Vision System Tutorial

v1.8.2

Table of Contents

1. Welcome

Welcome to use the tutorials for the Mech-Mind Vision System!

Introduction to the Vision System

What Is the 3D Robot Guidance Solution?

Components of a Typical 3D Vision-Guided Robotic Application

Introduction to Mech-Mind's Software Products

Integration with the Robot System

Getting Started Tutorials

Getting started tutorials guide beginners to deploy the first 3D vision-guided robotic application step by step.

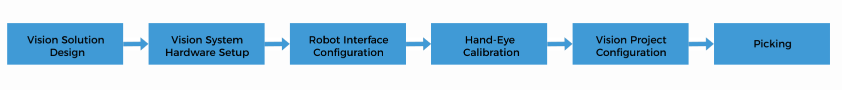
suite-tutorial:getting-started:first-application.pdf

suite-tutorial:single-case-cartons:single-case-cartons.pdf

suite-tutorial:getting-started-interface:first-application-interface.pdf

Practical Application Deployment

The approach to deploying a 3D vision-guided robotic application in actual projects is shown in the following figure:


Vision Solution Design
Vision System Hardware Setup
Robot Interface Configuration
Hand-Eye Calibration
Vision Project Configuration
Picking

Please click the links to view the detailed explanation of each deployment phase.

1 Vision Solution Design

2 Vision System Hardware Setup

3 Robot Communication Configuration

4 Hand-Eye Calibration

5 Vision Project Configuration

6 Picking

Application Optimization

Provides guidance on optimizing applications, such as improving picking accuracy and optimizing cycle time.

Topic: Improving Picking Accuracy

Ensure Qualified Picking Accuracy

Troubleshoot Picking Errors

1.1. Introduction to 3D Vision-Guided Robotics Solution

What Is the 3D Vision-Guided Robotics Solution?

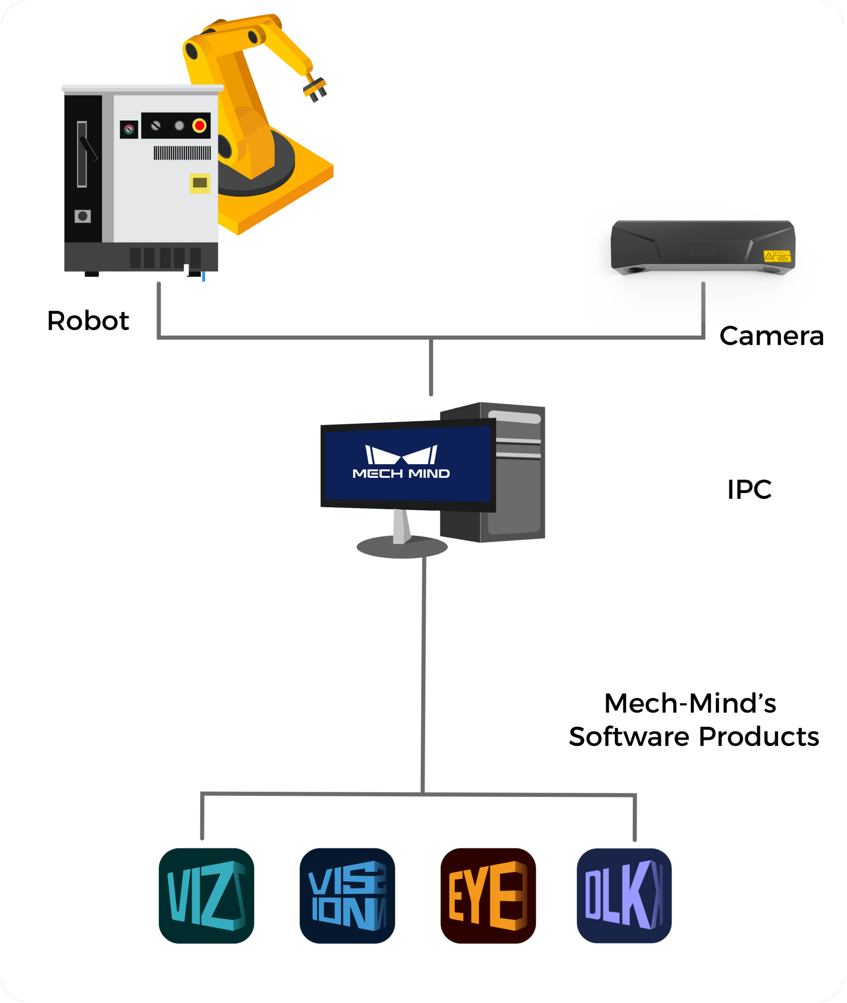
The 3D vision-guided robotics solution integrates Mech-Mind's sensing, perception, robot programming/planning technologies into a comprehensive solution, aiming to address the challenges that robots face in complex, unknown, or dynamic environments.

This solution offers a wide range of applications such as 3D vision guided workpiece loading, depalletizing and palletizing, locating and assembly, and piece picking. It is widely used in various industries such as automotive, logistics, supermarkets, and heavy industry.

By integrating the Mech-Mind Vision System into the robot system, a complete vision-guided robotic application can be built to guide the robot to perform intelligent tasks.

Components of a Typical 3D Vision-Guided Robotic Application

A typical vision-guided robotic application usually consists of the robot, camera(s), industrial personal computer (IPC), and Mech-Mind's software products.



Robot

A robot is a programmable multi-purpose handling device with some autonomy that can perform tasks such as movement, manipulation, or positioning. In the Mech-Mind Vision System, it performs intelligent tasks based on the results output by the vision system.

- A robot cell usually consists of a robot body, a controller, and a teach pendant.
- In industrial applications that require a high level of automation, a programmable logic controller (PLC) can be used to control the motion and operation of the robot. If a higher level of control and monitoring is required, a host computer can also be used to perform more complex programming and control of the robot, such as path planning, task planning, and motion control. The Mech-Mind Vision System supports collaboration with the PLC and host computer.
- In this article, “robot side” is used as a generic term for the robot, PLC, or host computer.

Camera

It is the Mech-Eye industrial 3D camera developed by Mech-Mind, which is used to capture image and location information of the objects.

IPC

It refers to the computer that provides the operating environment for the Mech-Mind’s software products.

Mech-Mind’s software products

Mech-Mind’s software products perform vision processing based on image and location information captured by the camera, and output the location and orientation of the objects and the planned motion path of the robot, thus guiding the robots to complete intelligent tasks, such as picking, depalletizing and palletizing, gluing, sorting, etc. The functions provided by different Mech-Mind software vary. For details, refer to Introduction to Mech-Mind’s Software Products.

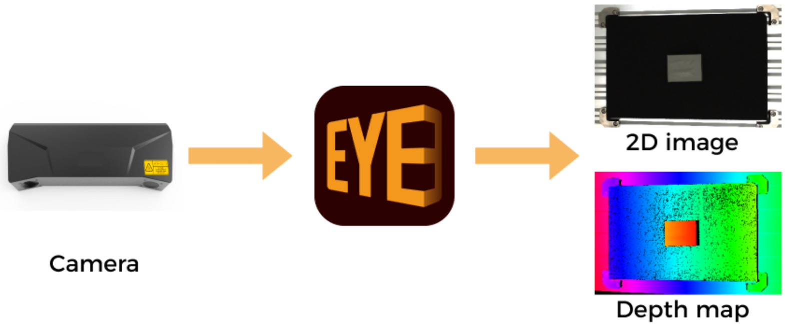
- In this article, the cameras and IPC provided by Mech-Mind, and Mech-Mind’s software products constitute the Mech-Mind Vision System.

Introduction to Mech-Mind’s Software Products

The Mech-Mind Vision System mainly involves the following software:

- Mech-Eye Viewer

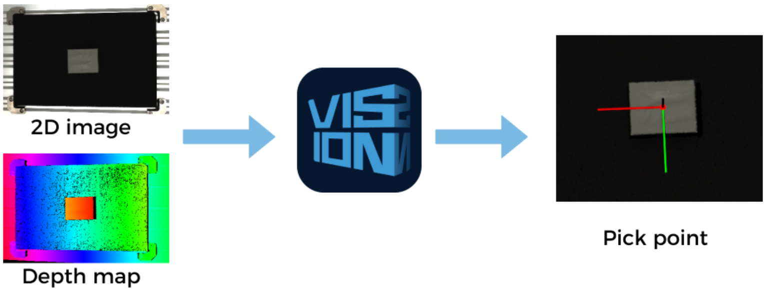
  Mech-Eye Viewer allows users to adjust the parameters of the Mech-Eye industrial 3D camera according to the characteristics of the target object, and obtain high-quality 2D images, depth maps, and point clouds.



- Mech-Vision

  Mech-Vision is a state-of-the-art machine vision software. With its fully graphical interface, advanced machine vision applications such as piece picking, high-precision positioning, assembly, industrial inspection/measurement, and automatic path planning can be completed without writing codes.

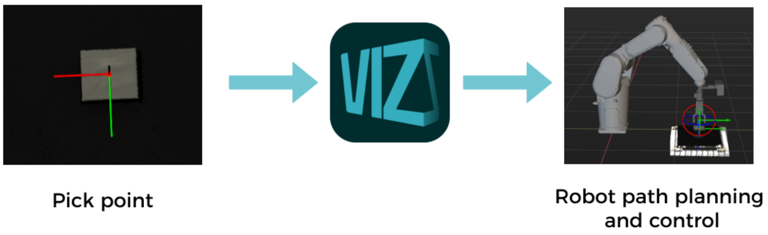
Based on the image data captured on-site, this software performs a series of vision processing and outputs the vision results (such as the position and orientation of the target object). In addition, based on the vision results, this software can perform collision-free path planning for the robot, and output the planned picking path.



- Mech-Viz

  Mech-Viz is a robot path planning software. It uses the information provided by Mech-Vision, including the point clouds and workpiece positions, to intelligently plan the robot path for picking, carrying, palletizing and depalletizing, and other complex application scenarios.

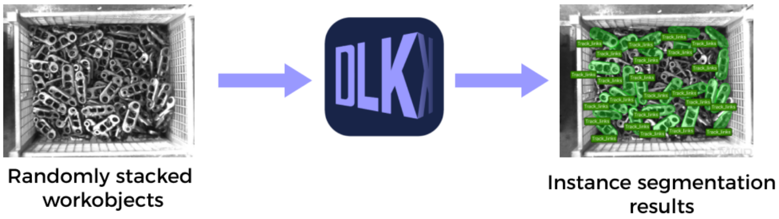
  This software allows users to build a workflow for the robot in a visualized manner and provides a 3D simulation function for validation before using the real robot. It has already been adapted to most major robot brands around the world.



- Mech-DLK

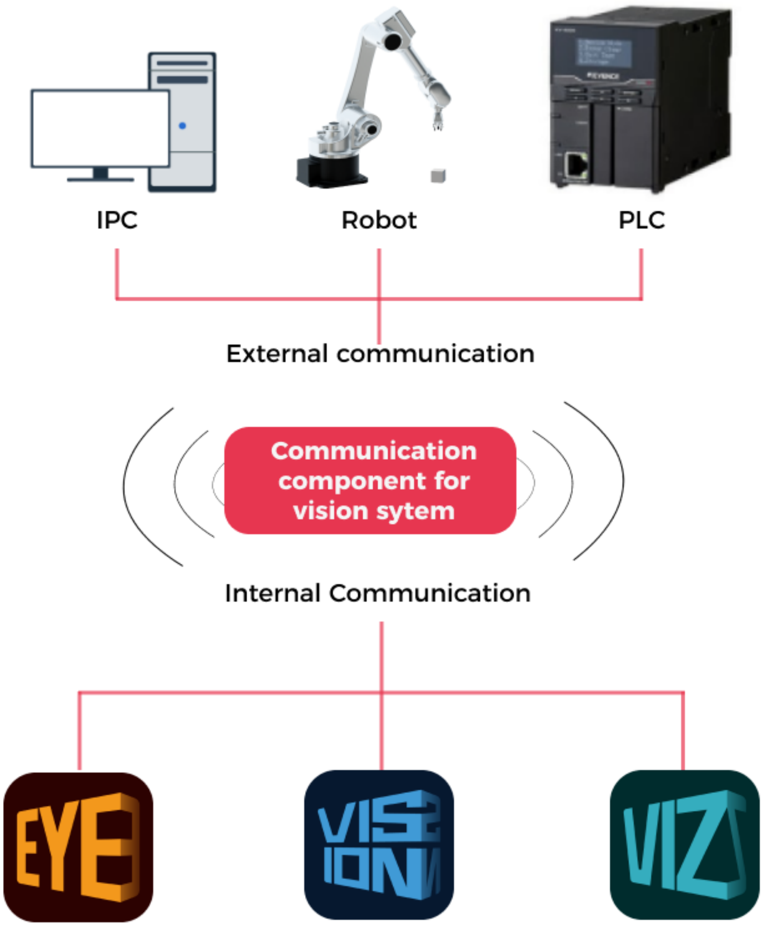
  Mech-DLK is machine vision deep learning software. With a variety of built-in industry-leading deep learning algorithms, it can solve many problems that traditional machine vision cannot handle, such as highly difficult segmentation, positioning, and classification.

  Through intuitive and simple UI interactions, even without programming or specialized deep learning knowledge, users can quickly implement model training and validation with this software.

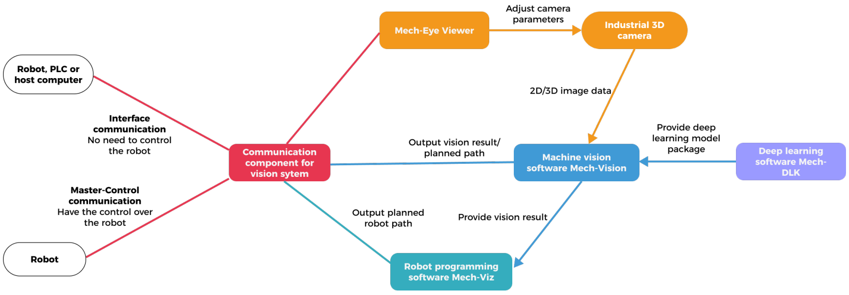


- Communication Component for the Vision System

The communication component for the vision system (runs in the background) enables data exchange among software within the vision system and provides standardized and customized communication with external devices, such as robots, PLCs, and host computers, which makes vision-guided intelligent robotic applications possible.



The relationship between Mech-Mind’s software products is shown in the following figure.


Mech-Eye Viewer
Adjust camera parameters
Industrial 3D camera
Robot, PLC or host computer
2D/3D image data
Interface communication
No need to control the robot
Provide deep learning model package
Output vision result/ planned path
Communication component for vision sytem
Machine vision software Mech-Vision
Deep learning software Mech-DLK
Master-Control communication
Have the control over the robot
Output planned robot path
Provide vision result
Robot
Robot programming software Mech-Viz

Integration with the Robot System

The Mech-Mind Vision System provides external communication interfaces, which allow it to communicate and integrate with the robot, PLC or host computer within the robot system. This enables the vision system to return the vision results or planned robot path (by Mech-Vision or Mech-Viz) to the robot side for use.

The Mech-Mind Vision System supports the following communication modes:

▶ https://docs.mech-mind.net/download/video/communication/communication-modes-en.mp4 *(video)*

- Master-Control communication

  In this mode, the vision side has control over the robot, i.e., the vision system acts as the master device, while the robot as the slave device. The vision system sends motion or IO commands to the robot, and the robot continuously listens to and executes these commands. The vision system controls the robot to perform tasks based on the planned path, such as loading or depalletizing/palletizing.

  When this mode is used, the robot can be controlled by loading a master-control program to the robot or by using the robot's SDK (Software Development Kit). This mode supports communication between the vision system and the robot.

- Interface communication

  In this mode, the vision system does not need to take control of the robot. Typically, the robot side acts as the master device, while the vision system is the slave device. The robot side and the vision system use the same standard communication protocol to communicate (such as TCP socket). The robot side sends requests while the vision system processes them and sends responses (poses and labels of the target objects) back. Depending on the request, the vision system returns either the vision result or the planned picking path. The robot makes further decisions or performs the appropriate tasks according to the responses returned from the vision system.

  When this mode is used, you should write the robot interface program (for communicating with the vision system) and the robot application program (for receiving the data returned from the vision system and controlling the robot to perform tasks), and load them to the robot. This mode supports communication between the vision system and the robot, the PLC, or the host computer.

For more information about the communication modes, refer to the section Communication Basics.

1.2. Practical Application Deployment

1.2.1. Vision Solution Design

Before you deploy the application, you need to first design the vision solution and select the camera model, IPC model, camera mounting mode, robot communication mode, and more according to the actual needs of your project. A good design helps to quickly deploy a 3D vision-guided application.

During the design phase of a 3D vision-guided application, you need to complete the following steps:

- Select camera model
- Determine camera mounting mode
- Select IPC model
- Select robot communication mode
- Determine whether to use deep learning

If the project requires a high picking accuracy, ensure a good picking accuracy of the application during deployment according to the guidance in Topic: Improving Picking Accuracy.

Select Camera Model

Mech-Eye industrial 3D camera is a high-performance camera developed by Mech-Mind. It features high accuracy, fast data acquisition, resistance to ambient light, and high-quality imaging and can generate high-quality 3D point cloud data of a variety of objects. Mech-Mind offers full model options to meet the needs of ambient light resistance, high accuracy, large field of view, high speed, and small size at different distances.

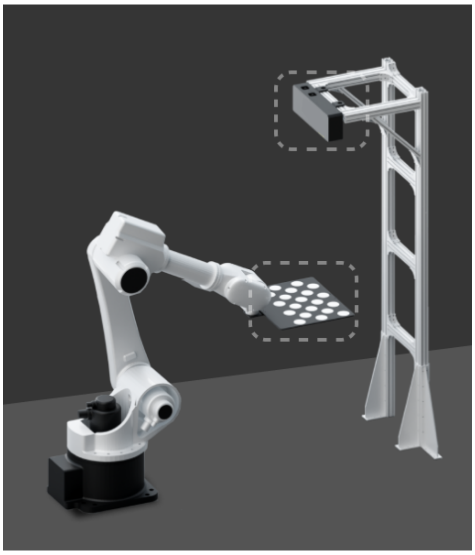
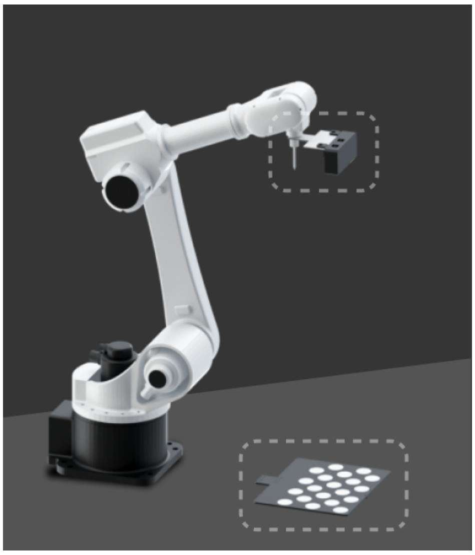
In real practice, please select the appropriate model according to the working distance, field of view, and accuracy of the camera.

To select the appropriate camera model, follow these steps:

1. Find the appropriate camera model by checking the Camera Models topic.
2. Use the Mech-Eye Industrial 3D Camera FOV Calculator to validate whether the selected camera can meet the project requirements.

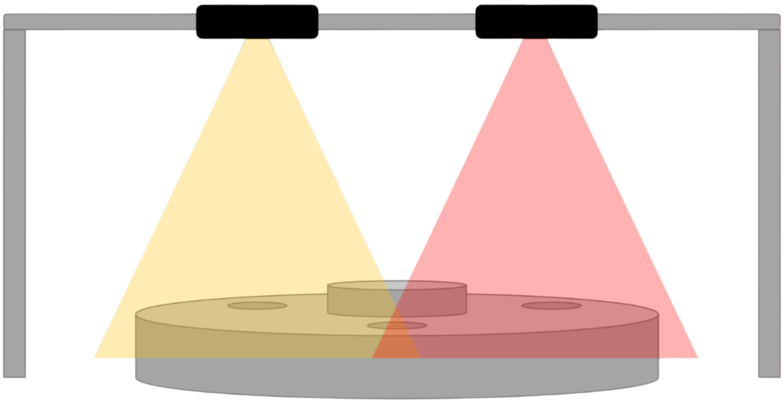
Determine Camera Mounting Mode

You can determine the camera mounting mode based on the relative position of the camera to the robot and the needs of the overall cycle time. The following table shows two common camera mounting modes.

Mounting mode	Eye to hand (ETH)	Eye in hand (EIH)
Description	The camera is mounted on a camera mounting frame independent of the robot.	The camera is mounted on the last joint of the robot and moves with the robot.
Illustration		

Characteristics	• This mounting mode allows the camera to capture images at a fixed position and angle, allowing for a larger field of view. • The robot does not need to wait for the camera to capture images while performing a task, thus increasing productivity and reducing cycle time.	• This mounting mode allows for flexible image-capturing positions and angles. The camera is mounted on the robot’s last joint and moves and positions itself accurately with the robot, allowing the camera to capture images of multiple stacks; the robot can adapt to incoming materials with different stack heights by raising the camera. • The robot needs to wait for the camera to capture images while performing the task, thus lengthening the overall cycle time.

In addition, to expand the camera’s field of view and improve the quality of the overlapped point cloud, a project may have two or more cameras installed for one station, which is called Eye to eye (ETE).



Regardless of the camera mounting mode you choose, you will need a camera mounting frame to mount the camera. For design instructions on camera mounting frames, please refer to Camera Mounting Frame Designs.

Select IPC Model

Mech-Mind IPC provides the standard operation environment for Mech-Mind’s software products and therefore can maximize the function and performance of the software.

Please select the appropriate IPC model according to the application scenarios and technical specifications of IPCs.

If you use your own computer or laptop (“non-standard IPC” for short) as the industrial personal computer (IPC), please refer to Non-standard IPC Setup to make sure the non-standard IPC meets the system configuration requirements and complete the corresponding settings.

Select Robot Communication Mode

The interface communication mode is usually suitable for practical applications on the production line, providing more flexible functionality as well as a shorter cycle time. The Master-Control communication mode is usually applied in the testing phase of a project for quick verification of the picking effect.

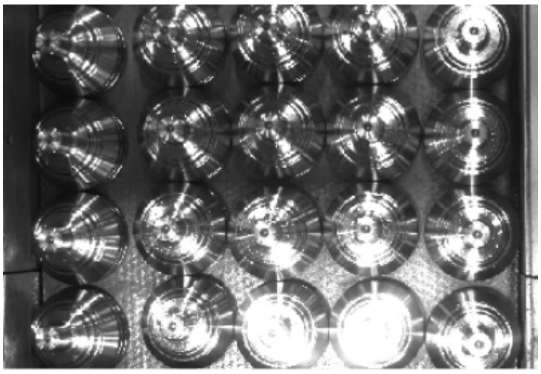
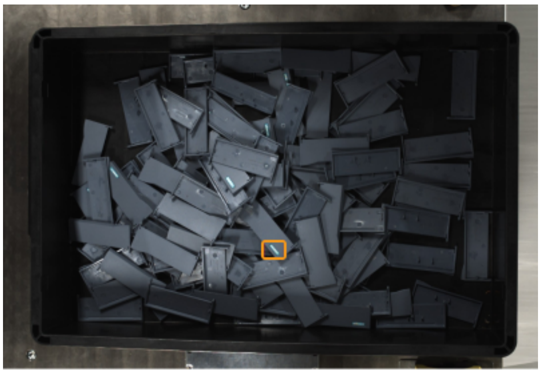
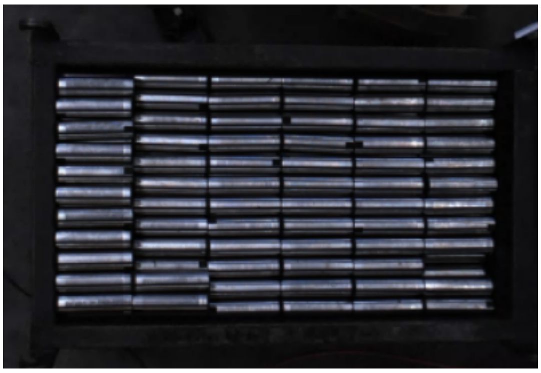
▶ https://docs.mech-mind.net/download/video/communication/communication-modes-en.mp4 *(video)*

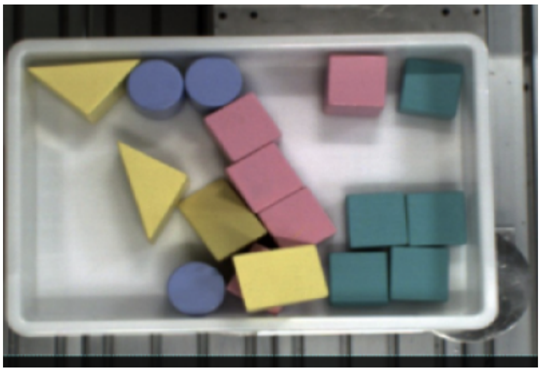
For more information about how to select the communication mode, please refer to the section Communication Mode Selection. For more information about the communication modes, please refer to the section Communication Basics.

