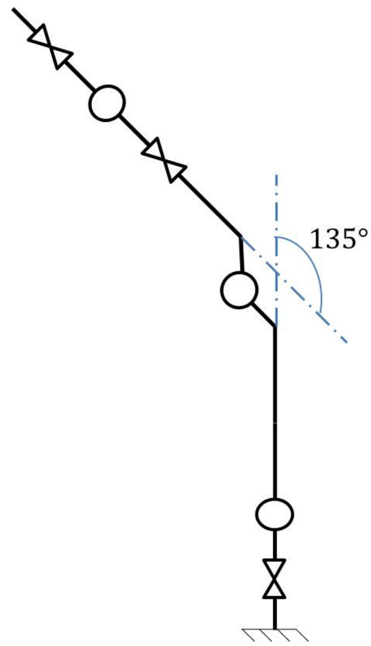


Figure 28: Home Position

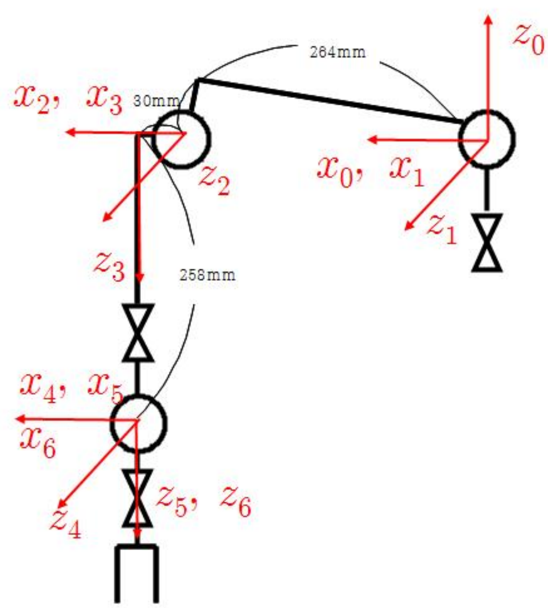


Figure 29: D-H configuration

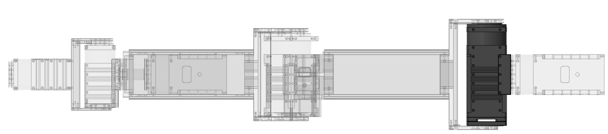


Figure 30: Link1

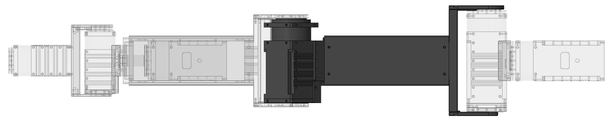


Figure 31: Link2

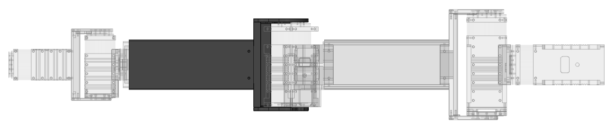


Figure 32: Link3

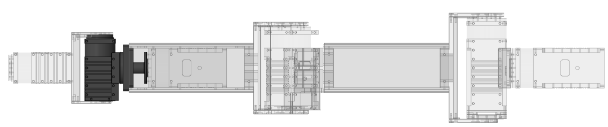


Figure 33: Link4

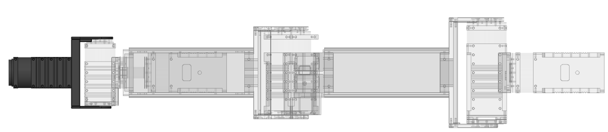


Figure 34: Link5

Table 3: Parameter for robotics arm

Item	openMANIPULATOR-P (RM-P60-RNH)
DOF Payload Reach Repeatability Weight Operating voltage	6 3 kg 645 mm $\pm$0.05 mm 5.76 kg 24 V
Resolution	Joint 1: -π(rad) $\sim$ π(rad),-501,923 $\sim$ 501,923 (pulse) Joint 2: -π(rad) $\sim$ π(rad),-501,923 $\sim$ 501,923 (pulse) Joint 3: -π(rad) $\sim$ π(rad),-501,923 $\sim$ 501,923 (pulse) Joint 4: -π(rad) $\sim$ π(rad),-501,923 $\sim$ 501,923 (pulse) Joint 5: -π(rad) $\sim$ π(rad),-303,750 $\sim$ 303,750 (pulse) Joint 6: -π(rad) $\sim$ π(rad),-303,750 $\sim$ 303,750 (pulse)
DYNAMIXEL Model Name	Joint 1, 2 : PH54-200-S500-R (200W) Joint 3, 4 : PH54-100-S500-R(100W) Joint 5, 6 : PH42-020-S300-R (20W)
Operating Range	Joint 1 : -π(rad) $\sim$ π(rad) Joint 2 : -$\frac{\pi}{2}$(rad) $\sim$ $\frac{\pi}{2}$(rad) Joint 3 : -$\frac{\pi}{2}$(rad) $\sim$ $\frac{3\pi}{4}$(rad) Joint 4 : -π(rad) $\sim$ π(rad) Joint 5 : -$\frac{\pi}{2}$(rad) $\sim$ $\frac{\pi}{2}$(rad) Joint 6 : -π(rad) $\sim$ π(rad)
Default ID	Joint 1 (ID:1), Joint 2 (ID:2), Joint 3 (ID:3), Joint 4 (ID:4), Joint 5 (ID:5), Joint 6 (ID:6)
Motor type Position sensor type	Brushless DC Servo(H54P Series), Coreless DC Motor(H42P Series) Absolute Encoder(for Homing), Incremental Encoder(for Control)
Communication Baudrate	1000000 bps