Determine whether to Use Deep Learning

Please refer to the Select Deep Learning Solution section to determine whether to use deep learning in your vision solution.

3D matching may not solve the following problems effectively during the vision recognition process. In this case, you need to use deep learning.

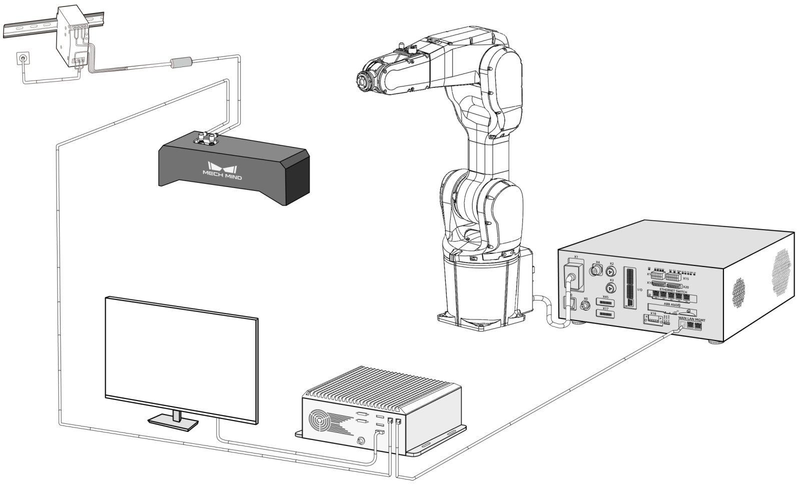
No.	Challenges to traditional methods	Illustration
1	The surfaces of the objects are highly reflective, and the quality of the point cloud is poor.	
2	The curve of the object point cloud is not obvious, and there are few feature points in the point cloud.	
3	When the objects are arranged neatly and placed closely against each other, the performance of point cloud clustering is poor.	

No.	Challenges to traditional methods	Illustration
4	The object feature is only visible in the color image but not visible in the point cloud.	
5	The project has a high requirement for the vision cycle time. When there is a large number of objects, the matching time using the point cloud model can be long.	

1.2.2. Vision System Hardware Setup

Vision system hardware setup refers to integrating the hardware (camera and industrial PC) into the actual environment to support the normal operation of the vision system.

In this phase, you need to install and connect the hardware of the vision system.



Follow the steps below to build the Mech-Mind Vision System: **Check the contents of the package** → **Install the hardware** → **Connect the network** → **Upgrade the software (optional)** → **Confirm that the vision system can capture images normally**.

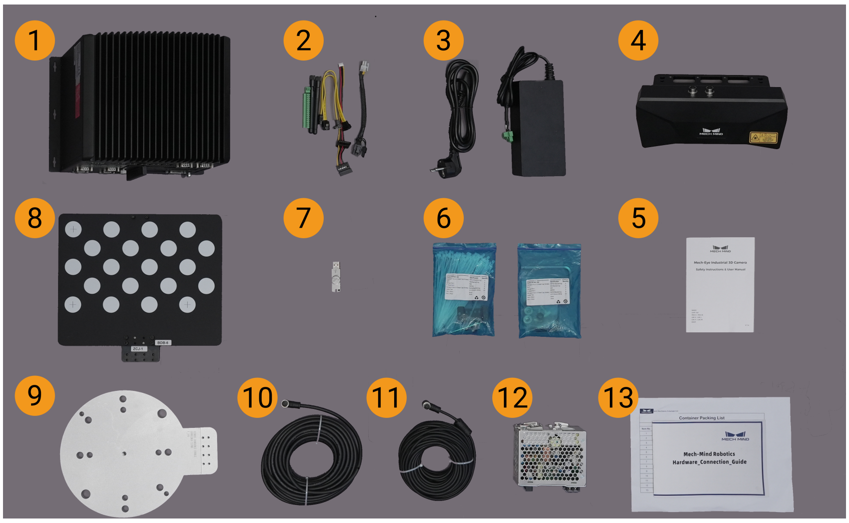
If the project requires a high picking accuracy, ensure a good picking accuracy of the

> application during deployment according to the guidance in Topic: Improving Picking Accuracy.

Check the Contents of the Package

1. Make sure that the package is intact when you receive it.
2. Check the contents against the "packing list" in the package to ensure that no devices or accessories are missing or damaged.

The following figure shows the devices and accessories included in a vision system shipment. The table below is for reference only. Please take the "packing list" in the package as final.



No.	Category	Name	Function
1	IPC and accessories	IPC	Provides running environment for Mech-Mind software
2		IPC accessories	Provides IPC accessories, such as antennas for WIFI connection
3		IPC power cable and adapter	Supplies power to IPC
4	Camera and accessories	Mech-Eye Industrial 3D Camera	Captures images
5		User manual	Mech-Eye Industrial 3D Camera User Manuals
6		Camera accessory box	Mounts the camera

No.	Category	Name	Function
7	Project accessories	License dongle	Authorizes the Software Suite license
8		Calibration board	Calibrates the camera
9		Flange adapter	Connects the calibration board to the robot flange
10		Camera DC power cable (standard: 20 meters)	Connects the camera and the DIN rail power supply; Longer power cables are an option
11		Camera Ethernet cable (standard: 20 meters)	Connects the camera and the IPC; longer camera Ethernet cables are an option
12		DIN rail power supply (standard)	Supplies power to Mech-Eye Industrial 3D Camera; The camera power adaptor is an option
13	Packing list		Lists all the devices and accessories in the package

Contact Mech-Mind if any items are missing or damaged.

Prepare Other Materials

In this tutorial, besides the items in the package, you still need to prepare the materials shown in the following table by yourself.

Item	Function
Monitor	Provides display for the IPC
HDMI cable	Connects the monitor and the IPC
RJ45 Ethernet cable	Connects the IPC and the robot controller

The IPC and robot controller usually are directly connected through an RJ45-to-RJ45 Ethernet cable, and the IPC and the camera are directly connected through the camera Ethernet cable. Alternatively, you can use a router to connect the IPC and the robot controller, and the IPC and the camera, which is not covered in this topic.

Install Hardware

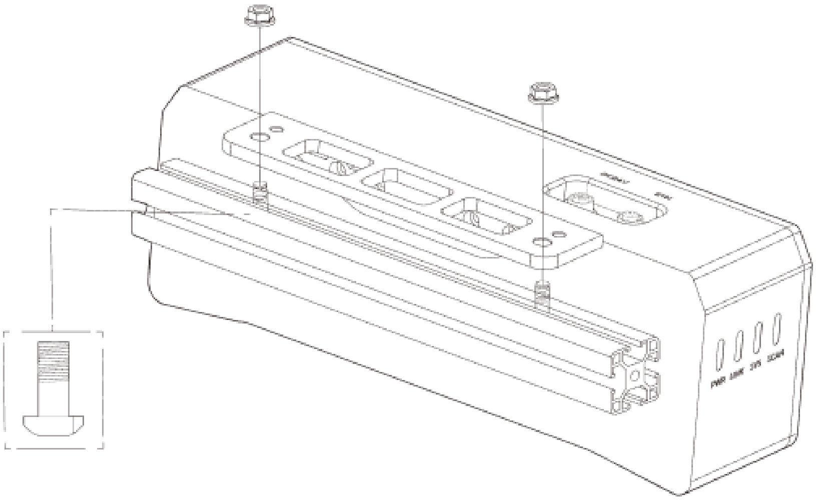
If the selected camera mounting mode is eye to hand (ETH) in the vision solution design stage, please follow the section Mount Camera (ETH); if the selected mounting mode is eye in hand (EIH), please follow the section Mount Camera (EIH).

Mount Camera (ETH)

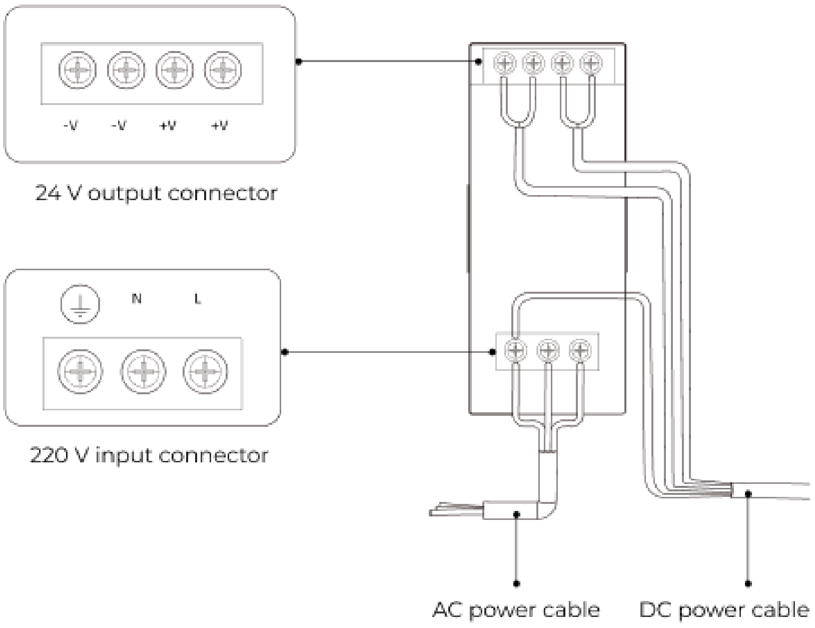
In the ETH mode, the camera is fixed on a stand independent of the robot.

Mount the camera mounting brackets according to the section Camera Bracket Design and Mounting in ETH Setup. After the camera bracket is mounted securely, you can start to mount the camera.

1. Get the camera mounting bolts and wrench from the camera accessory box.
2. Tighten the two bolts with the wrench as shown below.

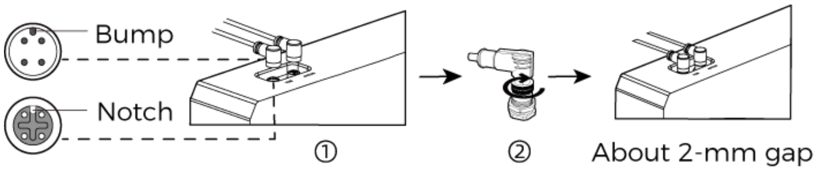


3. Please remove the lens protection film after mounting the camera.
4. Power the camera with the DIN rail power supply.
   - Connect the DC power cable:
     - Connect the +V wire to the +V connectors of the 24 V output connectors;
     - Connect the -V wire to the -V connectors of the 24 V output connectors;
     - Connect the PE wire to the 220 V input connector ⏚.



5. Install the Ethernet cable of the camera.

   Make sure the bump of the M12 connector and the notch of the ETH port, and tighten the nut after plugging in the cable.



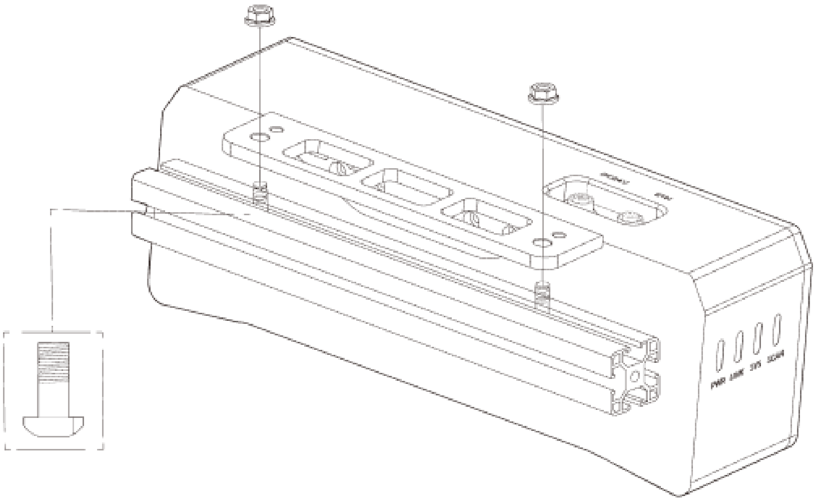
6. Secure, wire, and route the camera Ethernet cables and power cables according to the section Guidelines for Mounting Camera Cables.

Mount Camera (EIH)

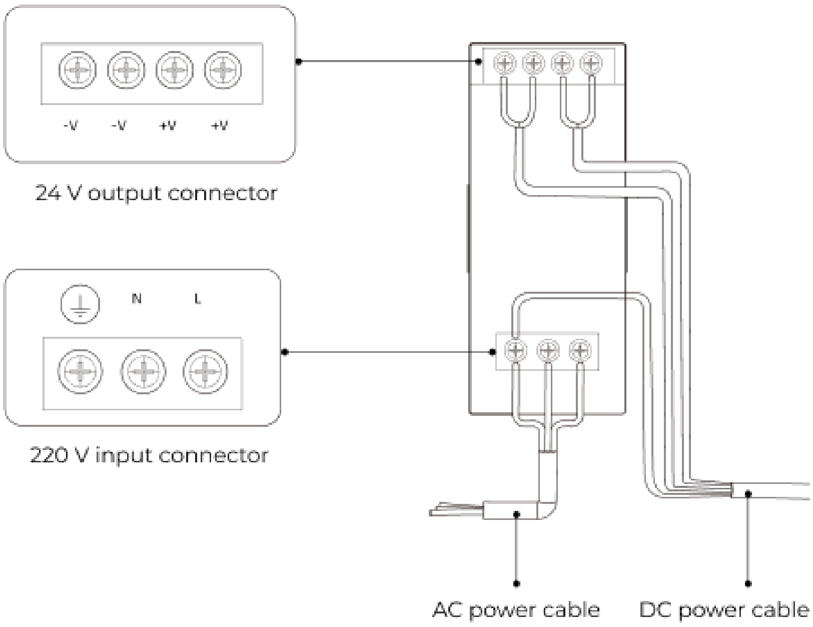
In the EIH mode, the camera is mounted on the last joint of the robot through a camera mounting bracket.

Mount the camera mounting brackets according to the section Camera Bracket Design and Mounting in EIH Setup. After the camera bracket is mounted securely, you can start to mount the camera.

1. Get the camera mounting bolts and wrench from the camera accessory box.
2. Tighten the two bolts with the wrench as shown below.

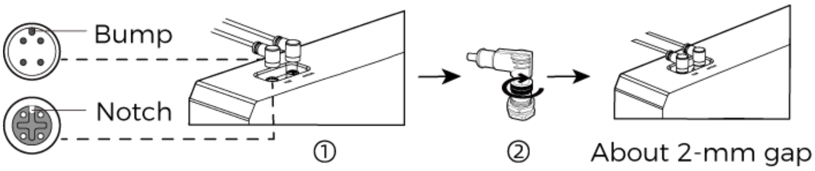


3. Please remove the lens protection film after mounting the camera.
4. Power the camera with the DIN rail power supply.
   - Connect the DC power cable:
     - Connect the +V wire to the +V connectors of the 24 V output connectors;
     - Connect the -V wire to the -V connectors of the 24 V output connectors;
     - Connect the PE wire to the 220 V input connector ⏚.



5. Install the Ethernet cable of the camera.

   Make sure the bump of the M12 connector and the notch of the ETH port, and tighten the nut after plugging in the cable.



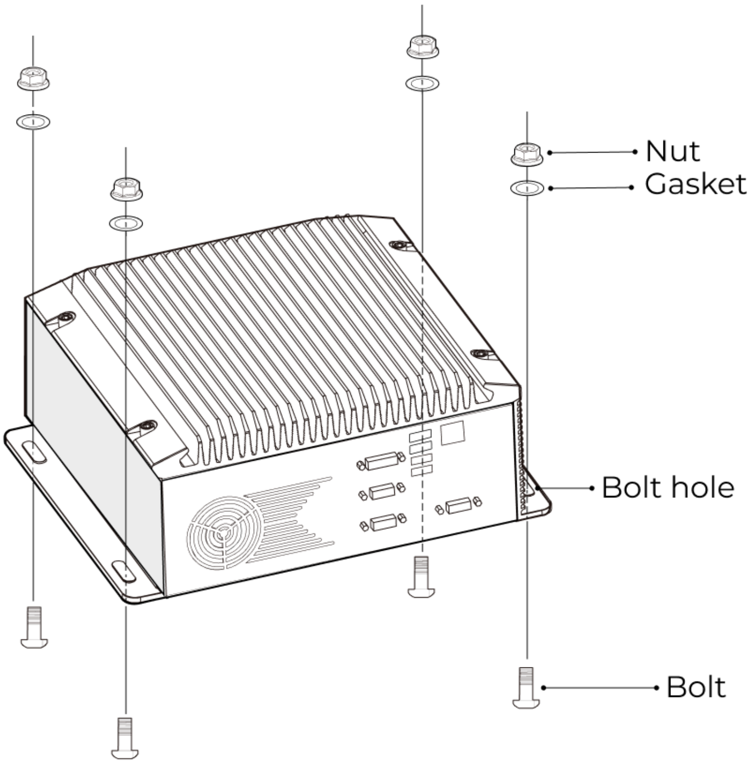
6. Secure, wire, and route the camera Ethernet cables and power cables according to the section Guidelines for Mounting Camera Cables.

Mount the IPC

The IPC is normally mounted in the control cabinet of the robot. The environment in which the IPC is mounted requires good heat dissipation, ventilation, and dust protection. It should be mounted at a location where Ethernet cables, HDMI cables, and USB ports can be easily mounted and maintained.

To mount the IPC, follow these steps:

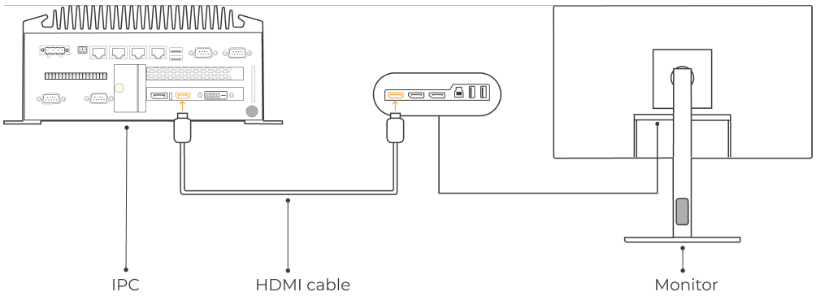
1. Prepare the wrench, bolts, nuts, and gaskets that are not included in the package beforehand.
2. If the robot controller is designed with mounting holes for the IPC in it, secure the IPC in the controller: place the bolt, gasket, and nut one by one, and tighten the two bolts with a wrench, as shown below.



If the location of the robot controller is already fixed, skip this step and just place the IPC inside the controller.

3. Connect the IPC and the monitor with the HDMI cable.

   Plug one end of the HDMI cable into the HDMI port of the monitor, and the other end into the HDMI port of the IPC, as shown below.



4. Connect the IPC to the power supply unit with the power adaptor.

   Plug the power cable of the power adaptor into the power connector of the IPC. Connect the adaptor to the power supply on the other end.

5. Insert the license dongle.

   Plug the license dongle into a USB port of the IPC.

6. After the IPC is connected to the power supply, switch on the IPC.

   - If the IPC is started normally, the power indicator should be solid on.

- If the IPC cannot be started, contact Mech-Mind Technical Support.

Connect the Network

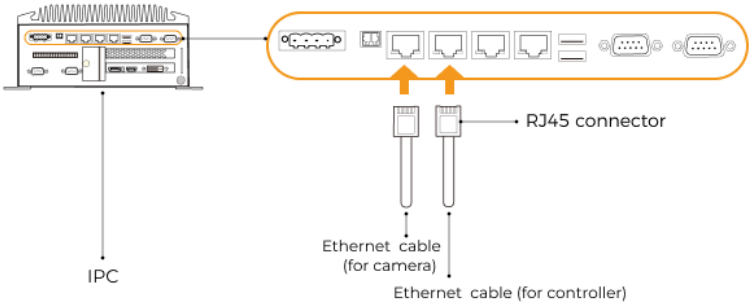
In this section, you will learn how to connect the network between the IPC and the camera, and between the IPC and the robot.

In the following sections, the following IP addresses will be used for network settings. Please adjust the network settings according to your actual network environment.

Device		IP Address
IPC	Ethernet port connecting to the camera	192.168.100.10
	Ethernet port connecting to the robot controller	192.168.200.10
Camera		192.168.100.20
Robot		192.168.200.20 (already set on the robot)

Connect the IPC and the Camera, and the IPC and the Robot Controller

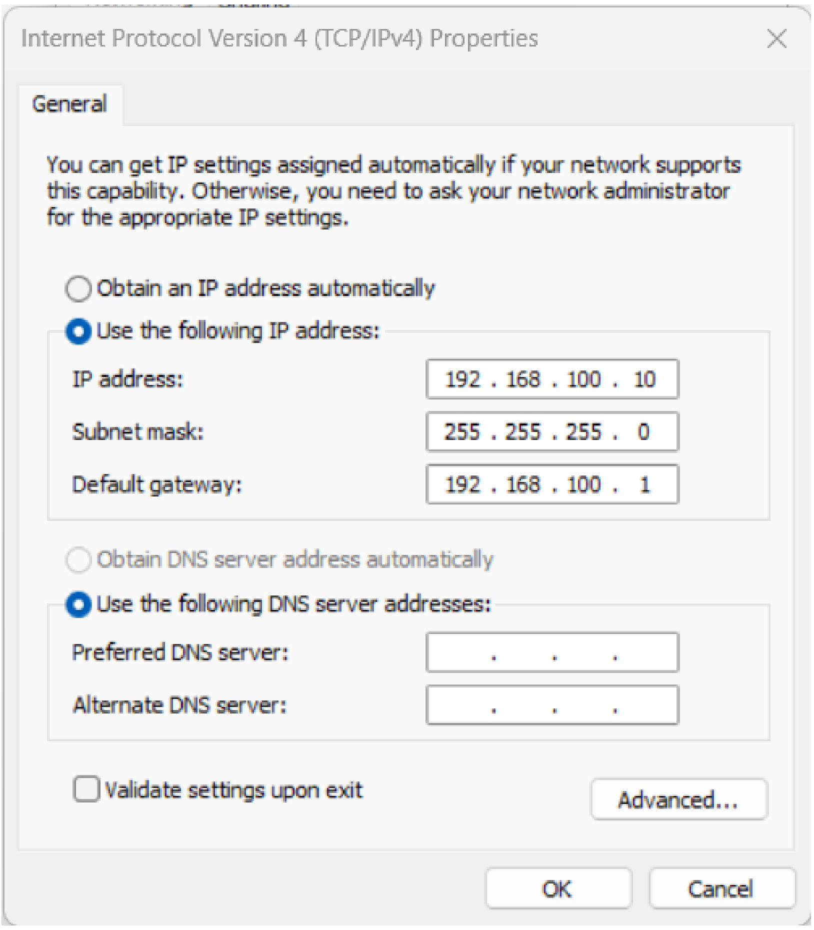
1. Plug the other end of the Ethernet cable connected to the camera into an Ethernet port of the IPC.



2. Use an RJ45-to-RJ45 Ethernet cable to plug one end into the Ethernet port of the robot controller and the other end into an Ethernet port of the IPC.

Set the IP Addresses on the IPC

1. Select Control Panel › Network and Internet › Network and Sharing Center › Change adapter settings on the IPC. The **Network Connections** page will be displayed.
2. Right-click the Ethernet port connected to the camera, and select **Rename** to rename the Ethernet port, such as “To_camera”.
3. Right-click the Ethernet port connected to the camera, and select **Properties** to enter the **Ethernet Properties** page.
4. Select the **Internet Protocol Version 4 (TCP/IPv4)** checkbox, and then click the **[Property]** button to enter the **Internet Protocol Version 4 (TCP/IPv4) Properties** page.
5. Select the **Use the following IP address** radio button, set the **IP address** to “192.168.100.10”, **Subnet mask** to “255.255.255.0”, and **Default gateway** to “192.168.100.1”, and then click the **[OK]** button.



Internet Protocol Version 4 (TCP/IPv4) Properties

General

You can get IP settings assigned automatically if your network supports this capability. Otherwise, you need to ask your network administrator for the appropriate IP settings.

○ Obtain an IP address automatically

● Use the following IP address:

IP address: 192 . 168 . 100 . 10

Subnet mask: 255 . 255 . 255 . 0

Default gateway: 192 . 168 . 100 . 1

○ Obtain DNS server address automatically

● Use the following DNS server addresses:

Preferred DNS server: . . .

Alternate DNS server: . . .

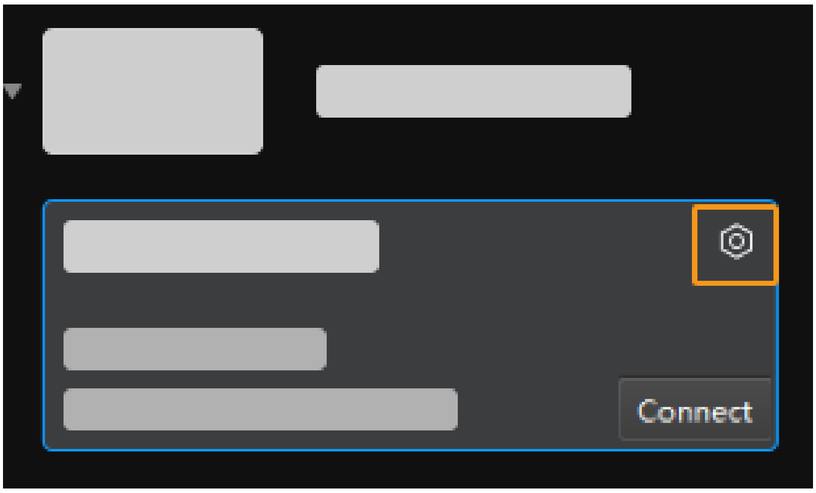
□ Validate settings upon exit

6. Repeat steps 2 to 5 to rename the Ethernet port connected to the robot controller (for example, “To_robot”), and set the IP address for this Ethernet port. For example, the IP address of this Ethernet port is “192.168.200.10”.

> The IP address of the robot and that of the IPC Ethernet port connected to the robot controller must be on the same subnet.

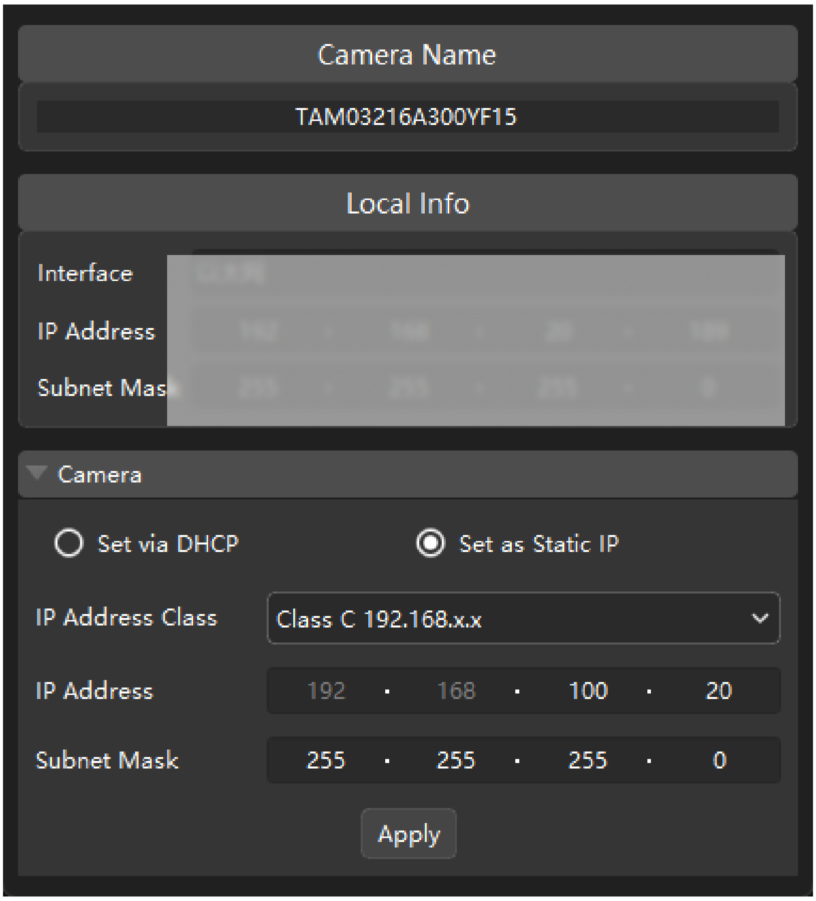
Set the Camera IP Address

1. Double-click on the IPC to open and run Mech-Eye Viewer.
2. Select the camera in the camera list, and hover the cursor on the camera. Click to open the **Config Camera Network** dialog box.



If the camera cannot be found or connected, please refer to Camera Troubleshooting.

3. In the **Camera** area, select the **Set as Static IP** radio button, set **IP Address Class** to “Class C 192.168.x.x”, **IP Address** to **192.168.100.20**, and **Subnet Mask** to “255.255.255.0” and then click **[Apply]**.



Camera Name

TAM03216A300YF15

Local Info

Interface

IP Address

Subnet Mask

Camera

○ Set via DHCP ● Set as Static IP

IP Address Class: Class C 192.168.x.x

IP Address: 192 . 168 . 100 . 20

Subnet Mask: 255 . 255 . 255 . 0

The IP address of the camera and that of the IPC Ethernet port connected to the camera must be on the same subnet.

Test the Network Connectivity

1. Press `Win` + `R` to open the **Run** dialog box.

2. Type cmd in the **Run** dialog box, and then click **[OK]**.
3. Type **ping XXX.XXX.XX.XX** in the command prompt window and press **[Enter]** to execute the command.

   💡 Replace XXX.XXX.XX.XX with the actual IP address of the camera/robot.

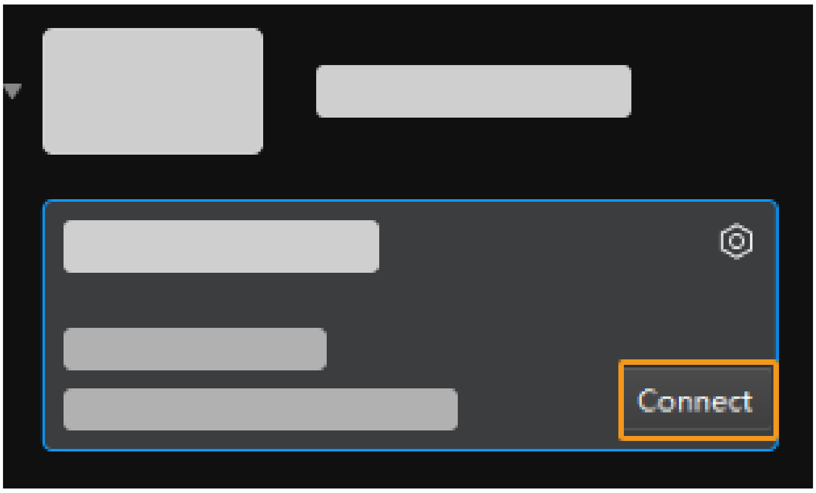
If the network connectivity is normal, you should receive the following response:

```
Pinging XXX.XXX.XX.XX with 32 bytes of data:
Reply from XXX.XXX.XX.XX: bytes=32 time<1ms TTL=128
Reply from XXX.XXX.XX.XX: bytes=32 time<1ms TTL=128
Reply from XXX.XXX.XX.XX: bytes=32 time<1ms TTL=128
Reply from XXX.XXX.XX.XX: bytes=32 time<1ms TTL=128
```

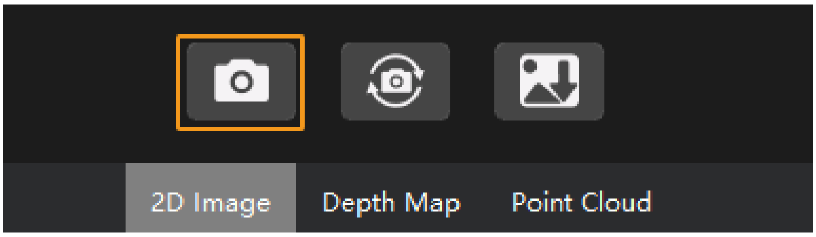
Confirm the Quality of Captured Images

After testing the network connectivity between the IPC and the camera/robot, please confirm that the vision system can capture images normally and confirm that the captured images meet quality standards:

1. Place workpieces at the center of the camera’s field of view (FOV), and make sure that the workpieces on the edge and top are all within the FOV.
2. Open and start Mech-Eye Viewer by double-clicking EYE on the IPC.
3. Select the camera in the camera list and click **[Connect]**.

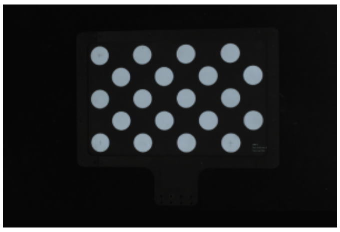
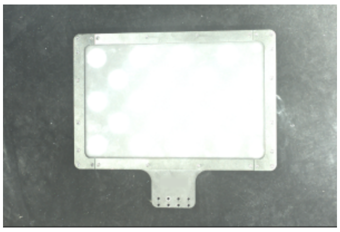
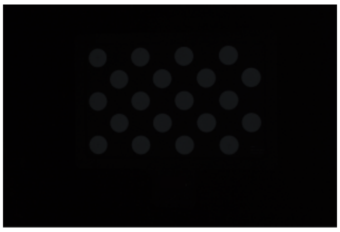
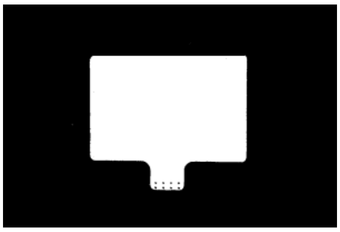
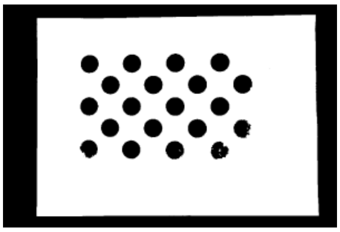
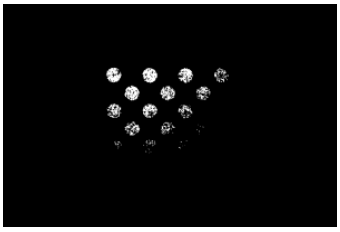


4. After the camera is connected, click **[Capture once]**.



5. Make sure that the captured images meet the following standards.

- 2D image: The captured 2D image is not significantly overexposed (too white to recognize objects) or underexposed (too dark to recognize object details).
- Depth map and point cloud: All relevant parts of the object can be recognized.

	Normal	Overexposed	Underexposed
2D image			
Point cloud			

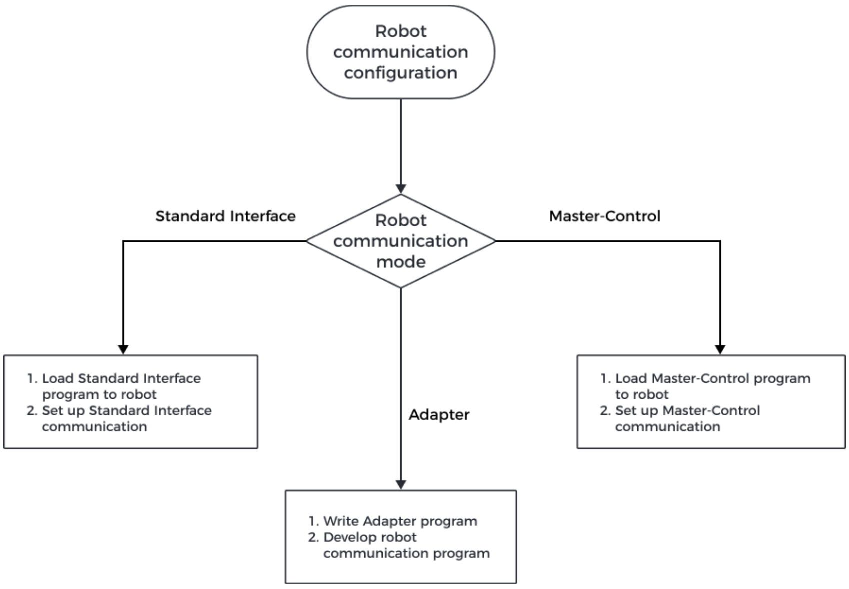
- If the captured images do not meet the required standard, please use Mech-Eye Viewer to adjust camera parameters.
- If the vision solution needs to use deep learning, please confirm the quality of captured images and capture the required quantity of images according to the section Image Capture Requirements for Deep Learning.

1.2.3. Robot Communication Configuration

You have completed the hardware integration of the vision system with the robot system in the Vision System Hardware Setup section. In this section, you will complete the communication integration between the two, ensuring that the vision system and the robot side can exchange data accurately in real time.

To configure the robot communication, you usually need to load the communication program and configuration files provided by Mech-Mind into the robot system and complete the corresponding communication configurations.

Depending on the different robot communication modes selected in the vision solution design phase, the configuration you need to complete in this phase also varies.


Robot communication configuration
Robot communication mode
Standard Interface
Master-Control
Adapter
1. Load Standard Interface program to robot
2. Set up Standard Interface communication
1. Load Master-Control program to robot
2. Set up Master-Control communication
1. Write Adapter program
2. Develop robot communication program

- If you have selected the Standard Interface communication mode in the vision solution design phase, you need to load the Standard Interface program and configuration files of the robot into the robot system in this phase. You also need to complete the corresponding communication configurations in this mode.

  You can search for the User Manual about the Standard Interface communication according to your robot or PLC brand in the Standard Interface Communication section.

- If you have selected the Master-Control communication mode in the vision solution design phase, you need to load the Master-Control program and configuration files of the robot into the robot system in this phase. You also need to complete the corresponding communication configurations in this mode.

  You can search for the User Manual about the Master-Control communication according to your robot brand in the Master-Control Communication section.

- If you have selected the Adapter communication mode in the vision solution design phase, you need to write an Adapter program for the vision system and a corresponding robot communication program for the robot side in this phase.

  Adapter communication is a customized communication mode, and thus it requires the user to have strong programming skills. For details about Adapter communication, please refer to Adapter Communication.

1.2.4. Hand-Eye Calibration

The hand-eye calibration establishes the transformation relationship between the camera and robot reference frames (that is camera extrinsic parameters). With this relationship, the object pose determined by the vision system can be transformed into that in the robot reference frame, which guides the robot in performing its tasks.

The hand-eye calibration process may vary with the robot type, communication mode, camera mounting mode, and the calibration data collection method.

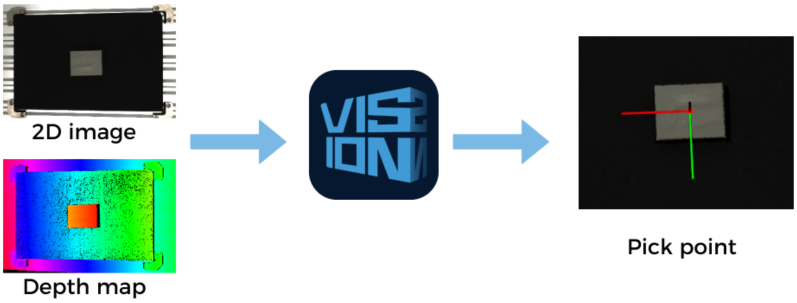
The calibration tool in Mech-Vision provides a standardized calibration process, which helps you complete hand-eye calibration with ease.

You only need to Select the Proper Calibration Process according to the specific scenario of your project and follow the operation guide to complete the hand-eye calibration.

- If the project requires a high picking accuracy, ensure a good picking accuracy of the application during deployment according to the guidance in Topic: Improving Picking Accuracy.
- If any problems occur during the calibration, please refer to Calibration Troubleshooting and FAQs for troubleshooting.

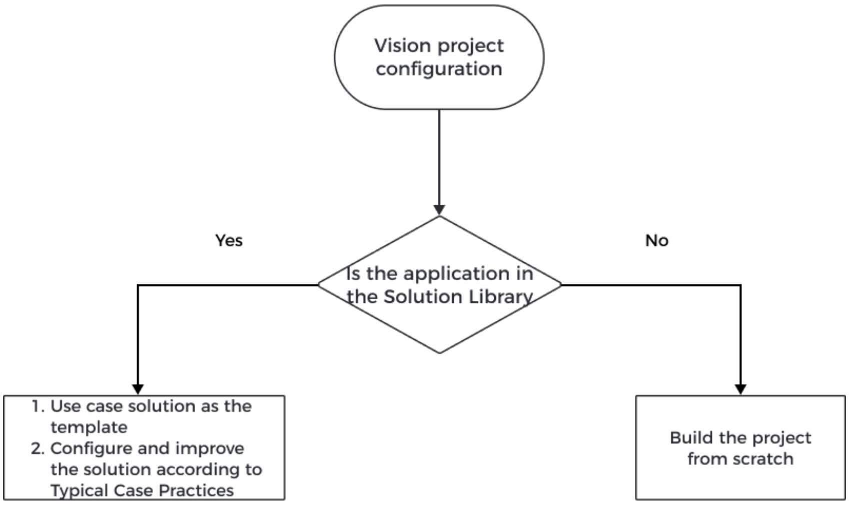
1.2.5. Vision Project Configuration

In this phase, you need to complete configurations on the vision project (that is the Mech-Vision project) to recognize and locate target objects.



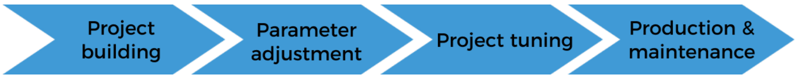
If the project requires a high picking accuracy, ensure a good picking accuracy of the application during deployment according to the guidance in Topic: Improving Picking Accuracy.

A vision project includes a series of vision processing steps, starting from image collection, followed by a series of algorithmic processes on image data (point cloud preprocessing, point cloud processing, 3D matching, pose adjustment, deep learning inference, etc.), finally outputting the vision results (poses, labels, etc.) used to guide the robot.


Vision project configuration
Yes
No
Is the application in the Solution Library
1. Use case solution as the template
2. Configure and improve the solution according to Typical Case Practices
Build the project from scratch

You do not need to build a vision project from scratch for typical application scenarios. For typical application scenarios in industries such as workpiece loading, depalletizing and palletizing, locating and assembly, piece picking, and quality inspection, Mech-Vision has already provided typical solution cases in its Solution Library. You can directly use these solution cases as your project template to create, configure, and tune a vision project quickly. The document “Typical Case Practices of 3D Vision System” will provide you with detailed guidance on configuring and tuning a vision project.

For cases not included in the Solution Library, you need to build the project from scratch and configure and tune it. The process of how to configure a vision project is shown in the figure below:



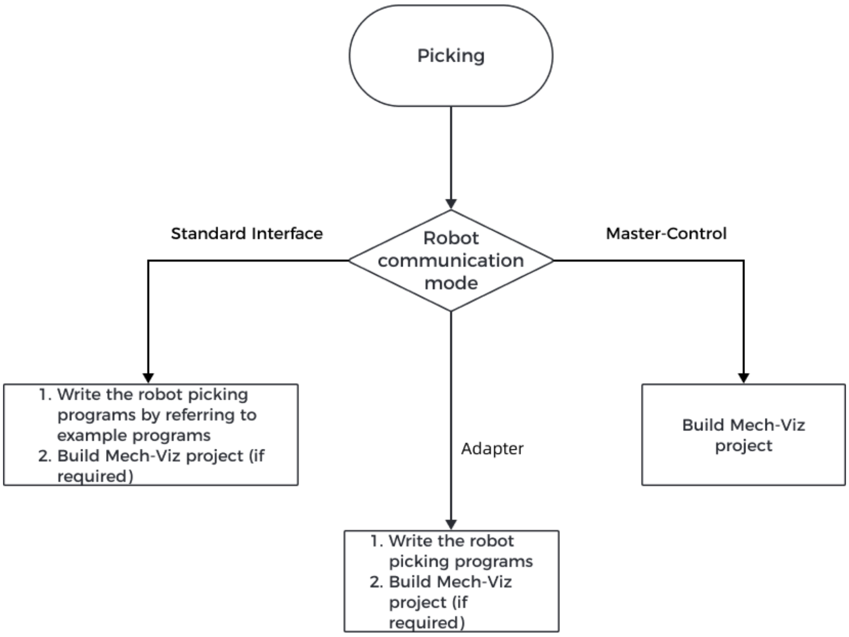
1 Build a project	Project Creation Framework	This section introduces to you how to create a simple project.
	Quick Guide for Projects	This section introduces to you the basic procedures of a project.
	Quick Guide for Steps	This section introduces to you the basic procedures for using Steps. Steps are the basics of a project. A Step is a minimum algorithm unit for data processing. By connecting different Steps in a project, you can achieve different data processing tasks.
2 Parameter adjustment	Adjust Camera Parameters	Adjust the camera parameters to ensure the quality of 2D images and depth maps captured by the camera. The quality of 2D images and depth maps affects the final vision results output by the vision project.
	Make the Point Cloud Model	The 3D matching algorithm recognizes target objects based on their point cloud models. Therefore, when you are configuring a vision project, you usually need to make the point cloud model.
	User Guide of Deep Learning	In scenarios with complex recognition requirements, for example, the workpieces are highly reflective or the quality of point clouds is poor, the 3D matching algorithm may not achieve optimal recognition performance. The deep learning algorithm can facilitate a better recognition performance. Please refer to this user guide to train and deploy a deep learning model.
	Pose Adjustment	After target objects are successfully recognized, you need to perform a series of processing on the poses so that the robot can pick them easily. You can quickly adjust the poses with the pose adjustment tool.
	Step Reference Guide	Please refer to the “Step Reference Guide” to adjust the parameters of each Step in your project.

3 Project tuning	Visualize Parameter Debugging	Run each Step to check the execution result in the Debug Output window, and run the whole project to check the vision results output from your project.
	Tune Deep Learning Model	In scenarios where deep learning is used, if the inference result is not satisfying, you need to fine-tune your deep learning model.
	Save Data	To facilitate future maintenance, you need to regularly back up project data.
4 Production & maintenance	Configure the Production Interface	Before the vision solution is delivered to the production line for use, you can configure a set of production interfaces, which will facilitate the on-site operator to get quick knowledge of production status, view the production results, switch workpieces or add new workpiece types and easily perform maintenance and troubleshooting.
	Use the Production Interface	The on-site operator can quickly learn how to use the production interface by referring to this document.

1.2.6. Picking

The ultimate goal of deploying a 3D vision-guided application is to achieve successful picking. In this phase, you need to build a robot picking workflow so that the robot can pick objects accurately based on vision results.

Depending on the different robot communication modes selected, the way to build the workflow also varies, as shown in the figure below.

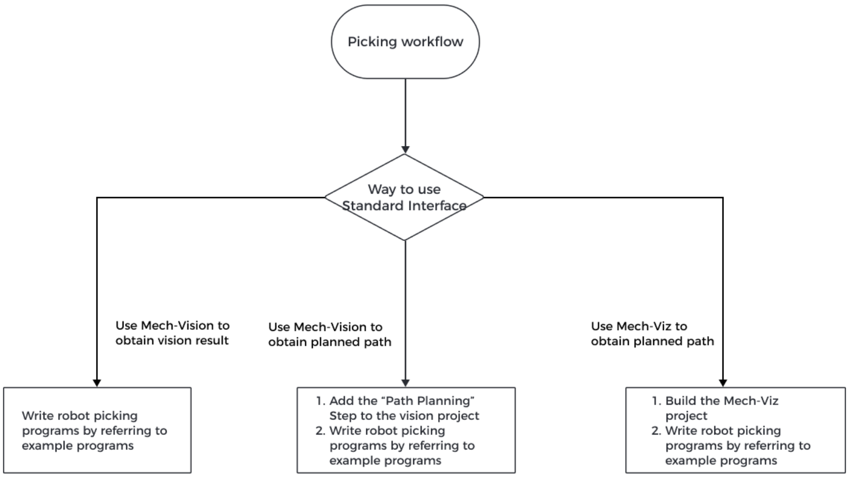

Picking
Robot communication mode
Standard Interface
Master-Control
Adapter
1. Write the robot picking programs by referring to example programs
2. Build Mech-Viz project (if required)
Build Mech-Viz project
1. Write the robot picking programs
2. Build Mech-Viz project (if required)

If the project requires a high picking accuracy, ensure a good picking accuracy of the application during deployment according to the guidance in Topic: Improving Picking Accuracy.

Build the Picking Workflow in the Standard Interface Communication Mode

If you have chosen to use the Standard Interface communication mode, you need to write a robot picking program for the robot side so that the robot can perform picking tasks under the vision guidance. Mech-Mind offers Standard Interface example programs for supported robots, and you can refer to those programs to write your robot picking program.

The Standard Interface communication mode can be used in three ways: using Mech-Vision to obtain vision results, using Mech-Vision for path planning, and using Mech-Viz for path planning. The robot picking workflow varies in different ways.