Table 4: D-H parameter

Link	Link Length(mm)	Link Twist(rad)	Joint Offset(mm)	Joint Angle(rad)	DXL Angle(rad)
1	0	$\frac{\pi}{2}$	0	0	0
2	265.69	0	0	0	$\frac{\pi}{2} - \arctan\frac{30}{264}$
3	30	$\frac{\pi}{2}$	0	0	$\frac{\pi}{4} + \arctan\frac{30}{264}$
4	0	$\frac{\pi}{2}$	258	0	0
5	0	-$\frac{\pi}{2}$	0	0	0
6	0	0	0	0	0

Table 5: Link1 parameters

Link1 (30)	
Mass [Kg]	9.4360595e-01
Center of Gravity [m]	x : 0.0000000 y: -1.7653633e-04 z : -1.0030209e-03
Inertia Tensor at center[Kg * m2]	Ixx Ixy Ixz : 1.5694005e-03 0.0000000e+00 0.0000000e+00 Iyx Iyy Iyz : 0.0000000e+00 4.5593385e-04 6.4581824e-09 Izx Izy Izz : 0.0000000e+00 6.4581824e-09 1.5561809e-03
Principal Moments of Inertia [Kg * m2]	I1 : 4.5498451e-04 I2 : 1.2037112e-02 I3 : 1.2989528e-02

Table 6: Link2 parameters

Link2 (31)	
Mass [Kg]	1.3825862e+00
Center of Gravity [m]	x : 1.5751501e-02 y: -2.2073221e-04 z : 1.9913687e-01
Inertia Tensor at center[Kg * m2]	Ixx Ixy Ixz : 1.2803228e-02 -2.4795661e-05 -1.4109928e-03 Iyx Iyy Iyz : -2.4795661e-05 1.2037834e-02 1.4050354e-05 Izx Izy Izz : -1.4109928e-03 1.4050354e-05 2.2600348e-03
Principal Moments of Inertia [Kg * m2]	I1 : 2.0744566e-03 I2 : 1.2037112e-02 I3 : 1.2989528e-02

Table 7: Link3 parameters

Link3 (32)	
Mass [Kg]	1.2126965e+00
Center of Gravity [m]	x : 3.0352597e-04 y: 4.1703880e-05 z : 3.8074728e-01
Inertia Tensor at center[Kg * m2]	Ixx Ixy Ixz : 4.0514666e-03 -9.7882420e-07 5.8582868e-05 Iyx Iyy Iyz : -9.7882420e-07 3.6839091e-03 1.5501165e-06 Izx Izy Izz :5.8582868e-05 1.5501165e-06 1.1420941e-03
Principal Moments of Inertia [Kg * m2]	I1 :1.1409140e-03 I2 : 3.6839076e-03 I3 : 4.0526481e-03

Table 8: Link4 parameters

Link4 (33)	
Mass [Kg]	4.6635550e-01
Center of Gravity [m]	x : -2.1388493e-06 y: -2.2290515e-03 z : 5.1387207e-01
Inertia Tensor at center[Kg * m2]	Ixx Ixy Ixz : 4.2687885e-04 0.0000000e+00 0.0000000e+00 Iyx Iyy Iyz : 0.0000000e+00 2.4774029e-04 3.5109263e-06 Izx Izy Izz : 0.0000000e+00 3.5109263e-06 3.2985146e-04
Principal Moments of Inertia [Kg * m2]	I1 : 2.4759044e-04 I2 : 3.3000130e-04 I3 : 4.2687885e-04

Table 9: Link4 parameters

Link5 (34)	
Mass [Kg]	4.2561606e-01
Center of Gravity [m]	x : -2.1268822e-06 y: 1.8039922e-04 z : 5.9028250e-01
Inertia Tensor at center[Kg * m2]	Ixx Ixy Ixz : 5.1431341e-04 0.0000000e+00 0.0000000e+00 Iyx Iyy Iyz : 0.0000000e+00 4.2820999e-04 1.6136111e-06 Izx Izy Izz : 0.0000000e+00 1.6136111e-06 2.1038697e-04
Principal Moments of Inertia [Kg * m2]	I1 : 2.1037502e-04 I2 : 4.2822194e-04 I3 : 5.1431341e-04