Picking workflow
Way to use Standard Interface
Use Mech-Vision to obtain vision result
Use Mech-Vision to obtain planned path
Use Mech-Viz to obtain planned path
Write robot picking programs by referring to example programs
1. Add the "Path Planning" Step to the vision project
2. Write robot picking programs by referring to example programs
1. Build the Mech-Viz project
2. Write robot picking programs by referring to example programs

Build the Picking Workflow in the Master-Control Communication Mode

If you have chosen to use the Master-Control communication mode, you can build the robot picking workflow (that is the Mech-Viz project) in a graphical manner in Mech-Viz.

1 Build a Project	Getting Started	Understand the general process of building a Mech-Viz project.
2 Model Preparation	Use Model Editor to Create Models	Please use the Model Editor to create the tool, object, and scene models for the Mech-Viz project, which will be used in 3D simulation and collision detection.
3 Configuration of the Project Resources	Object Configuration	Import your object model and configure the object. You can configure attributes such as Workobject symmetry and Picking relaxation to improve the success rate of object picking.
	Tool Configuration	Import the tool model and configure the tool. You need to adjust the position and size of the tool model as well as the control logic of the tool.
	Configure the Scene Object	Configure the objects in real scenarios to make the scene in the software closer to the real scenario, which facilitates collision detection and path planning.

4 Simulation and Improvement	Collision Detection Configuration	Configure the collision detection to prevent the robot from colliding during its movement so that a planned path without collision will be output for the robot.
	Use Plan History to Facilitate Improvement	Plan history records the whole process of path planning from Mech-Viz and displays the collision results in a visualized way. You can refer to the fixes for planning failures to troubleshoot.

2. Getting Started Tutorial: Vision-Guided Robotic Picking of Metal Parts (Master-Control Communication)

In this tutorial, you will learn how to deploy a simple 3D vision-guided robotic application of picking small metal parts in the Master-Control communication mode.

Application Overview

- Camera: Mech-Eye PRO M camera, mounted in eye to hand (ETH) mode
- Robot: ABB_IRB_1300_11_0_9
- Workpiece: track links (made of metal)

  - For this application, you are required to prepare a model file in CAD format for the workpiece, which will be used to generate the point cloud matching model. Click here to download the model file.
  - This application uses a real camera to capture images of the track links for recognition. If you want to use a virtual camera, please click here to download image data of the track links.

- End tool: gripper

  For this application, you are required to prepare a model file in OBJ format for the gripper, which will be used for collision detection during path planning. Click here to download the model file.

- Scene object: scene object model

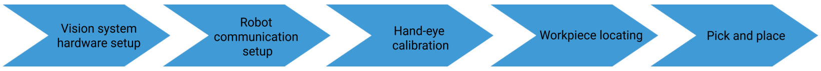
  This application requires a scene model file in STL format, which is used to simulate a real scene and is used for collision detection in path planning. Click here to download the model file.

- Used software: Mech-Vision 1.7.4, Mech-Viz 1.7.4, Mech-Center 1.7.4, and Mech-Eye Viewer 2.1.0
- Communication mode: Master-Control communication

If you are using a different camera model, robot brand, or workpiece than in this example, please refer to the reference information provided in the corresponding steps to make adjustments.

How to Deploy a Vision-Guided Robotic Application?

The deployment of the vision-guided robotic application can be divided into five phases, as shown in the figure below:



The following table describes the five phases of deploying a vision-guided robotic application.

No.	Phase	Description
1	Vision system hardware setup	Install and connect hardware of the Mech-Mind Vision System.
2	Robot communication setup	Load the robot master-control program and the configuration files to the robot system and set up the communication between the vision system and the robot, thus helping the Mech-Mind Vision System obtain control over the robot.
3	Hand-eye calibration	Perform the automatic hand-eye calibration in the eye-to-hand setup, to establish the transformation relationship between the camera reference frame and the robot reference frame.
4	Workpiece locating	Use the “General Workpiece Recognition” case project to calculate the workpiece poses and output the vision result.
5	Pick and place	Use Mech-Viz to create a workflow that can guide the robot to repeatedly pick and place workpieces.

Next, follow subsequent sections to complete the application deployment.

2.1. Vision System Hardware Setup

In this tutorial, you will learn how to build the Mech-Mind Vision System.

Follow the steps below to build the Mech-Mind Vision System: **Check the contents of the package** → **Install the hardware** → **Connect the network** → **Upgrade the software (optional)** → **Confirm that the vision system can capture images normally**.

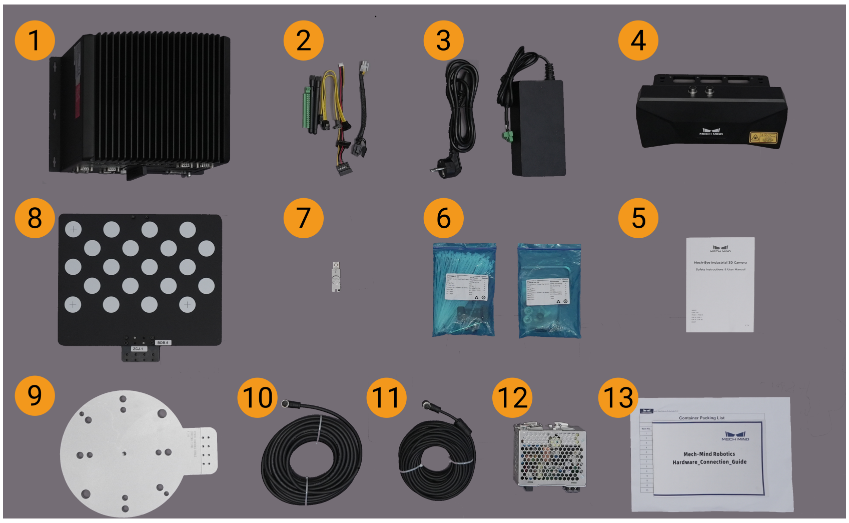
▶ https://www.youtube.com/watch?v=Utnjo0l0UAU/PLVcMd7cW2rXVtrAejMyVQni2dUDv8bxje *(YouTube video)*

Video Tutorial: Vision System Hardware Setup

Check Contents of the Package

1. Make sure that the package is intact when you receive it.
2. Check the contents against the “packing list” in the package to ensure that no devices or accessories are missing or damaged.

The following figure shows the devices and accessories included in a vision system shipment. The table below is for reference only. Please take the “packing list” in the package as final.



No.	Category	Name	Function
1	IPC and accessories	IPC	Provides running environment for Mech-Mind software
2		IPC accessories	Provides IPC accessories, such as antennas for WIFI connection
3		IPC power cable and adapter	Supplies power to IPC
4	Camera and accessories	Mech-Eye Industrial 3D Camera	Captures images
5		User manual	Mech-Eye Industrial 3D Camera User Manuals
6		Camera accessory box	Mounts the camera
7	Project accessories	License dongle	Authorizes the Software Suite license
8		Calibration board	Calibrates the camera
9		Flange adapter	Connects the calibration board to the robot flange
10		Camera DC power cable (standard: 20 meters)	Connects the camera and the DIN rail power supply; Longer power cables are an option
11		Camera Ethernet cable (standard: 20 meters)	Connects the camera and the IPC; longer camera Ethernet cables are an option
12		DIN rail power supply (standard)	Supplies power to Mech-Eye Industrial 3D Camera; The camera power adaptor is an option
13	Packing list		Lists all the devices and accessories in the package

Contact Mech-Mind if any items are missing or damaged.

Prepare Other Materials

In this tutorial, besides the items in the package, you still need to prepare the materials shown in the following table by yourself.

Item	Function
Monitor	Provides display for the IPC
HDMI cable	Connects the monitor and the IPC
RJ45 Ethernet cable	Connects the IPC and the robot controller

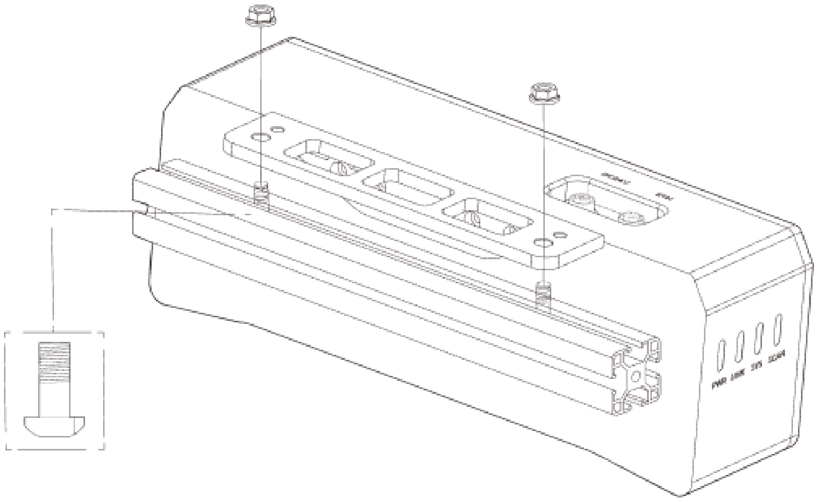
In this tutorial, the IPC and robot controller are directly connected through an RJ45-to-RJ45 Ethernet cable, and the IPC and the camera are directly connected through the camera Ethernet cable. Alternatively, you can use a router to connect the IPC and the robot controller, and the IPC and the camera, which is not covered in this topic.

Install Hardware

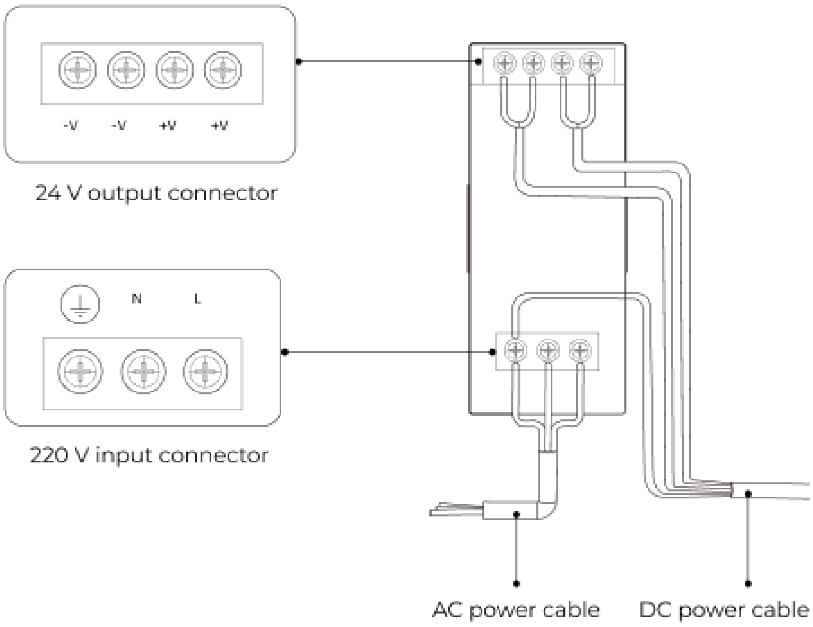
Mount the Camera

In this tutorial, the camera is mounted on the camera mounting frame (that is the eye to hand (ETH) mounting mode). In addition, the camera can also be mounted onto the end terminal of the robot (that is the eye in hand (EIH) mounting mode).

1. Get the camera mounting bolts and wrench from the camera accessory box.
2. Tighten the two bolts with the wrench as shown below.

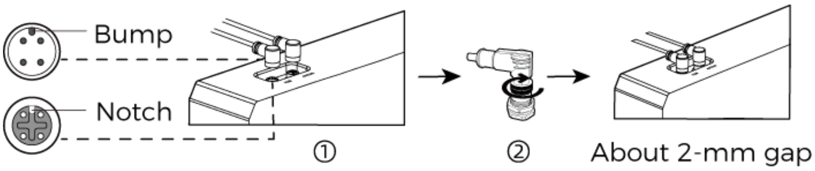


3. Please remove the lens protection film after mounting the camera.
4. Power the camera with the DIN rail power supply.
   - Connect the DC power cable:
     - Connect the +V wire to the +V connectors of the 24 V output connectors;
     - Connect the -V wire to the -V connectors of the 24 V output connectors;
     - Connect the PE wire to the 220 V input connector ⏚.



5. Install the Ethernet cable of the camera.

   Make sure the bump of the M12 connector and the notch of the ETH port, and tighten the nut after plugging in the cable.

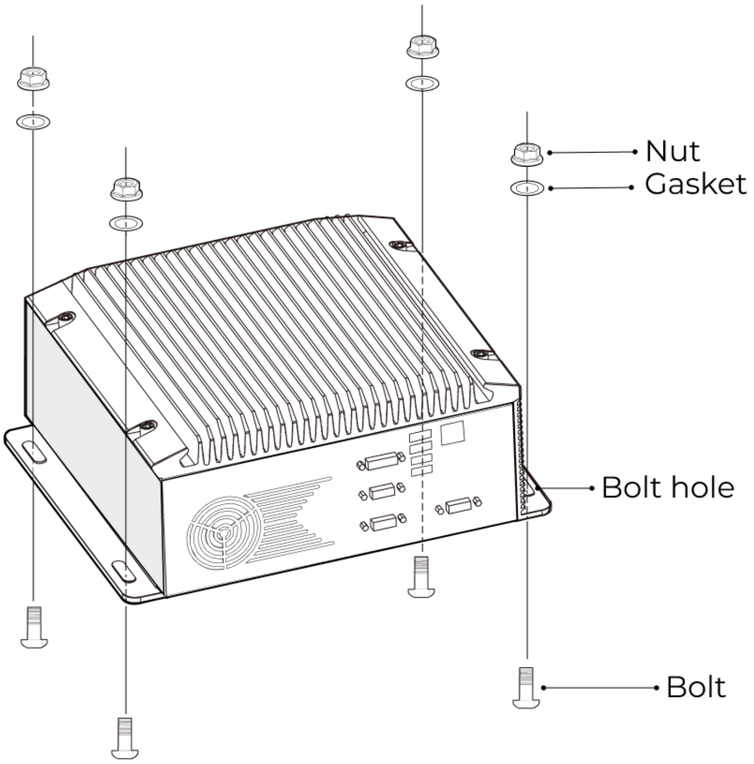


Mount the IPC

The IPC is normally mounted in the control cabinet of the robot. The environment in which the IPC is mounted requires good heat dissipation, ventilation, and dust protection. It should be mounted at a location where Ethernet cables, HDMI cable, and USB ports can be easily mounted and serviced.

To mount the IPC, follow these steps:

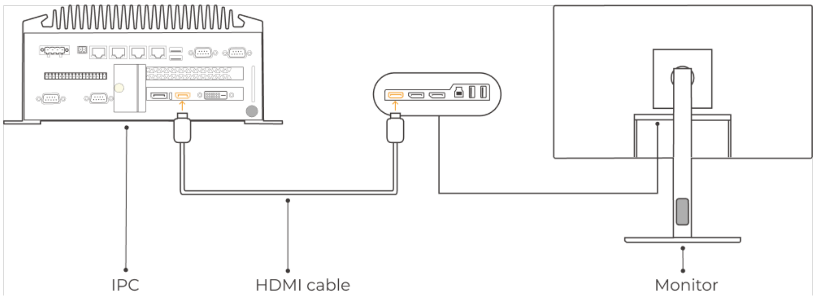
1. Prepare the wrench, bolts, nuts, and gaskets that are not included in the package beforehand.
2. If the robot controller is designed with mounting holes for the IPC in it, secure the IPC in the controller: place the bolt, gasket, and nut one by one, and tighten the two bolts with a wrench, as shown below.



If the location of the robot controller is already fixed, skip this step and just place the IPC inside the controller.

3. Connect the IPC and the monitor with the HDMI cable.

   Plug one end of the HDMI cable into the HDMI port of the monitor, and the other end into the HDMI port of the IPC, as shown below.



4. Connect the IPC to the power supply unit with the power adaptor.

   Plug the power cable of the power adaptor into the power connector of the IPC. Connect the adaptor to the power supply on the other end.

5. Insert the license dongle.

   Plug the license dongle into a USB port of the IPC.

6. After the IPC is connected to the power supply, switch on the IPC.

   - If the IPC is started normally, the power indicator should be solid on.

- If the IPC cannot be started, contact Mech-Mind Technical Support.

Connect the Network

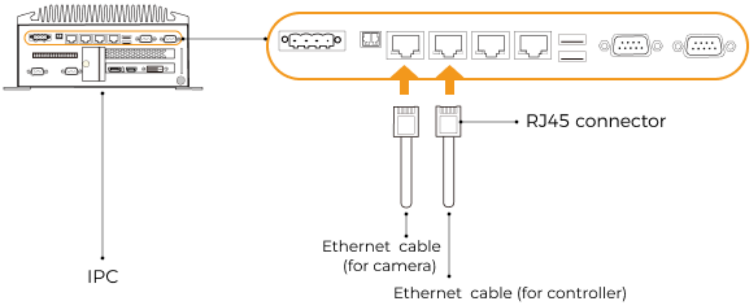
In this section, you will learn how to connect the network between the IPC and the camera, and between the IPC and the robot.

In the following sections, the following IP addresses will be used for network settings. Please adjust the network settings according to your actual network environment.

Device		IP Address
IPC	Ethernet port connecting to the camera	192.168.100.10
	Ethernet port connecting to the robot controller	192.168.200.10
Camera		192.168.100.20
Robot		192.168.200.20 (already set on the robot)

Connect the IPC and the Camera, and the IPC and the Robot Controller

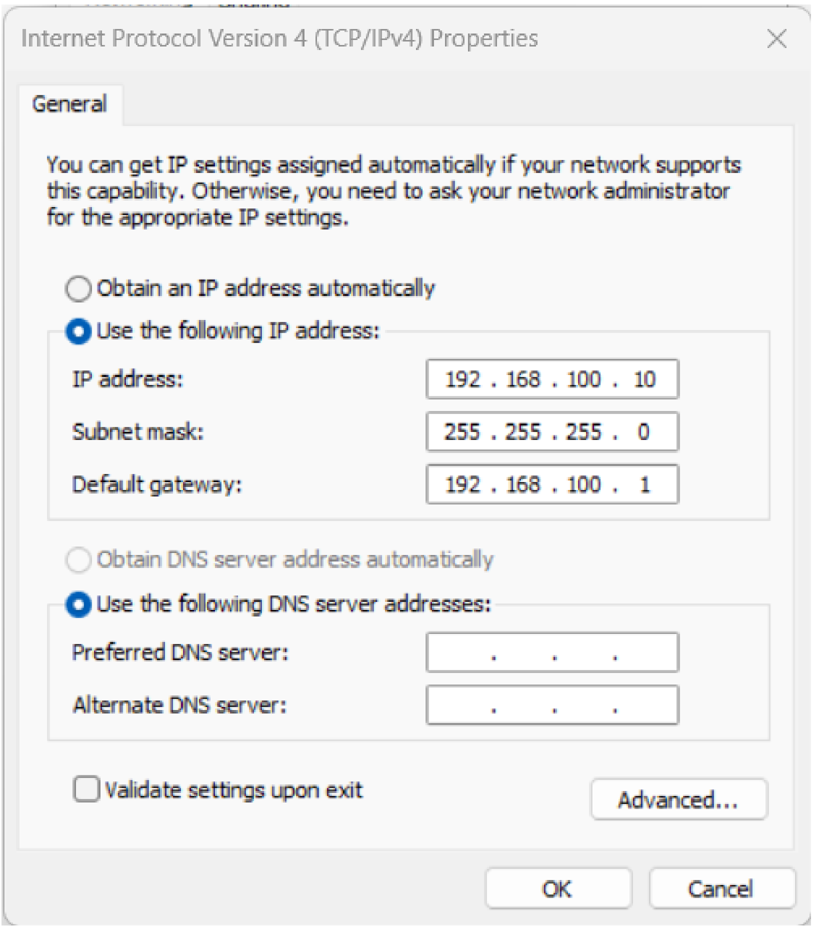
1. Plug the other end of the Ethernet cable connected to the camera into an Ethernet port of the IPC.



2. Use an RJ45-to-RJ45 Ethernet cable to plug one end into the Ethernet port of the robot controller and the other end into an Ethernet port of the IPC.

Set the IP Addresses on the IPC

1. Select Control Panel › Network and Internet › Network and Sharing Center › Change adapter settings on the IPC. The **Network Connections** page will be displayed.
2. Right-click the Ethernet port connected to the camera, and select **Rename** to rename the Ethernet port, such as "To_camera".
3. Right-click the Ethernet port connected to the camera, and select **Properties** to enter the **Ethernet Properties** page.
4. Select the **Internet Protocol Version 4 (TCP/IPv4)** checkbox, and then click the **[Property]** button to enter the **Internet Protocol Version 4 (TCP/IPv4) Properties** page.
5. Select the **Use the following IP address** radio button, set the **IP address** parameter to "192.168.100.10", **Subnet mask** to "255.255.255.0", and **Default gateway** to "192.168.100.1", and then click the **[OK]** button.



Internet Protocol Version 4 (TCP/IPv4) Properties

General

You can get IP settings assigned automatically if your network supports this capability. Otherwise, you need to ask your network administrator for the appropriate IP settings.

○ Obtain an IP address automatically

● Use the following IP address:

IP address: 192 . 168 . 100 . 10

Subnet mask: 255 . 255 . 255 . 0

Default gateway: 192 . 168 . 100 . 1

○ Obtain DNS server address automatically

● Use the following DNS server addresses:

Preferred DNS server: . . .

Alternate DNS server: . . .

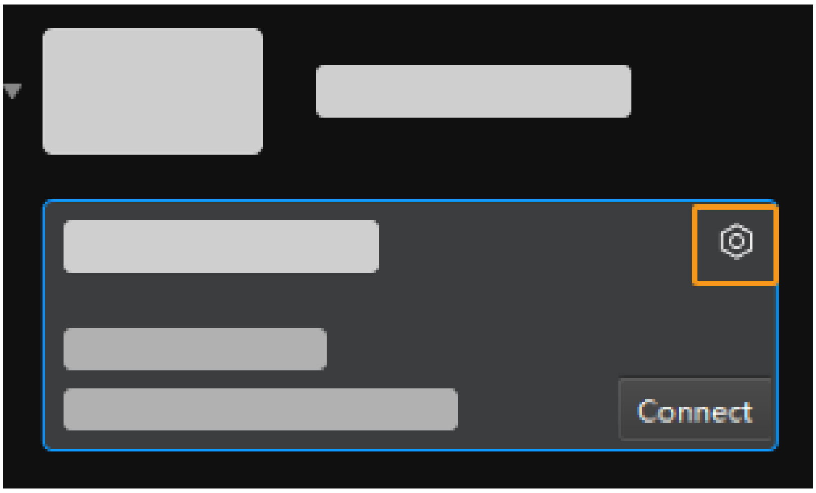
□ Validate settings upon exit

6. Repeat steps 2 to 5 to rename the Ethernet port connected to the robot controller (for example, “To_robot”), and set the IP address for this Ethernet port. For example, set the IP address of this Ethernet port to “192.168.200.10”.

> The IP address of the robot and that of the IPC Ethernet port connected to the robot controller must be on the same subnet.

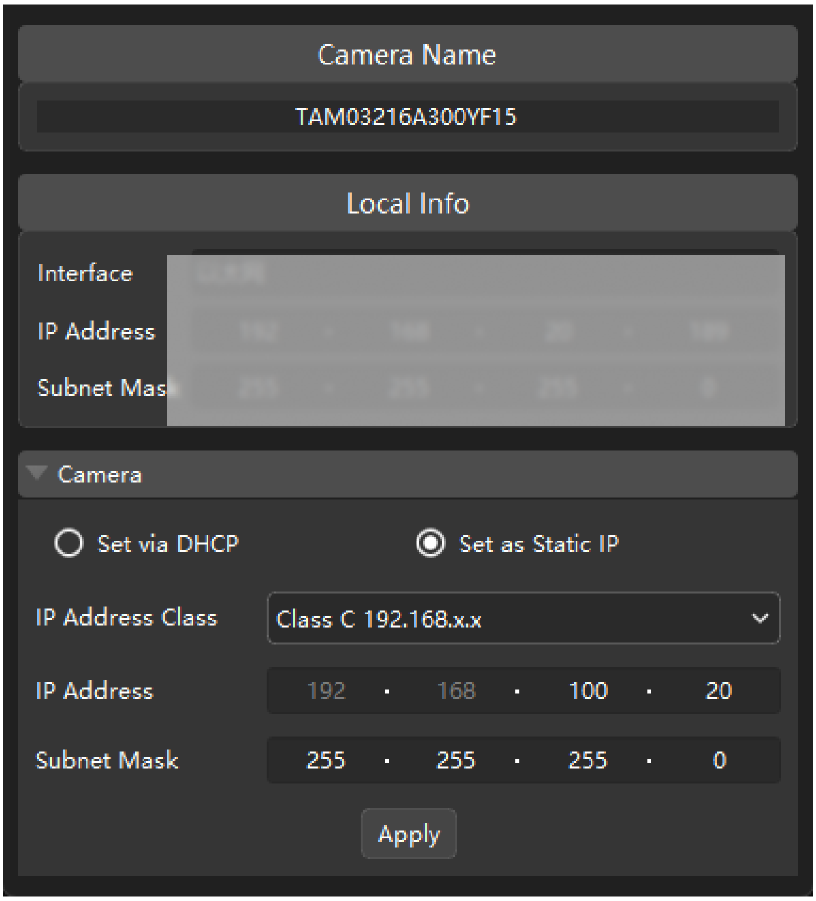
Set the Camera IP Address

1. Double-click EYE on the desktop of the IPC to open and run Mech-Eye Viewer.
2. Select the camera in the camera list, and hover the cursor on the camera. Click ⚙ to open the **Config Camera Network** dialog box.



If the camera cannot be found or connected, please refer to Camera Troubleshooting.

3. In the **Camera** area, select the **Set as Static IP** radio button, set **IP Address Class** to “Class C 192.168.x.x”, **IP Address** to **192.168.100.20**, and **Subnet Mask** to “255.255.255.0” and then click **[Apply]**.



Camera Name

TAM03216A300YF15

Local Info

Interface [redacted]

IP Address

Subnet Mas

Camera

○ Set via DHCP ● Set as Static IP

IP Address Class: Class C 192.168.x.x

IP Address: 192 · 168 · 100 · 20

Subnet Mask: 255 · 255 · 255 · 0

The IP address of the camera and that of the IPC Ethernet port connected to the camera must be on the same subnet.

Test the Network Connectivity

1. Press `Win` + `R` to open the **Run** dialog box.

2. Type cmd in the **Run** dialog box, and then click **[OK]**.
3. Type **ping XXX.XXX.XX.XX** in the command prompt window and press **[Enter]** to execute the command.

> Replace XXX.XXX.XX.XX with the actual IP address of the camera/robot.

If the network connectivity is normal, you should receive the following response:

```
Pinging XXX.XXX.XX.XX with 32 bytes of data:
Reply from XXX.XXX.XX.XX: bytes=32 time<1ms TTL=128
Reply from XXX.XXX.XX.XX: bytes=32 time<1ms TTL=128
Reply from XXX.XXX.XX.XX: bytes=32 time<1ms TTL=128
Reply from XXX.XXX.XX.XX: bytes=32 time<1ms TTL=128
```

Upgrade the Software (Optional)

The IPC purchased from Mech-Mind already has the latest version of Mech-Mind software installed.

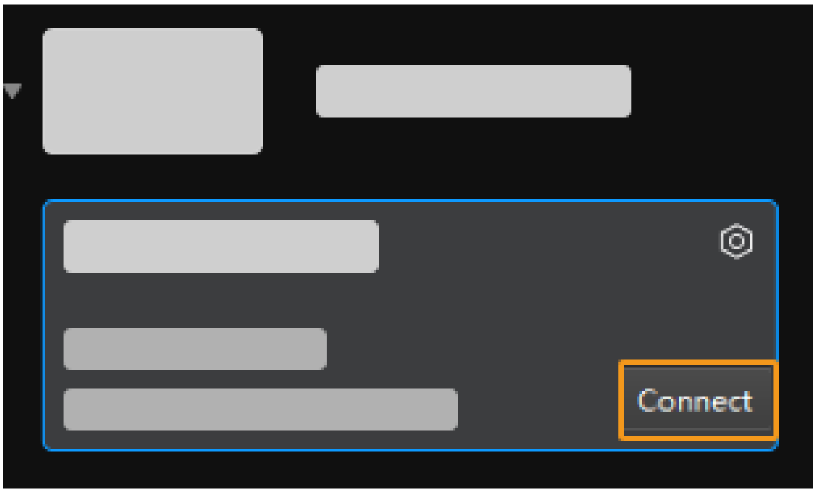
Please check if all software on the IPC is running with the latest version. If so, skip this section; if not, follow the sections below to update the software to the latest versions.

- Downloading and installing Mech-Eye SDK
- Downloading and installing Mech-Vision

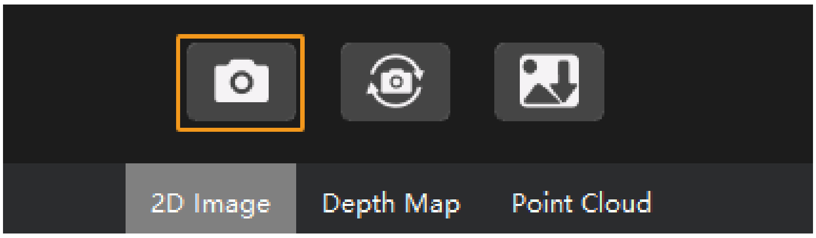
Confirm That the Vision System Can Capture Images Normally

After testing the network connectivity between the IPC and the camera/robot, please confirm that the vision system can capture images normally:

1. Place workpieces at the center of the camera's field of view (FOV), and make sure that the workpieces on the edge and top are all within the FOV.
2. Open and start Mech-Eye Viewer by double-clicking EYE on the desktop of the IPC.
3. Select the camera in the camera list and click **[Connect]**.

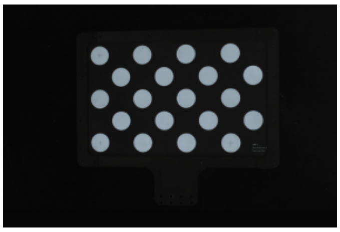
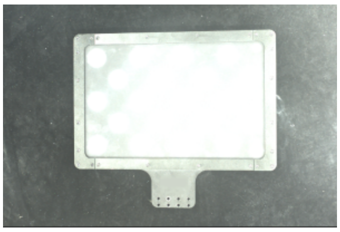
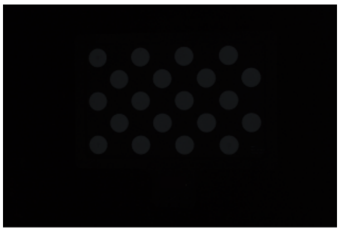
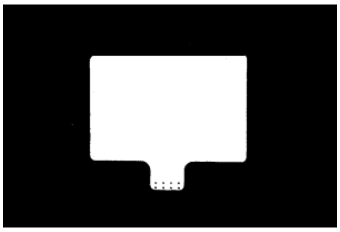
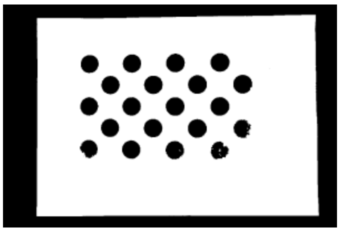
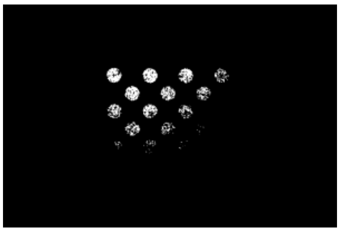


4. After the camera is connected, click **[Capture once]**.



5. Make sure that the captured images meet the following standard.

   - 2D image: the captured 2D image is not significantly overexposed (too white to recognize objects) or underexposed (too dark to recognize object details).
   - Depth map and point cloud: All relevant parts of the object can be recognized.

	Normal	Overexposed	Underexposed
2D image			
Point cloud			

If the captured images do not meet the required standard, please use Mech-Eye Viewer to adjust camera parameters.

Up to now, you have learned how to build the vision system.

2.2. Robot Communication Setup

In this tutorial, you will learn how to load the Master-Control program files to the ABB robot and configure the Master-Control communication.

- In this section, you will load the robot Master-Control program and the configuration files to the robot system to establish the communication between the vision system and the robot, thus helping the Mech-Mind Vision System obtain control over the robot.
- If you are using the robots of other brands, you can find instructions on setting up the Master-Control communication configuration with the desired robot from the section Master-Control Communication.

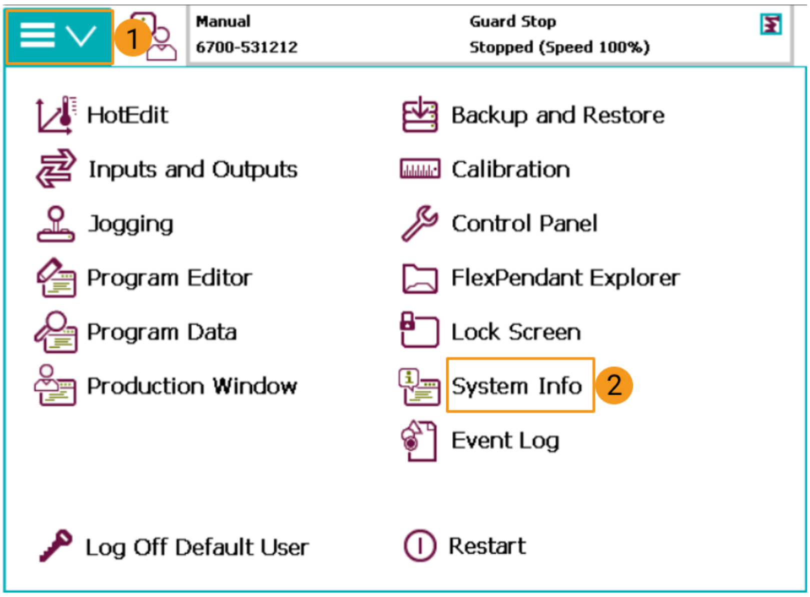
▶ https://www.youtube.com/watch?v=jyZYv4cK090/PLVcMd7cW2rXVtrAejMyVQni2dUDv8bxje *(YouTube video)*

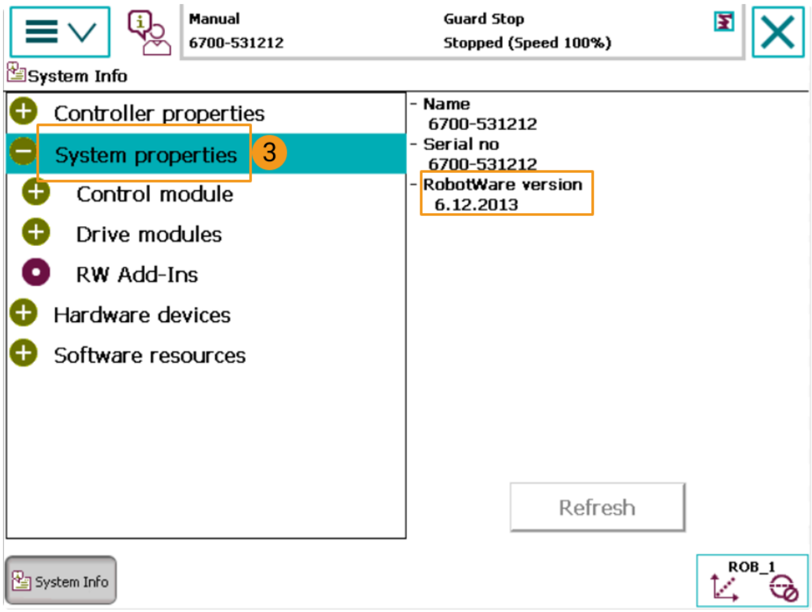
Video Tutorial: Robot Interface Configuration

Preparation

Check the Controller and RobotWare Version

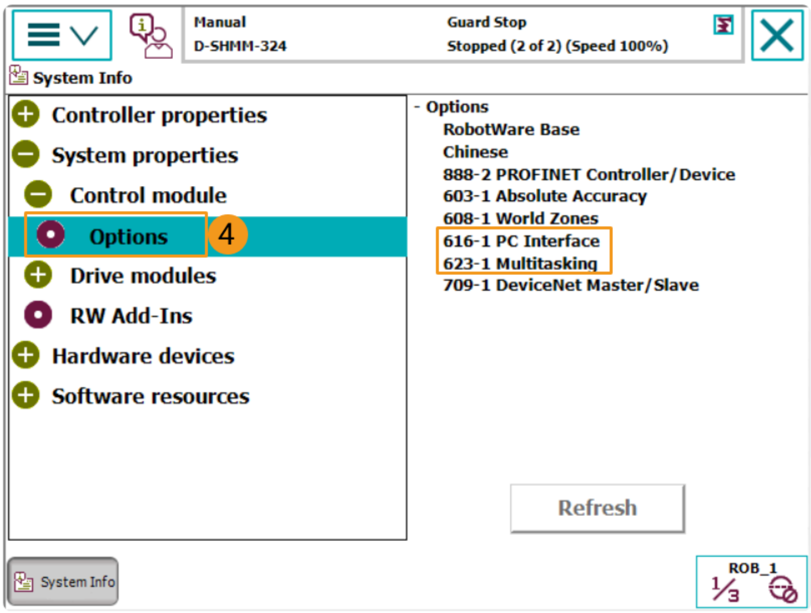
1. Make sure that D652 or DSQC1030 IO board has been installed on the robot controller.
2. Confirm that the RobotWare version is 6.0 or above on the teach pendant.





- Name
  6700-531212
- Serial no
  6700-531212
- RobotWare version
  6.12.2013

3. Confirm that the following control modules have been installed on the teach pendant.
   - 623-1 Multitasking
   - 616-1 PCInterface



- Options
  - RobotWare Base
  - Chinese
  - 888-2 PROFINET Controller/Device
  - 603-1 Absolute Accuracy
  - 608-1 World Zones
  - 616-1 PC Interface
  - 623-1 Multitasking
  - 709-1 DeviceNet Master/Slave

> If the preceding conditions cannot be met, the vision side cannot communicate with the robot through the master-control mode. Please contact the vendor of your robot.

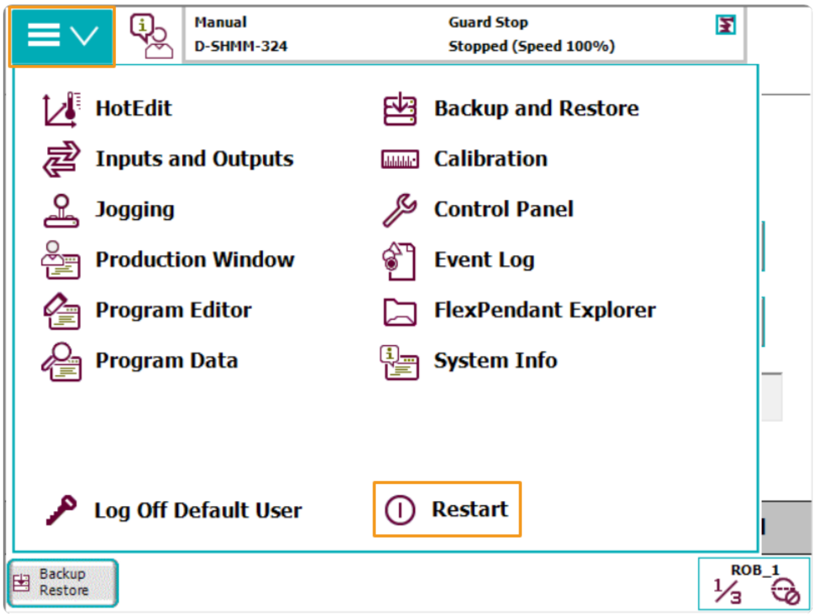
Reset the Robot System

After checking the software and hardware status of the controller, reset the robot system. If you are using a new robot, skip this section.

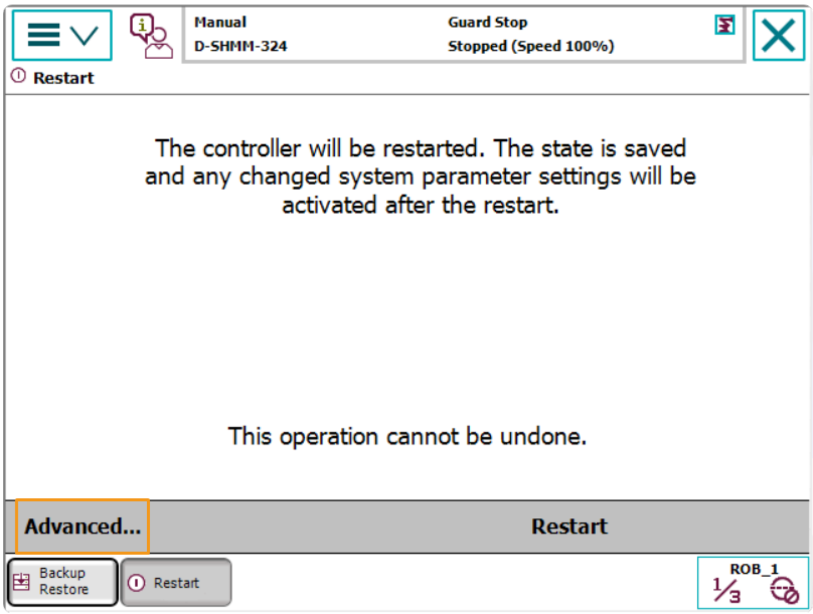
> Resetting the system will restore the factory settings. Please make sure that you have completed the backup operation.

To reset the robot system, follow these steps:

1. On the teach pendant, go to the home page using the menu in the upper-left corner, and then press [Restart].



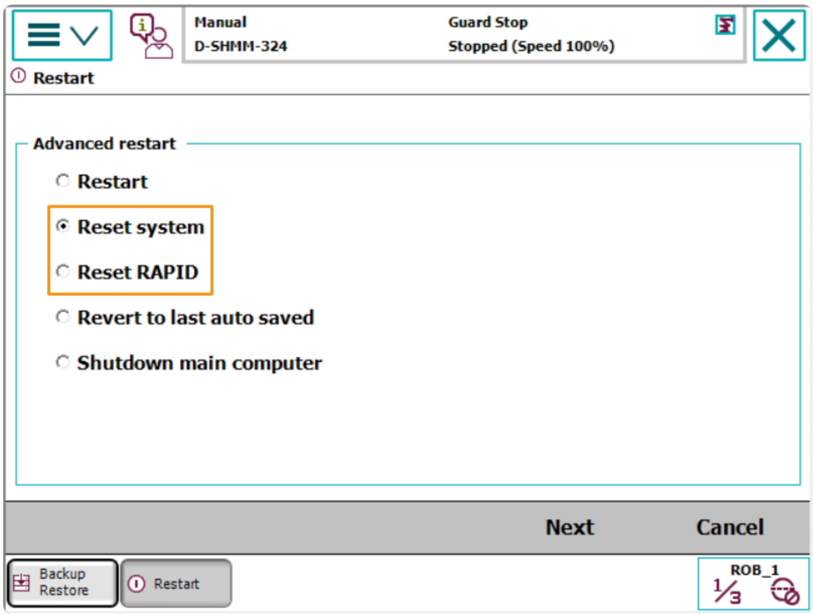
2. Press [Advanced…].



The controller will be restarted. The state is saved and any changed system parameter settings will be activated after the restart.

This operation cannot be undone.

3. Select Reset system and then press [Next].



Advanced restart

○ Restart

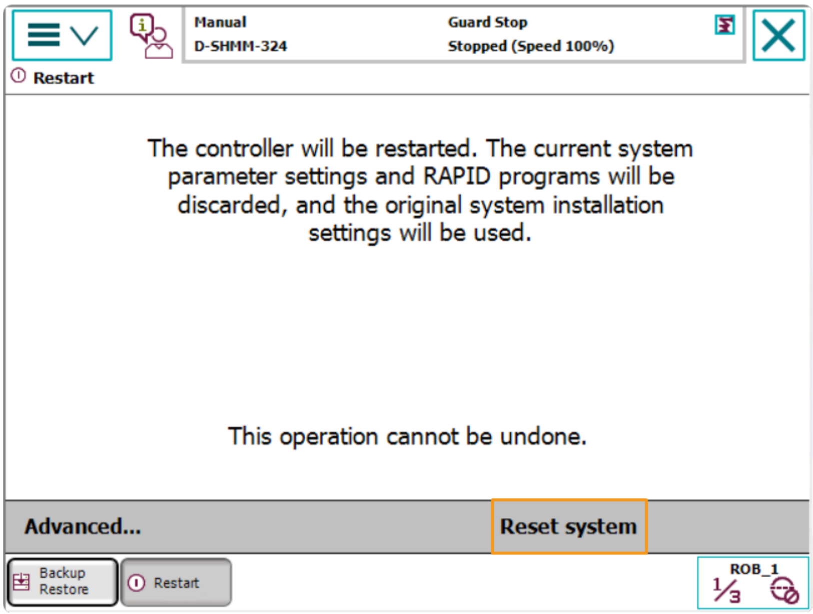
● Reset system

○ Reset RAPID

○ Revert to last auto saved

○ Shutdown main computer

4. Press [Reset system].



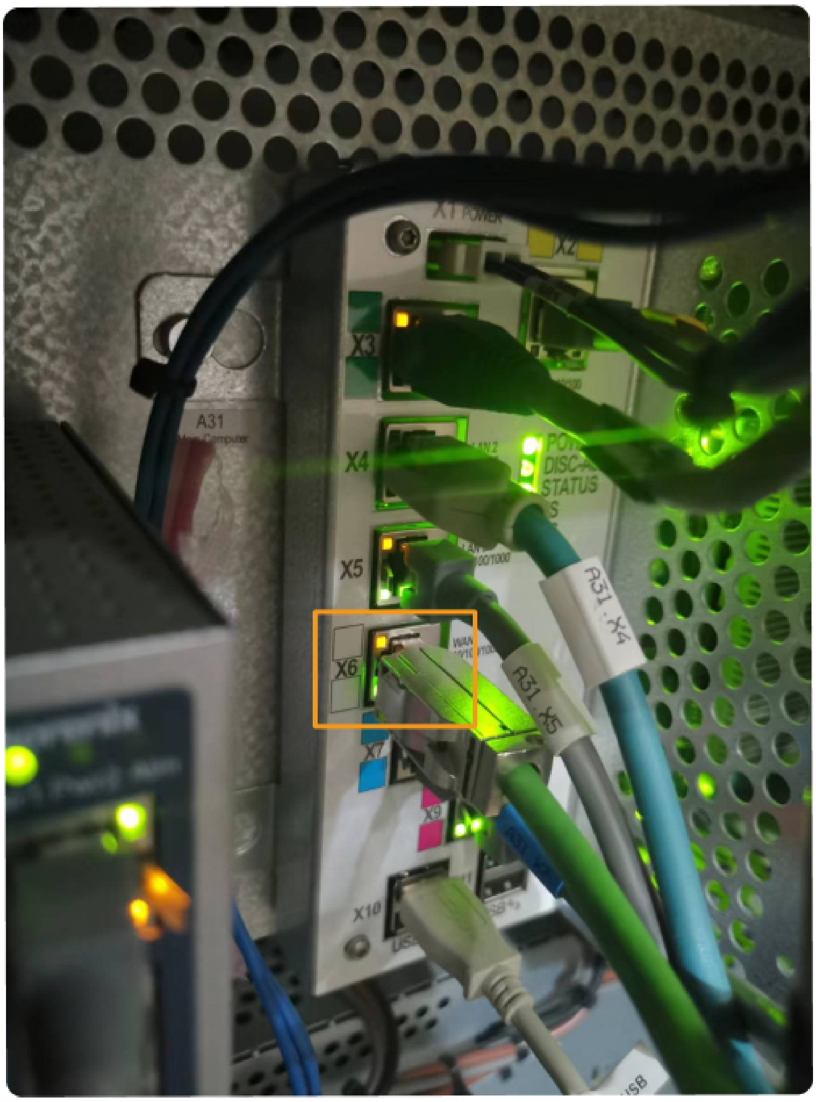
The controller will be restarted. The current system parameter settings and RAPID programs will be discarded, and the original system installation settings will be used.

This operation cannot be undone.

Resetting the system takes 1 to 2 minutes. Resetting is completed when the home page is displayed again on the teach pendant.

Connect the Network

1. Connect the IPC Ethernet Cable to the **X6 (WAN)** port on the robot controller, as shown below.



2. Make sure that the IP address of the ABB robot and that of the IPC are in the same subnet.

Prepare the Program Files

1. On the IPC, open the `Mech-Center/Robot_Server/Robot_FullControl/abb/server on ABB` folder in the installation directory of the Mech-Vision and Mech-Viz software.
2. Copy this folder to the USB flash drive, and insert it into the computer installing the RobotStudio software.

> RobotStudio is the simulation and offline programming software for ABB robots. This software can be installed on the IPC or another computer. In this example, RobotStudio is installed on another computer.

File description:

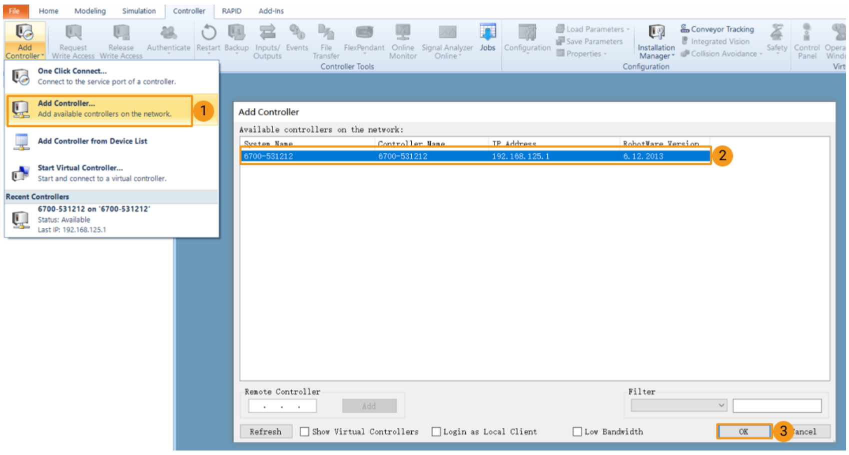
- “MM” folder: includes the robot program modules.
- “config”: robot configuration files.
  - If the D652 IO board is used on site, you should use the D652.cfg and SYS.cfg files.
  - If the DSQC1030 IO board is used on site, you should use the DSQC1030.cfg and SYS.cfg files.

- If neither the D652 IO board nor the DSQC1030 IO board is used on site, you should use the EIO.cfg and SYS.cfg files.

Load the Program Files to the Robot

Open RobotStudio and Connect to the Controller

Click the **Controller** menu of RobotStudio, and select Add Controller › Add Controller on the toolbar. In the prompted **Add Controller** dialog box, select the controller, and click the [OK] button.



Add Controller

Available controllers on the network:

System Name	Controller Name	IP Address	RobotWare Version
6700-531212	6700-531212	192.168.125.1	6.12.2013

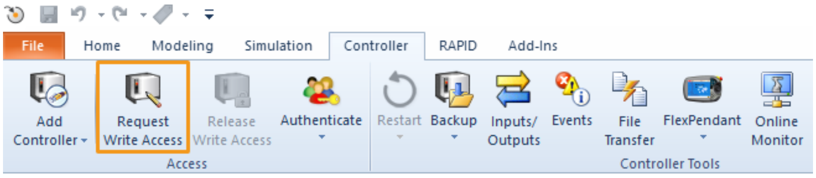
Remote Controller

Filter

□ Show Virtual Controllers □ Login as Local Client □ Low Bandwidth

Obtain Write Access to the Robot

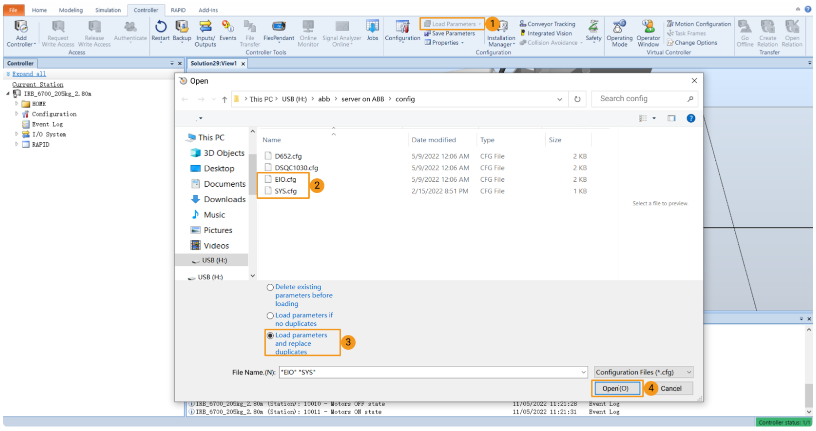
1. On the toolbar, click **Request Write Access** to request the write access to the teach pendant.



2. In the prompted **Request for Write Access** dialog box on the teach pendant, tap [**Grant**].

Load the Robot Configuration Files

1. Click the **Controller** menu of RobotStudio, and select Load Parameters on the toolbar. Select the configuration files to import from the USB flash drive, select the **Load parameters and replace duplicates** radio button, and then click the [**Open**] button.



Open

This PC › USB (H:) › abb › server on ABB › config

Name	Date modified	Type	Size
D652.cfg	5/9/2022 12:06 AM	CFG File	2 KB
DSQC1030.cfg	5/9/2022 12:06 AM	CFG File	2 KB
EIO.cfg	5/9/2022 12:06 AM	CFG File	2 KB
SYS.cfg	2/15/2022 8:51 PM	CFG File	1 KB

○ Delete existing parameters before loading

○ Load parameters if no duplicates

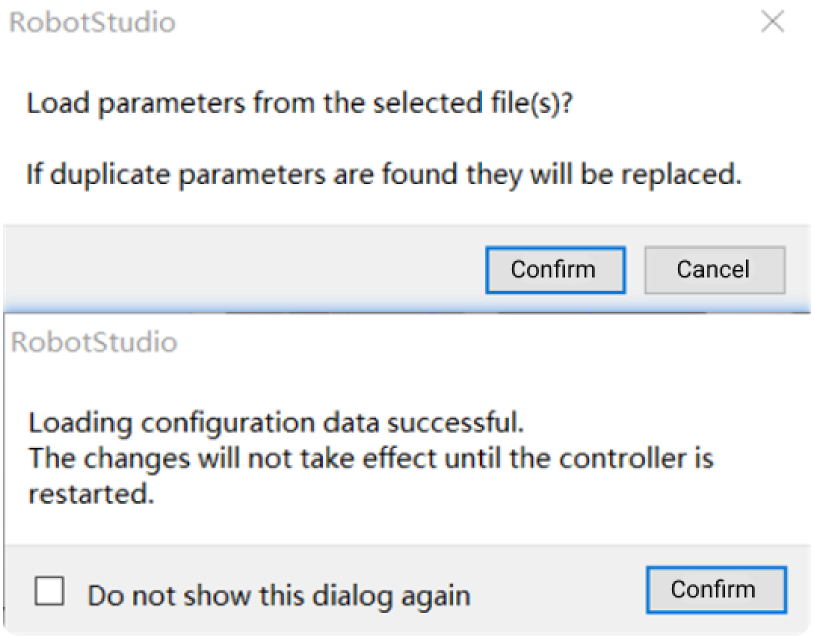
● Load parameters and replace duplicates

File Name.(N): "EIO" "SYS"

Configuration Files (*.cfg)

```
IRB_6700_205kg_2.80m (Station): 10010 - Motors OFF state    11/05/2022 11:21:28   Event Log
IRB_6700_205kg_2.80m (Station): 10011 - Motors ON state     11/05/2022 11:21:31   Event Log
```

2. In the prompted dialog boxes, click [Confirm].



RobotStudio

Load parameters from the selected file(s)?

If duplicate parameters are found they will be replaced.

Confirm Cancel

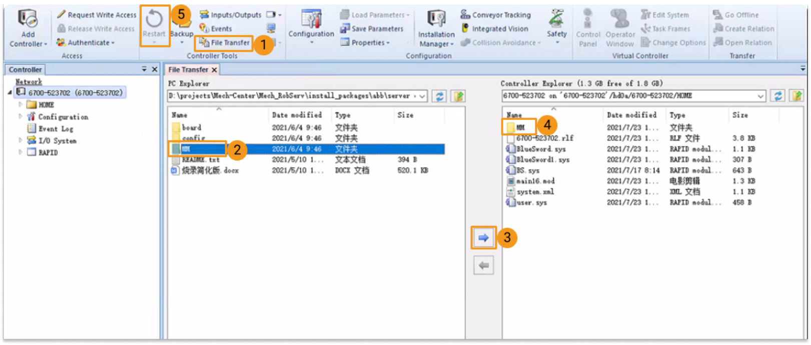
RobotStudio

Loading configuration data successful.
The changes will not take effect until the controller is restarted.

□ Do not show this dialog again Confirm

Load the Robot Program Modules

1. Click the **Controller** menu of RobotStudio, and select **File Transfer** on the toolbar. On the left panel of the **File Transfer** interface, select the "MM" folder, and then click the **Transfer** button to transfer this folder to the HOME directory of the robot system.



2. Click the **Controller** menu of RobotStudio, and select **Restart** on the toolbar to restart the robot system.

Till now, you have loaded the master-control program and the configuration files to the robot.

Test Whether Master-Control Communication Can Be Established

After the robot system has been rebooted, perform the following steps to test whether the master-control communication can be established with the robot:

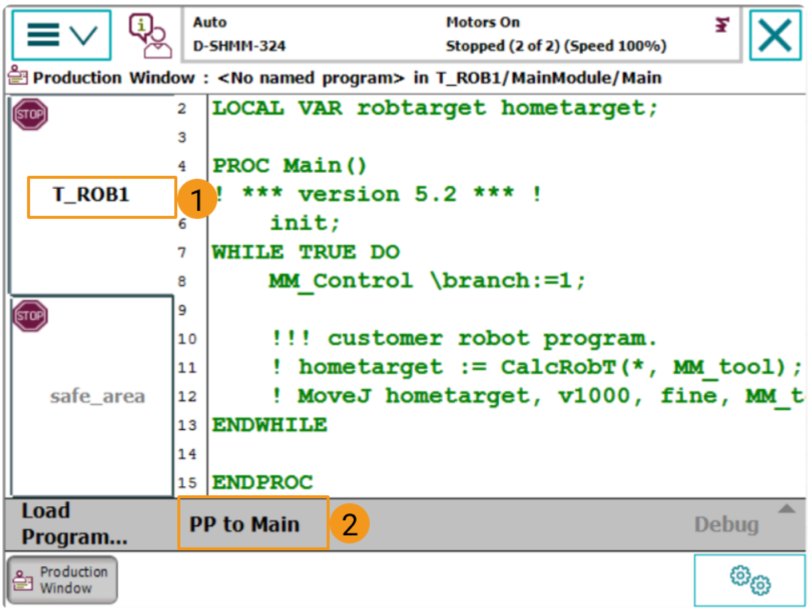
Switch the Robot to Automatic Mode

1. In the robot controller, switch the robot to the automatic mode by using the switch key.
2. In the prompted dialog box on the teach pendant, tap [Confirm].
3. In the robot controller, press the motor power button to power up the robot. When the robot is

powered on, this button is solid on.

Run the Main Program

1. On the teach pendant, move the PP of tasks T_ROB1 to main, and tap [PP to Main].

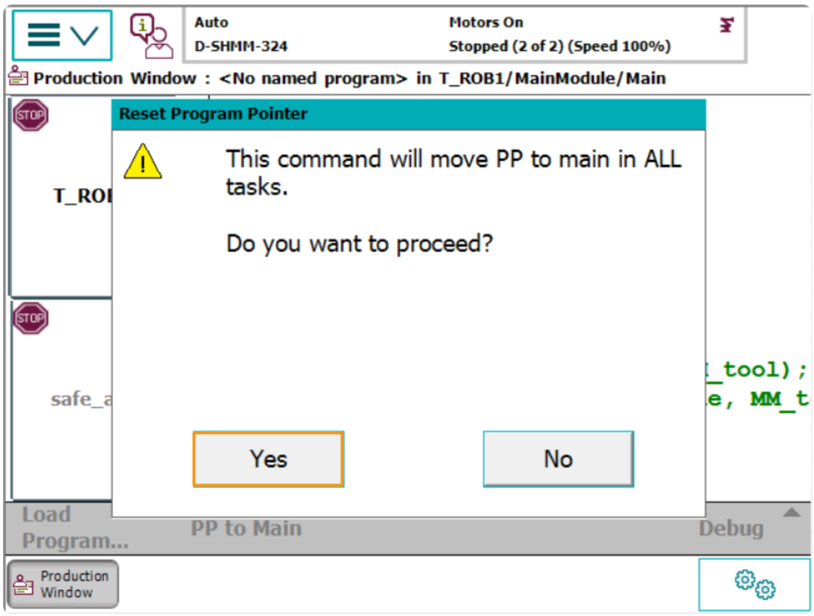


Auto
D-SHMM-324
Motors On
Stopped (2 of 2) (Speed 100%)

Production Window : <No named program> in T_ROB1/MainModule/Main

```
LOCAL VAR robtarget hometarget;

PROC Main()
! *** version 5.2 *** !
    init;
WHILE TRUE DO
    MM_Control \branch:=1;

    !!! customer robot program.
    ! hometarget := CalcRobT(*, MM_tool);
    ! MoveJ hometarget, v1000, fine, MM_t
ENDWHILE

ENDPROC
```

2. In the prompted dialog box, click [Yes].



Auto
D-SHMM-324
Motors On
Stopped (2 of 2) (Speed 100%)

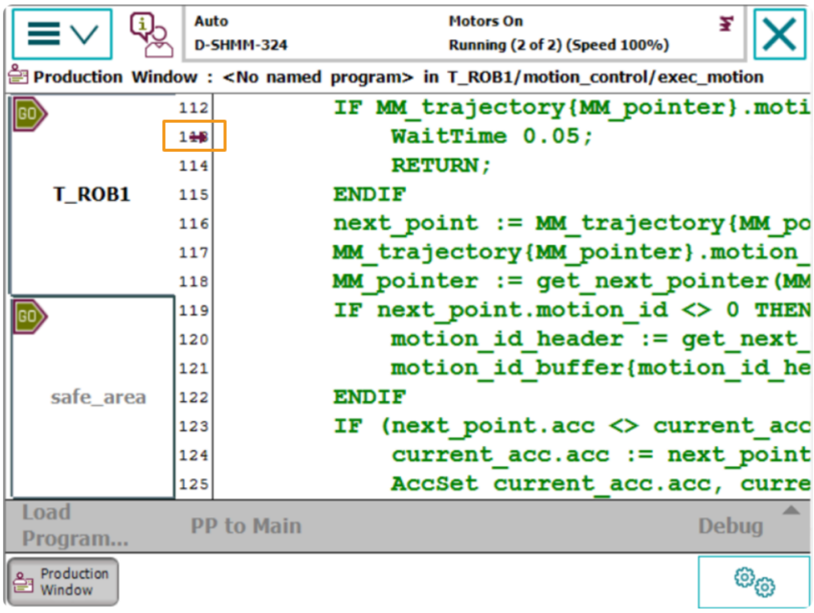
Production Window : <No named program> in T_ROB1/MainModule/Main

Reset Program Pointer

This command will move PP to main in ALL tasks.

Do you want to proceed?

3. Press the run button on the right of the teach pendant.



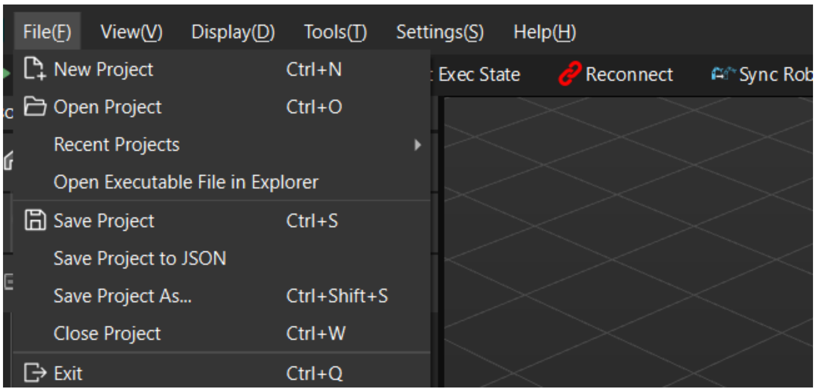
Auto
D-SHMM-324
Motors On
Running (2 of 2) (Speed 100%)

Production Window : <No named program> in T_ROB1/motion_control/exec_motion

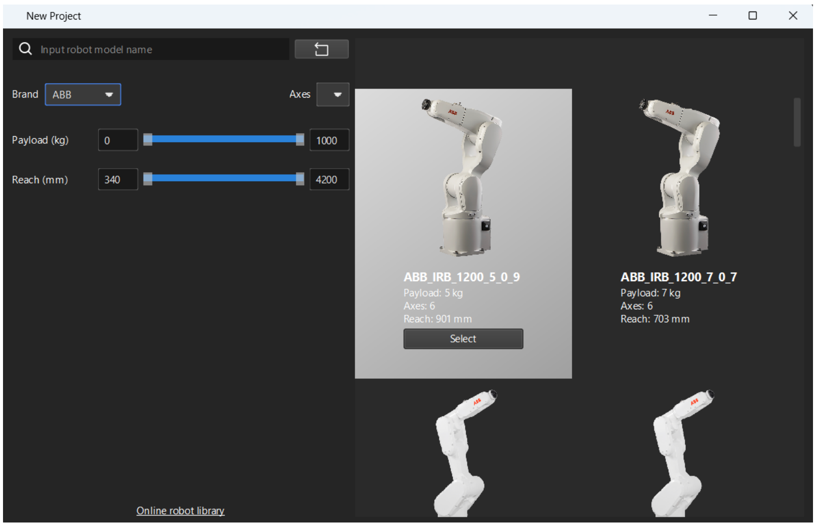
```
IF MM_trajectory{MM_pointer}.moti
    WaitTime 0.05;
    RETURN;
ENDIF
next_point := MM_trajectory{MM_po
MM_trajectory{MM_pointer}.motion_
MM_pointer := get_next_pointer(MM
IF next_point.motion_id <> 0 THEN
    motion_id_header := get_next_
    motion_id_buffer{motion_id_he
ENDIF
IF (next_point.acc <> current_acc
    current_acc.acc := next_point
    AccSet current_acc.acc, curre
```

Create a Mech-Viz Project

1. Open Mech-Viz, select File › New Project.



2. Set **Brand** to “ABB”, select the robot model “ABB_IRB_1300_11_0_9” on the right panel, and then click **[Select]**.



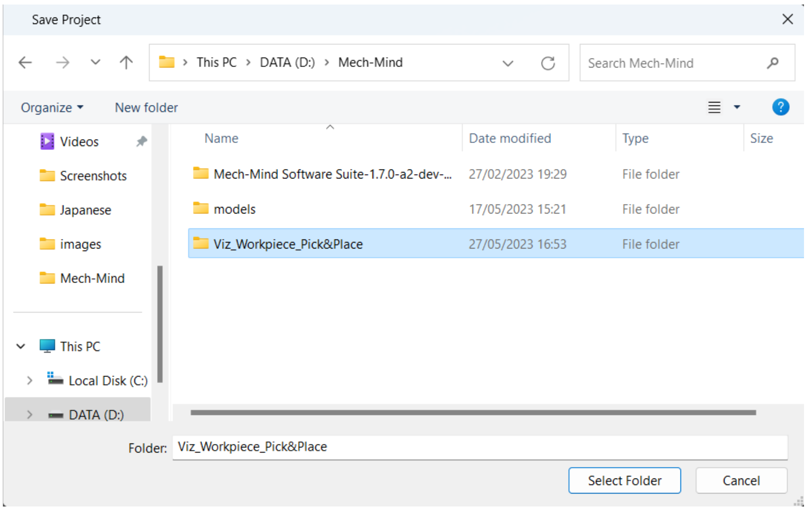
New Project

Brand: ABB
Payload (kg): 0 – 1000
Reach (mm): 340 – 4200

ABB_IRB_1200_5_0_9
Payload: 5 kg
Axes: 6
Reach: 901 mm

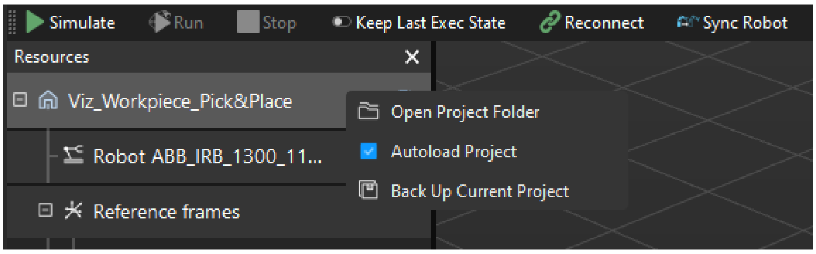
ABB_IRB_1200_7_0_7
Payload: 7 kg
Axes: 6
Reach: 703 mm

3. Press Ctrl + S, create a new folder named “Viz_Workpiece_Pick&Place”, select it, and then click [Select folder].

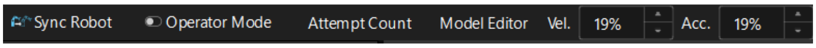

Save Project
This PC › DATA (D:) › Mech-Mind
Search Mech-Mind
Organize
New folder
Videos
Screenshots
Japanese
images
Mech-Mind
This PC
Local Disk (C:)
DATA (D:)
Name
Date modified
Type
Size
Mech-Mind Software Suite-1.7.0-a2-dev-... 27/02/2023 19:29 File folder
models 17/05/2023 15:21 File folder
Viz_Workpiece_Pick&Place 27/05/2023 16:53 File folder
Folder: Viz_Workpiece_Pick&Place
Select Folder
Cancel

After the Mech-Viz project is saved successfully, in the **Resources** panel, the project name is displayed as “Viz_Workpiece_Pick&Place”.

4. Right-click the project name, and select **Autoload project** checkbox.



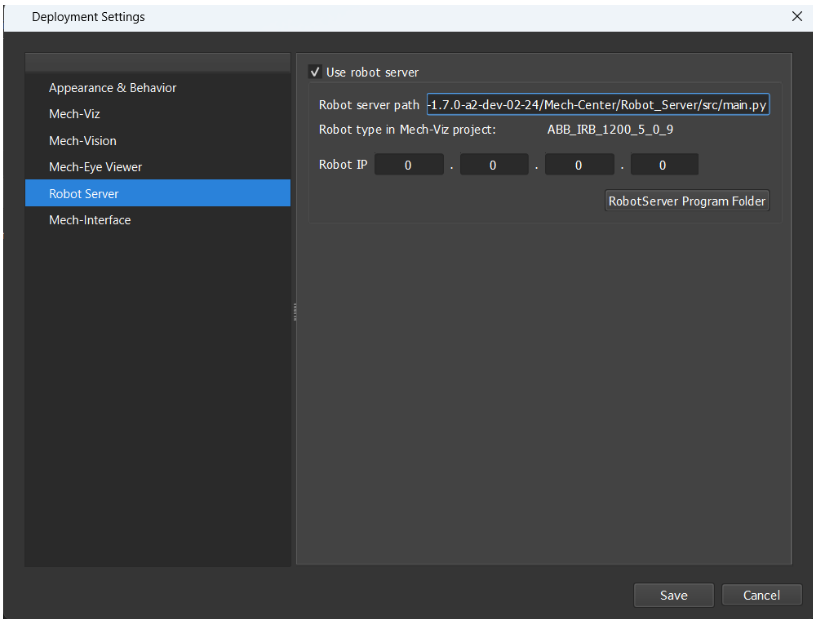
1. For the robot motion safety, on the toolbar, set **Vel.** (velocity) and **Acc.** (acceleration) to a small value, such as 5%.



2. Press Ctrl + S to save the project.

Configure Master-Control Communication Settings on Mech-Center

1. Open Mech-Center, select Deployment Settings › Robot Server, and then select **Enable Robot Server** checkbox.
2. Make sure that the robot model matches the real robot, set **Robot IP** to the IP address of the real robot, and then click [Save].


Deployment Settings

☑ Use robot server

Robot server path -1.7.0-a2-dev-02-24/Mech-Center/Robot_Server/src/main.py

Robot type in Mech-Viz project: ABB_IRB_1200_5_0_9

Robot IP 0 . 0 . 0 . 0

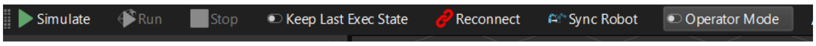
Connect to the Robot

In Mech-Center, click **Connect Robot** on the toolbar.

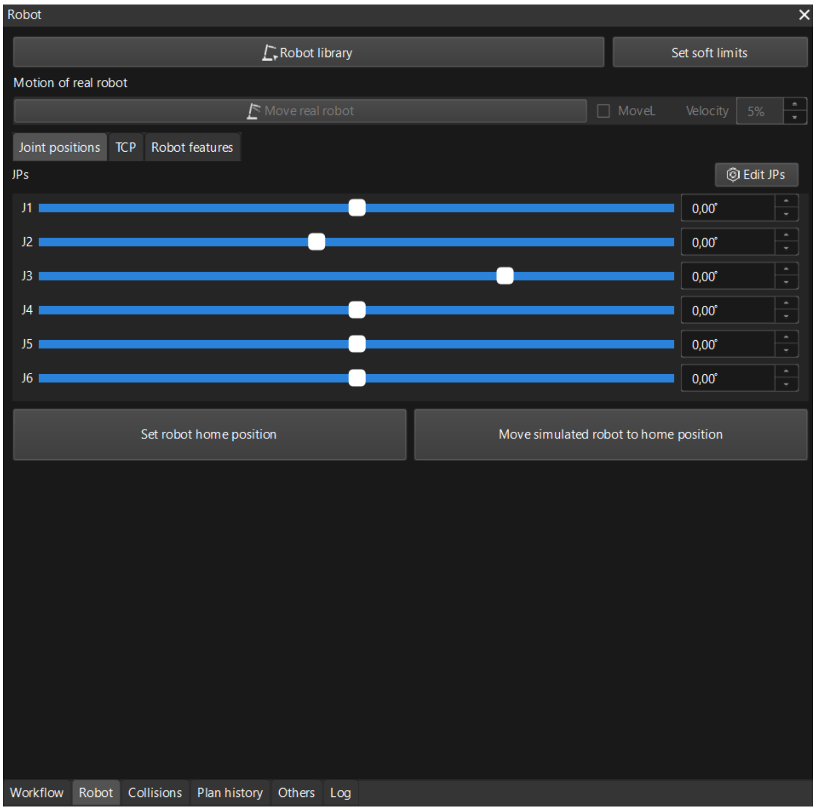
- If the robot is successfully connected, a robot icon with the robot model will be displayed in the Service Status bar, and the corresponding message is printed in the Log panel.
- If the connection fails, please check if any of the operations so far was incorrect.

Move the Robot

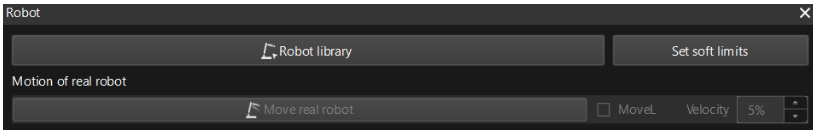
1. In Mech-Viz, click **Sync Robot** in the toolbar to synchronize the pose of the simulated robot to that of the real robot. Then, click **Sync Robot** again to disable the synchronization.



2. On the **Robot** tab, change the joint position of J1 slightly (for example, from 0° to 3°). The simulated robot will move accordingly.



3. Click [**Move real robot**].



When moving the robot, please ensure the safety of personnel. In the case of an emergency, press the emergency stop button on the teach pendant!

If you observe that the real robot moves to the pose of the simulated robot, the master-control communication has been established successfully.

2.3. Hand-Eye Calibration

In this tutorial, you will perform the automatic hand-eye calibration in the eye-to-hand setup.

The hand-eye calibration establishes **the transformation relationship between the camera and robot reference frames (that is camera extrinsic parameters)**. With this relationship, the object pose determined by the vision system can be transformed into that in the robot reference frame, which guides the robot in performing its tasks.

▶ https://www.youtube.com/watch?v=oMG6fCJFUYQ/PLVcMd7cW2rXVtrAejMyVQni2dUDv8bxje *(YouTube video)*

Video Tutorial: Hand-Eye Calibration (Master-Control)

Preparation before Calibration

In this section, you will mount the calibration board, adjust the camera parameters, and complete the pre-calibration configuration.

Mount the Calibration Board

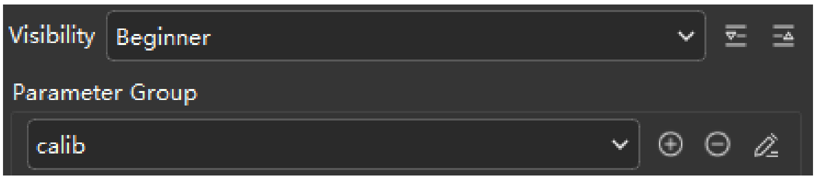
In the eye-to-hand setup, the calibration board needs to be mounted to the robot flange adapter.

Follow these steps:

1. Take the calibration board and flange adapter out of the system delivery package.
2. Use screws, gaskets, and nuts to fasten the flange adapter to the flange adapter.
3. Use screws, gaskets, and nuts to secure the calibration board to the flange adapter.
4. After mounting the calibration board, move the robot to the top of the lowest workpiece in the work area and into the camera's field of view (FOV).

Adjust Camera Parameters

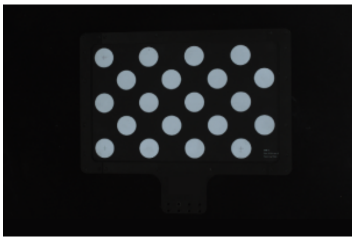
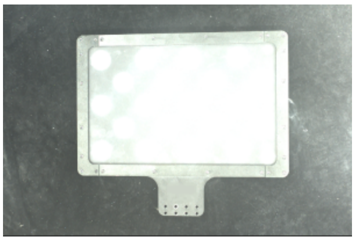
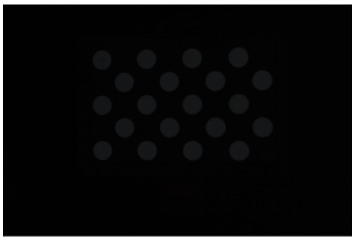
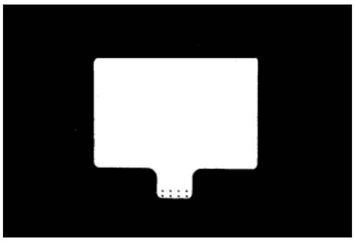
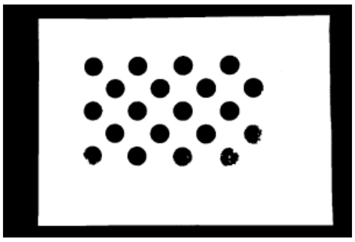
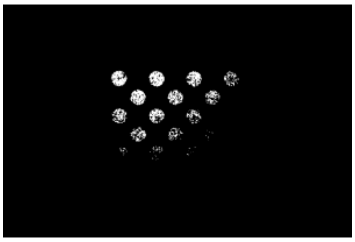
1. In Mech-Eye Viewer, connect to the camera, and then set **Parameter Group** to "calib".



Visibility: Beginner

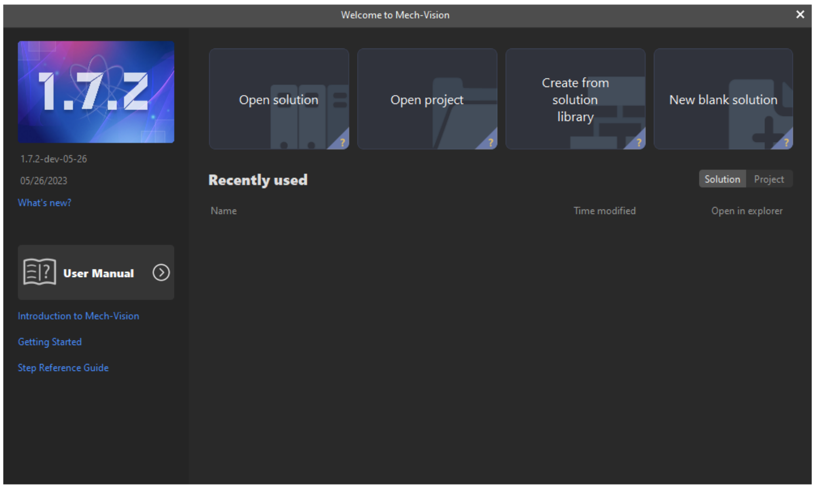
Parameter Group: calib

2. Adjust the 2D parameters to make sure that the 2D image of the calibration board is clear and neither overexposed nor underexposed.
3. Adjust 3D parameters to make sure that the obtained point clouds of the circles on the calibration board are complete and have clear contours. It is recommended to set the **Surface Smoothing** and **Outlier Removal** parameters of **Point Cloud Processing** to **Normal** to reduce point cloud fluctuation.

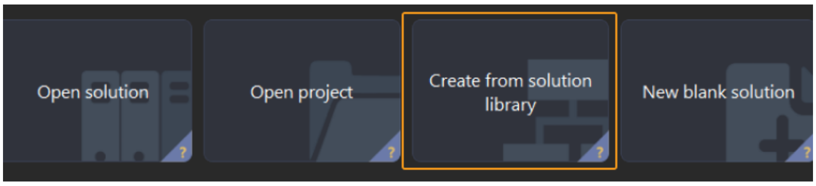
	Normal	Overexposed	Underexposed
2D image			
Point cloud			

Create a Mech-Vision Project and Save It

1. Open Mech-Vision. If the Welcome interface as shown below is displayed, it indicates that Mech-Vision is started successfully.

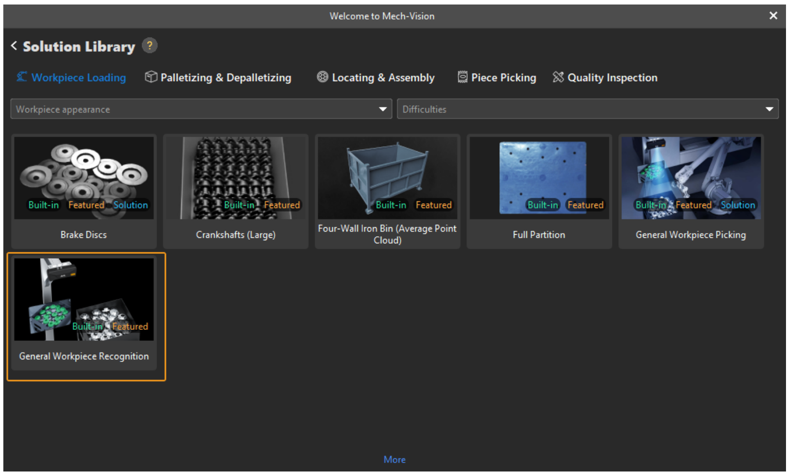


2. In the Welcome interface of Mech-Vision, click **[Create from solution library]** to open the Solution Library.



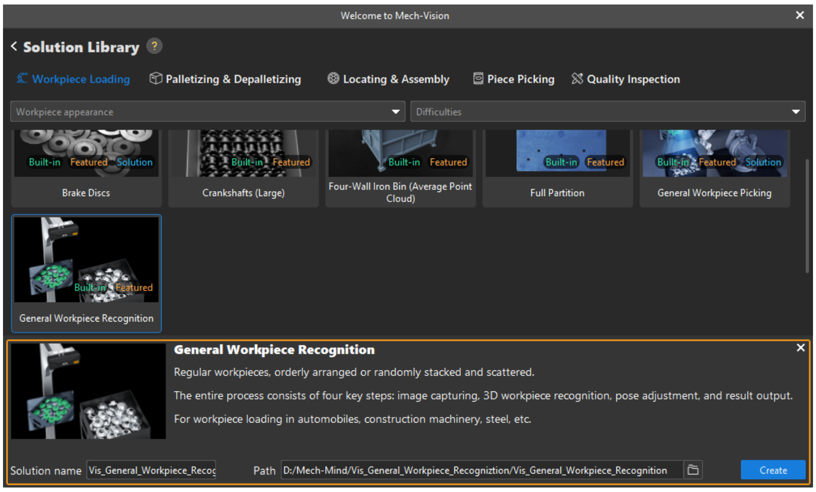
The Solution Library is a resource library that provides typical solutions or projects (with sample data) from various application scenarios.

3. Select the **General Workpiece Recognition** project from the Solution Library.



If you cannot find this project in the Solution Library, click **[More]** at the bottom of the Solution Library interface.

4. After this project is selected, the information introducing this project will be displayed on the lower part of the Solution Library interface. Set the solution name and path, and then click **[Create]**.



Welcome to Mech-Vision

Solution Library

General Workpiece Recognition

Regular workpieces, orderly arranged or randomly stacked and scattered.

The entire process consists of four key steps: image capturing, 3D workpiece recognition, pose adjustment, and result output.

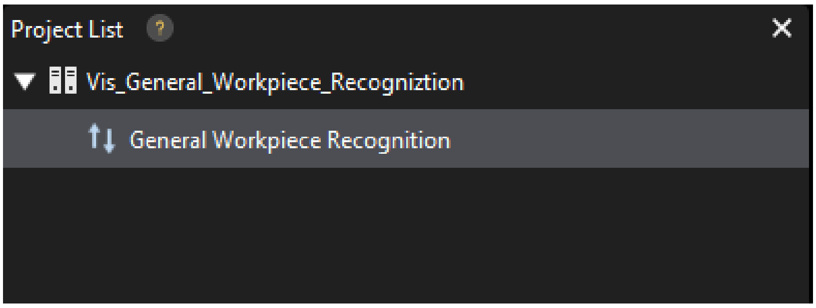
For workpiece loading in automobiles, construction machinery, steel, etc.

Solution name Vis_General_Workpiece_Recog

Path D:/Mech-Mind/Vis_General_Workpiece_Recogniztion/Vis_General_Workpiece_Recognition

After the project is created, the created solution and project will be displayed in the project list in the upper-left corner of the Mech-Vision main interface.

- A **solution** is a set of configurations and data related to robots and robot communication, vision processing, path planning, etc. that are required for the machine vision application.
- A **project** is a workflow of vision processing in the solution. Normally, a solution only contains one project, but it may contain more than one project in complex application scenarios. In this tutorial, the solution requires only one project.

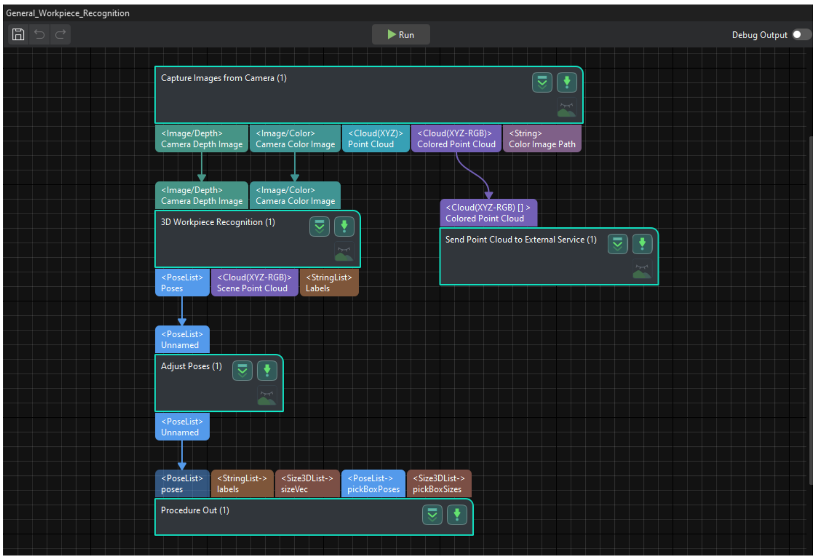


Project List

Vis_General_Workpiece_Recogniztion

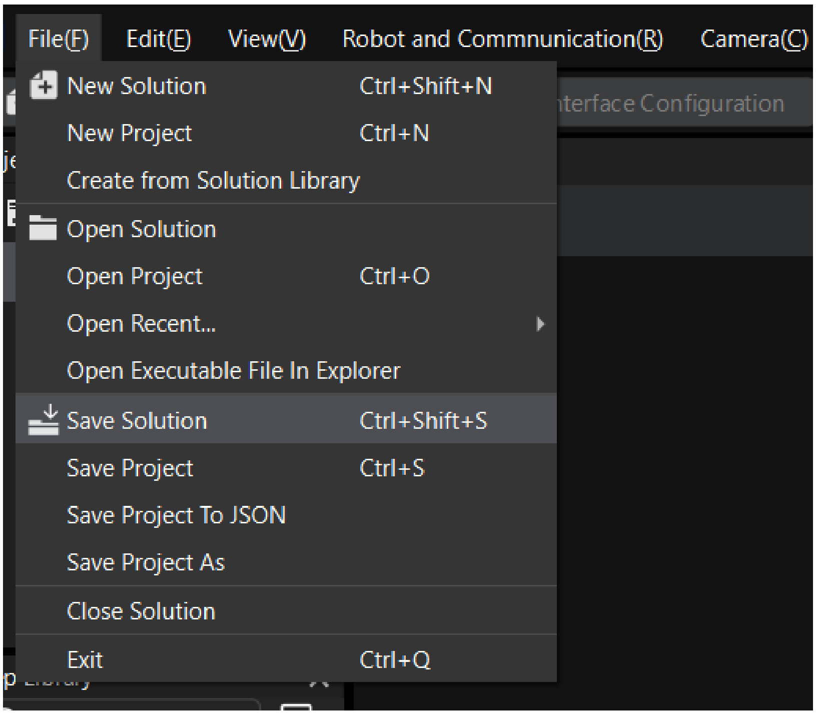
General Workpiece Recognition

In the Graphical Programming Workspace of the main interface, the workflow of the “General Workpiece Recognition” project will be displayed.



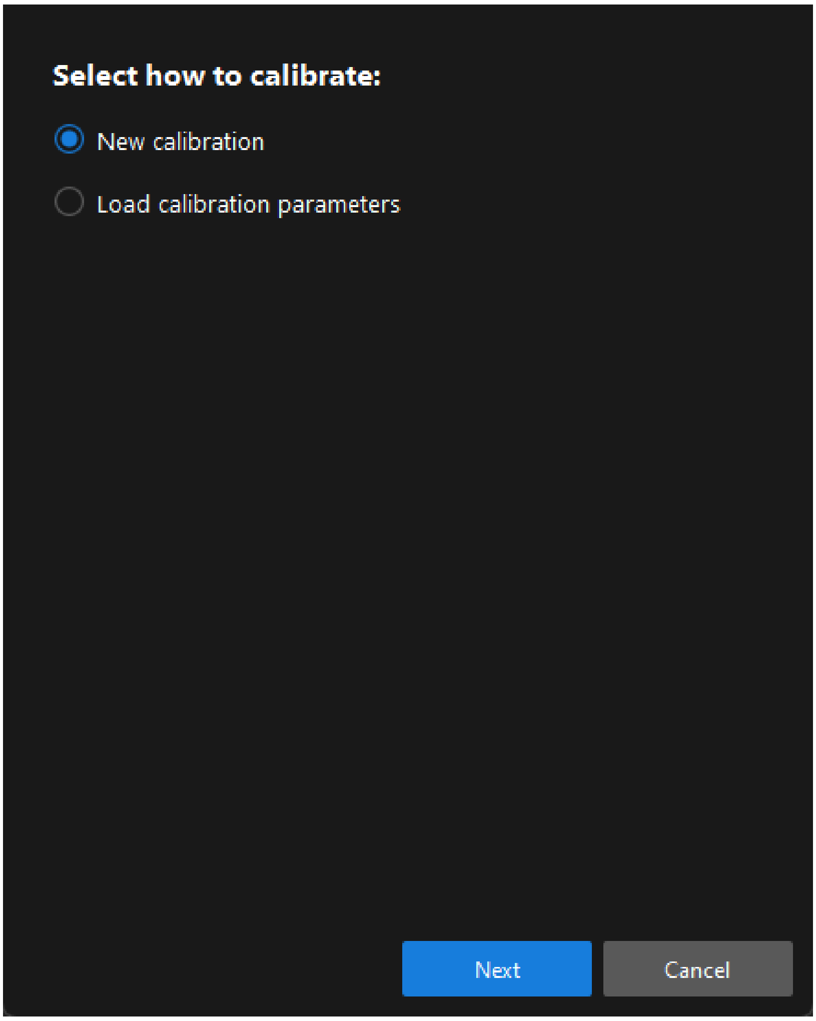
General_Workpiece_Recognition
Run
Debug Output
Capture Images from Camera (1)
<Image/Depth> Camera Depth Image
<Image/Color> Camera Color Image
<Cloud(XYZ)> Point Cloud
<Cloud(XYZ-RGB)> Colored Point Cloud
<String> Color Image Path
<Image/Depth> Camera Depth Image
<Image/Color> Camera Color Image
3D Workpiece Recognition (1)
<PoseList> Poses
<Cloud(XYZ-RGB)> Scene Point Cloud
<StringList> Labels
<Cloud(XYZ-RGB) [] > Colored Point Cloud
Send Point Cloud to External Service (1)
<PoseList> Unnamed
Adjust Poses (1)
<PoseList> Unnamed
<PoseList> poses
<StringList-> labels
<Size3DList-> sizeVec
<PoseList-> pickBoxPoses
<Size3DList-> pickBoxSizes
Procedure Out (1)

5. On the menu bar, select File › Save Solution.



Pre-calibration Configuration

1. In Mech-Vision, click the **[Camera Calibration (Standard)]** button on the toolbar. The **Configuration before Calibration** window will be prompted.
2. In the **Select how to calibrate** window, select the **New calibration** radio button, and then click the **[Next]** button.

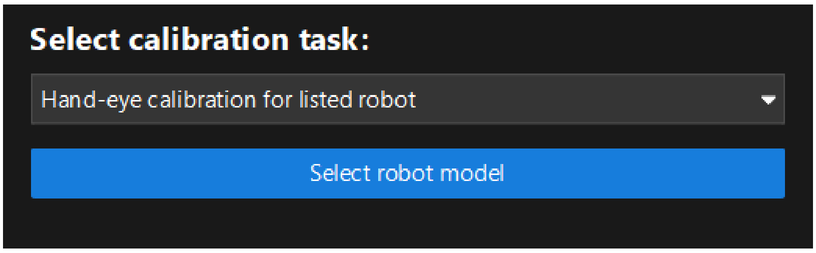


Select how to calibrate:

● New calibration

○ Load calibration parameters

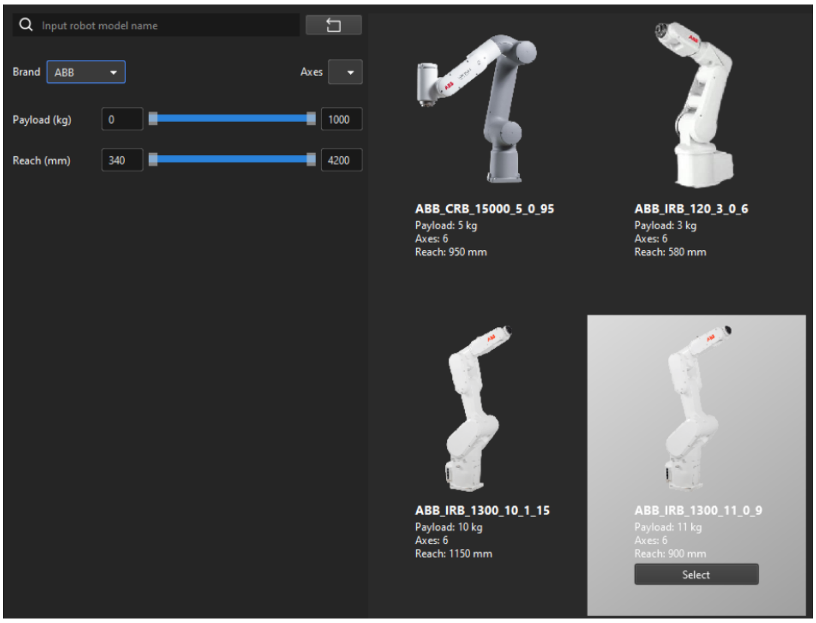
3. In the **Select calibration task** window, select **Hand-eye calibration for listed robot** from the drop-down list box, and click the **[Select robot model]** button.



Select calibration task:

Hand-eye calibration for listed robot

4. Click the **Brand** dropdown box, and select “ABB”. Then, at the right panel, select model “ABB_IRB_1300_11_0_9”, click the **[Select]** button, and then click the **[Next]** button.



Input robot model name

Brand: ABB

Axes:

Payload (kg): 0 – 1000

Reach (mm): 340 – 4200

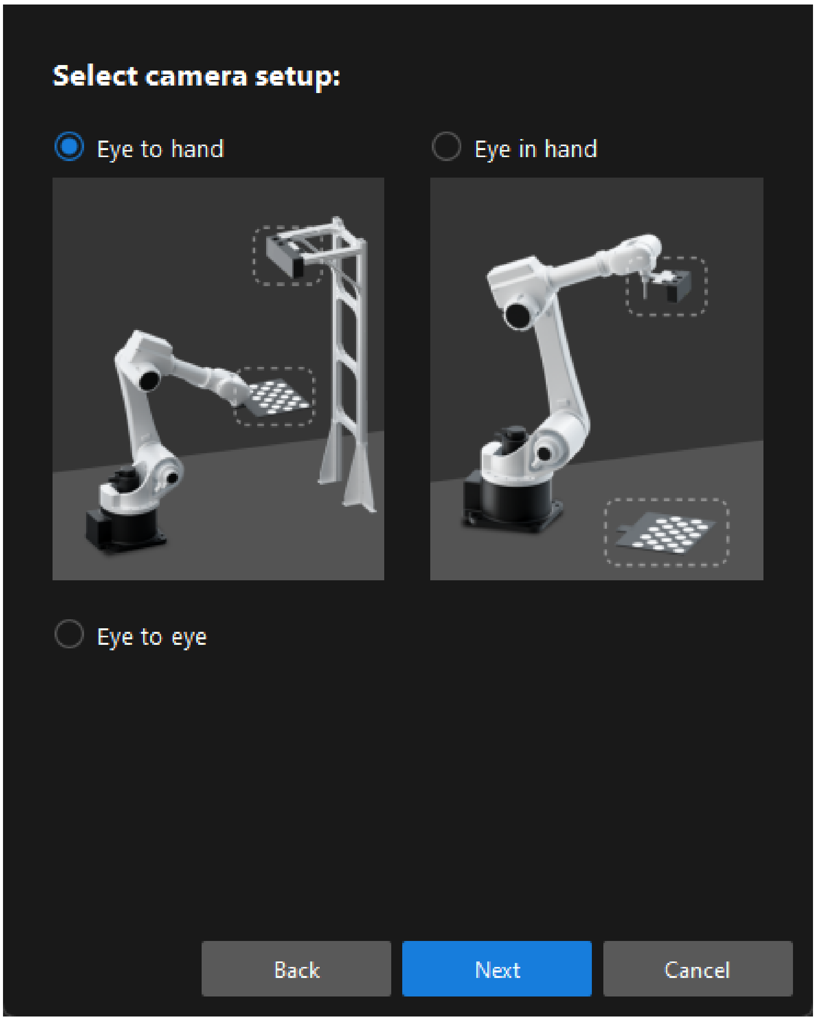
ABB_CRB_15000_5_0_95
Payload: 5 kg
Axes: 6
Reach: 950 mm

ABB_IRB_120_3_0_6
Payload: 3 kg
Axes: 6
Reach: 580 mm

ABB_IRB_1300_10_1_15
Payload: 10 kg
Axes: 6
Reach: 1150 mm

ABB_IRB_1300_11_0_9
Payload: 11 kg
Axes: 6
Reach: 900 mm

5. In the **Select camera setup** window, select the **Eye to hand** radio button, and then click the **[Next]** button.



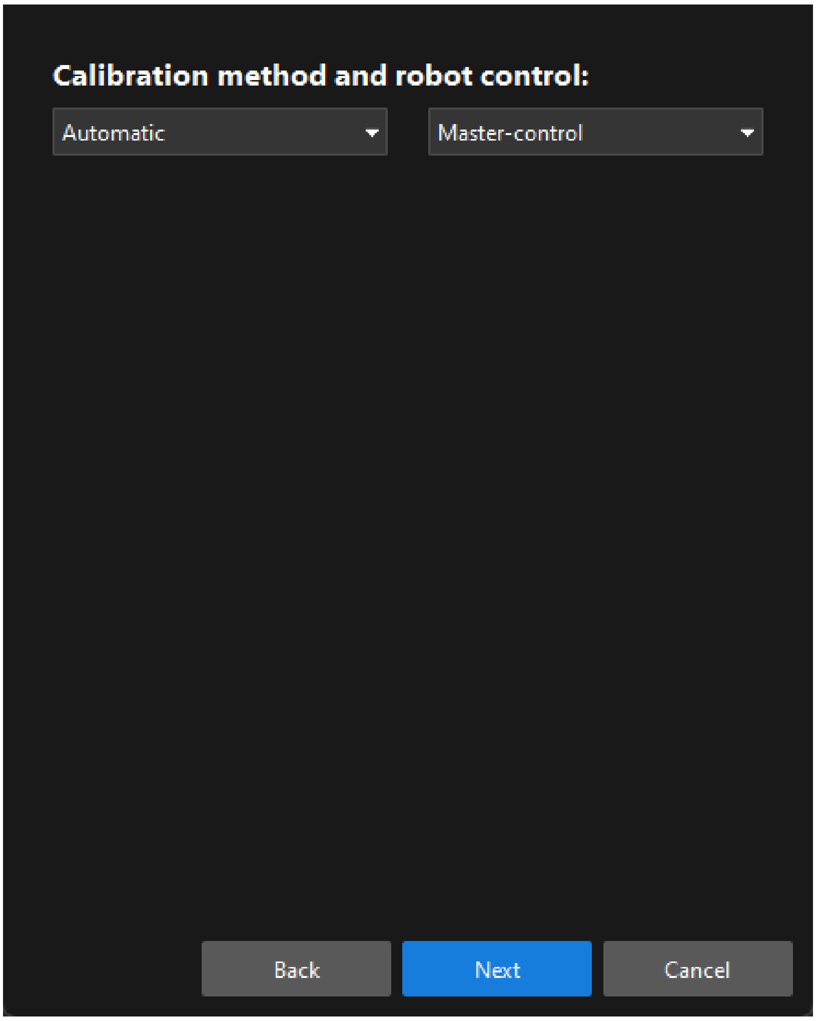
Select camera setup:

● Eye to hand

○ Eye in hand

○ Eye to eye

6. In the **Calibration method and robot control** window, select **Automatic** and **Master-control**, and then click the **[Next]** button.

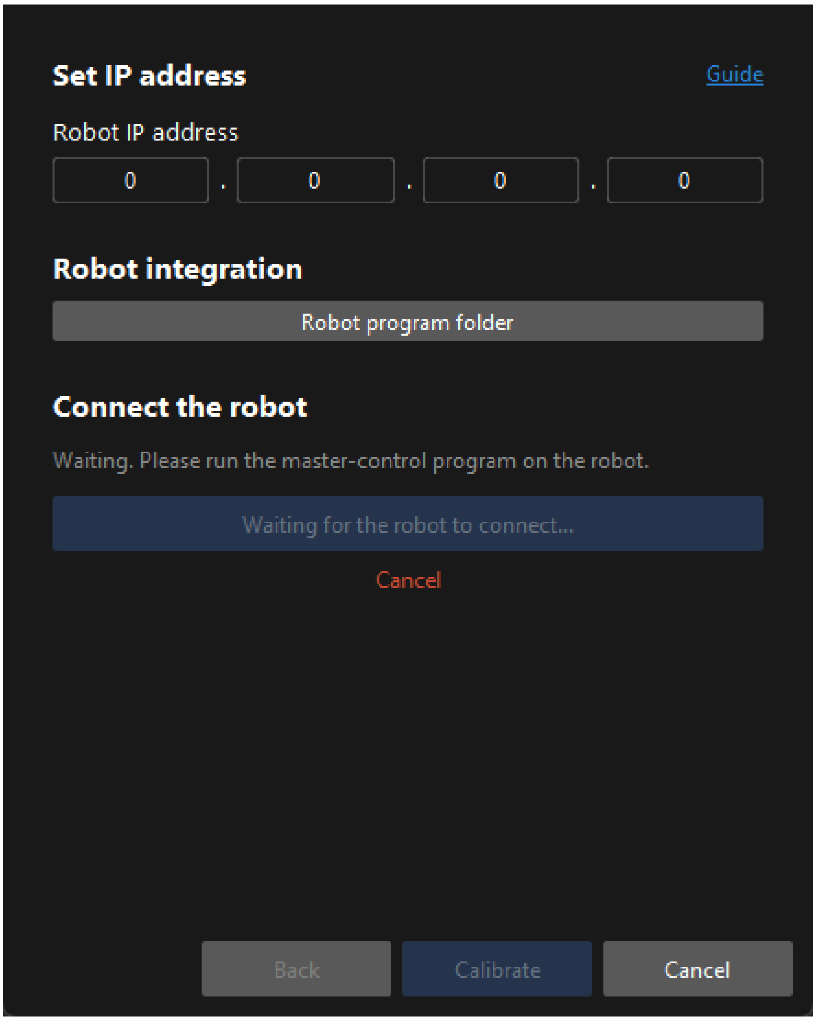


Calibration method and robot control:

Automatic

Master-control

7. In the **Communication Settings** window, set the **Robot IP address** parameter to the real robot's IP address.



Set IP address

Robot IP address

0 . 0 . 0 . 0

Robot integration

Connect the robot

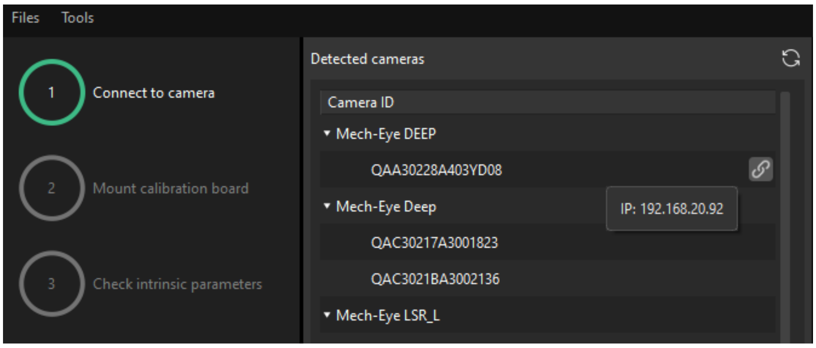
Waiting. Please run the master-control program on the robot.

8. On the robot teach pendant, confirm that the MM main program has been started.
9. Return to Mech-Vision, and click the [**Connect the robot**] button in the **Connect the robot** area. The button will turn into **Waiting for the robot to connect...**
10. Wait until the "Connected" status message is displayed in the **Connect the robot** area, and then click the [**Calibrate**] button. The **Calibration (Eye to Hand)** window will be prompted.

Calibration Procedure

Connect to a Camera

1. In the **Connect to Camera** step, find the camera to connect in the **Detected Cameras** list, and click the 🔗 button.



Detected cameras

Camera ID
- Mech-Eye DEEP
  - QAA30228A403YD08
- Mech-Eye Deep
  - QAC30217A3001823
  - QAC3021BA3002136
- Mech-Eye LSR_L

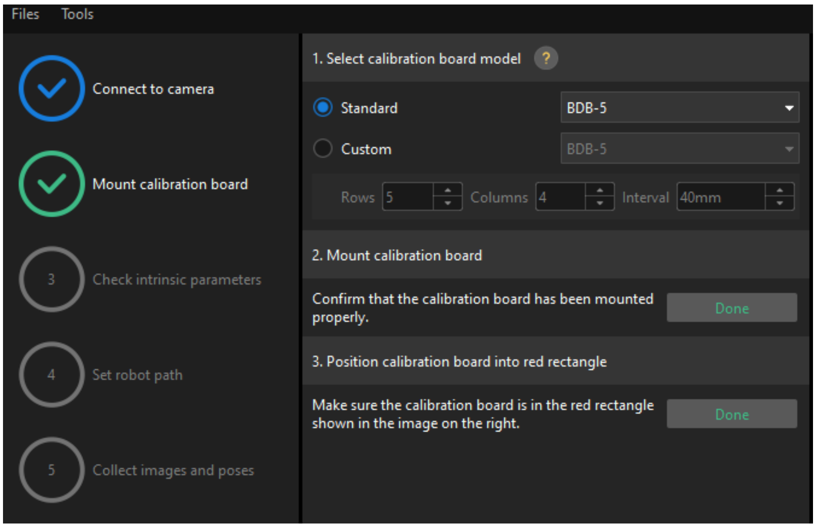
IP: 192.168.20.92

2. After the camera is connected, click the **[Capture once]** or **[Capture live]** button.
3. In the right **Image viewer** panel, ensure that the captured 2D image and depth map meet the calibration requirements and click the **[Next]** button on the bottom bar.

> If the captured image does not meet the calibration requirements, you need to open the Mech-Eye Viewer software to adjust the 2D and 3D exposure parameters and re-capture images.

Mount the Calibration Board

1. In the **Mount calibration board** step, select the **Standard** radio button and then select the calibration board model according to its model nameplate in the **1. Select calibration board type** area.
2. Make sure that the calibration board has been attached to the robot flange securely, and then click the **[Confirm]** button in the **2. Mount calibration board** area.
3. Confirm that the calibration board is in the center of the camera's field of view (the red rectangle), and then click the **[Confirm]** button in the **3. Position calibration board into red rectangle** area.



1. Select calibration board model

● Standard BDB-5

○ Custom BDB-5

Rows 5 Columns 4 Interval 40mm

2. Mount calibration board

Confirm that the calibration board has been mounted properly.

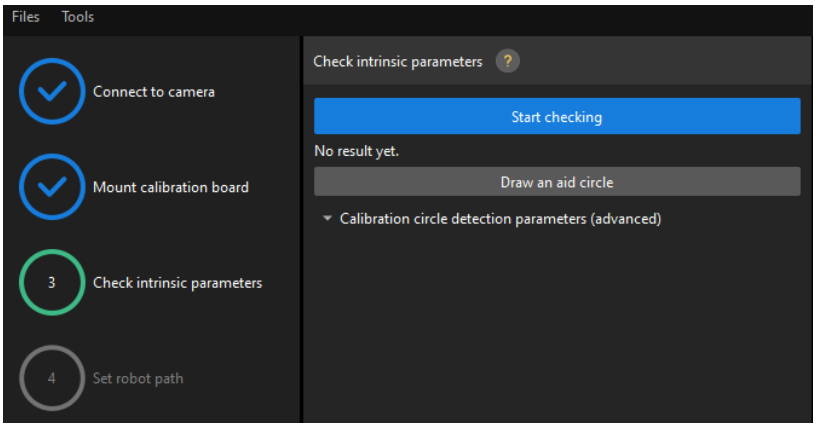
3. Position calibration board into red rectangle

Make sure the calibration board is in the red rectangle shown in the image on the right.

4. After all the operations related to the calibration board are completed, click the **[Next]** button on the bottom bar.

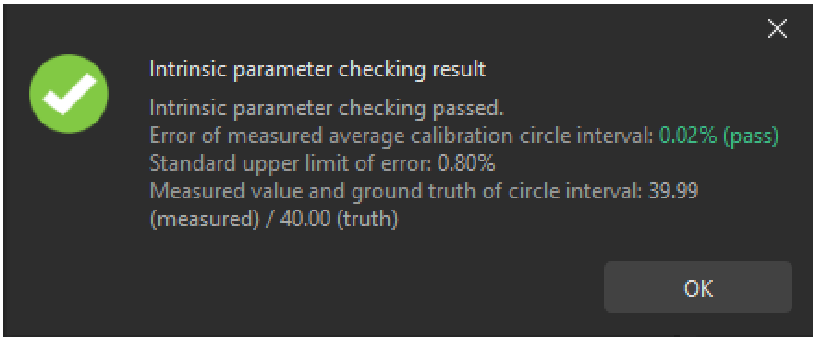
Check Intrinsic Parameters

1. In the **Check intrinsic parameters** step, click the **[Start checking]** button.


Files
Tools
Connect to camera
Mount calibration board
3
Check intrinsic parameters
4
Set robot path
Check intrinsic parameters
Start checking
Draw an aid circle
Calibration circle detection parameters (advanced)

No result yet.

2. Confirm that the camera intrinsic parameters check passes, and then click the [**Next**] button on the bottom bar.



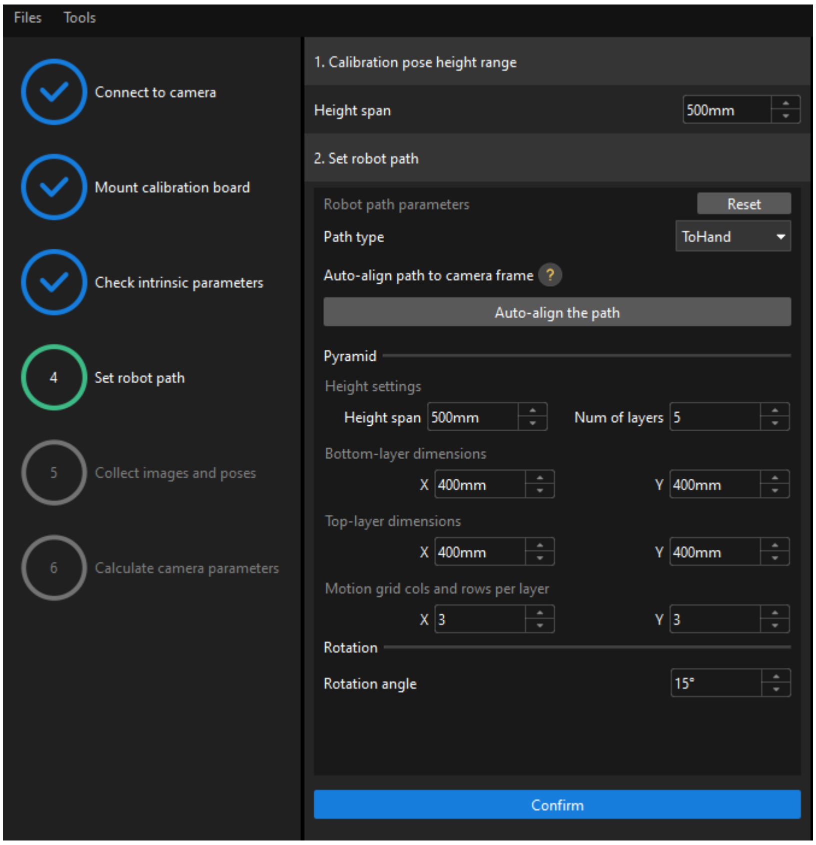
Intrinsic parameter checking result

Intrinsic parameter checking passed.
Error of measured average calibration circle interval: 0.02% (pass)
Standard upper limit of error: 0.80%
Measured value and ground truth of circle interval: 39.99 (measured) / 40.00 (truth)

If the camera intrinsic parameter check fails, adjust detection parameters by drawing aid circles or manually adjusting the calibration circle detection parameters.

Set Robot Path

1. In the **Set robot path** step, specify the **Height span** parameter. Please set this parameter according to the working distance range of the calibration board in the camera height direction.



1. Calibration pose height range

Height span 500mm

2. Set robot path

Robot path parameters

Path type ToHand

Auto-align path to camera frame ?

Pyramid

Height settings

Height span 500mm Num of layers 5

Bottom-layer dimensions

X 400mm Y 400mm

Top-layer dimensions

X 400mm Y 400mm

Motion grid cols and rows per layer

X 3 Y 3

Rotation

Rotation angle 15°

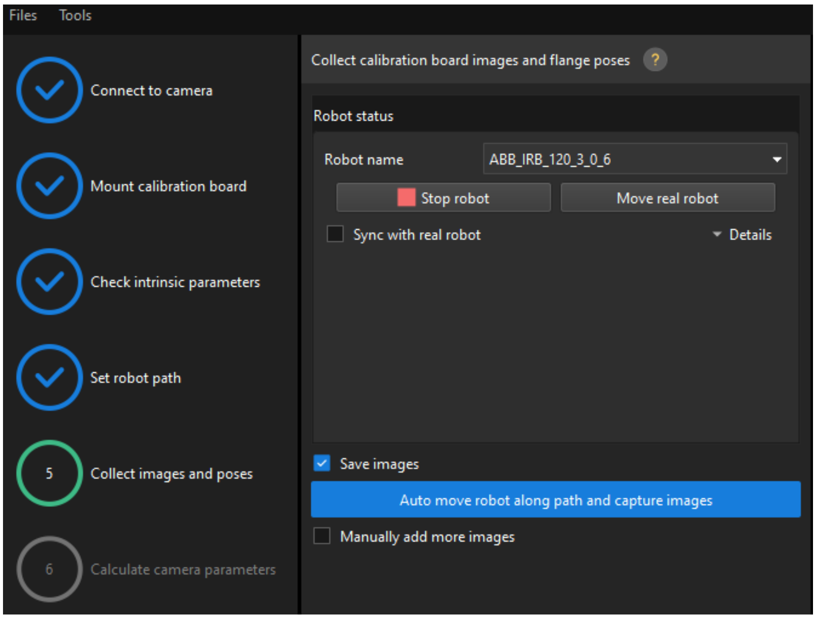
2. Set the **Path type** to **ToHand**, specify the pyramid parameters **Height span**, **Num of layers**, **Bottom-layer dimensions X/Y**, **Top-layer dimensions X/Y**, and **Motion grid cols and rows per layer**, specify the satellite parameter **Rotation angle**, and then click the **[Confirm]** button.

   The set robot path should cover the work area.

3. In the right **Scene Viewer** panel, confirm that the waypoints of the automatically generated motion path will not collide with obstacles in the environment, and then click the **[Next]** button.

Collect Images and Poses

1. In the **Collect images and poses** step, select the **Save images** checkbox.



Collect calibration board images and flange poses ?

Robot status

Robot name: ABB_IRB_120_3_0_6

Stop robot | Move real robot

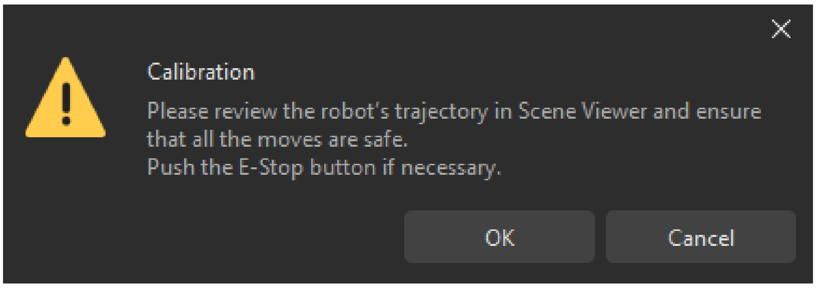
□ Sync with real robot ▾ Details

☑ Save images

Auto move robot along path and capture images

□ Manually add more images

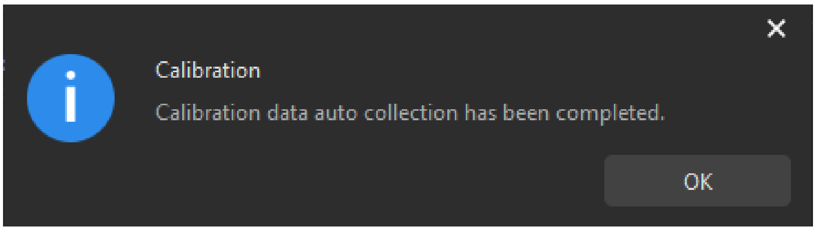
2. Click the [**Control robot to auto move along path and capture images**] button.
3. Read the safety window carefully and click the [**OK**] button.



Calibration

Please review the robot's trajectory in Scene Viewer and ensure that all the moves are safe.
Push the E-Stop button if necessary.

4. Wait until the robot finishes moving along the preset path and the camera finishes capturing images on all waypoints. Captured images can be seen in the right **Marker Image and Pose List** panel during this process.
5. After the automatic image capturing finishes, click the [**OK**] button, and then click the [**Next**] button on the bottom bar.

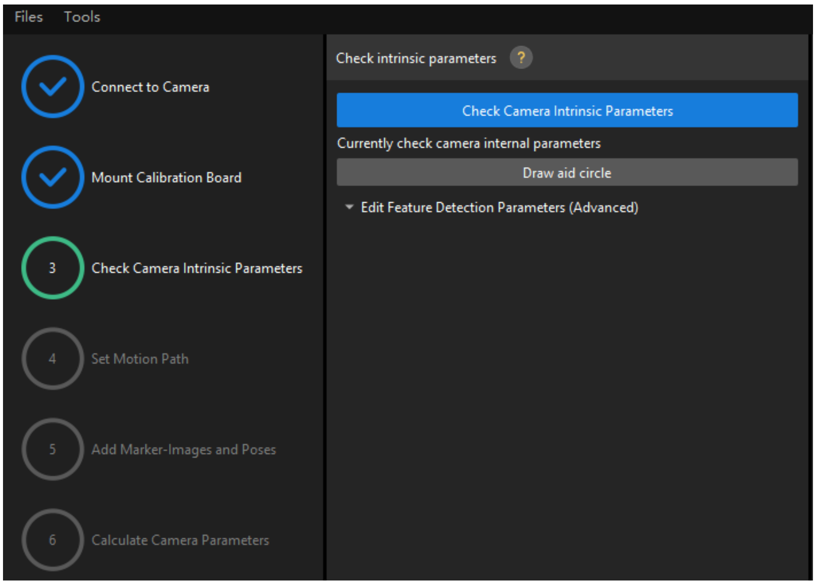


Calibration

Calibration data auto collection has been completed.

Calculate Camera Parameters

1. In the **Calculate camera parameters** window, click the [**Calculate camera extrinsic parameters**] button.


Files
Tools
Connect to Camera
Mount Calibration Board
3
Check Camera Intrinsic Parameters
4
Set Motion Path
5
Add Marker-Images and Poses
6
Calculate Camera Parameters
Check intrinsic parameters
?
Check Camera Intrinsic Parameters
Draw aid circle
Edit Feature Detection Parameters (Advanced)

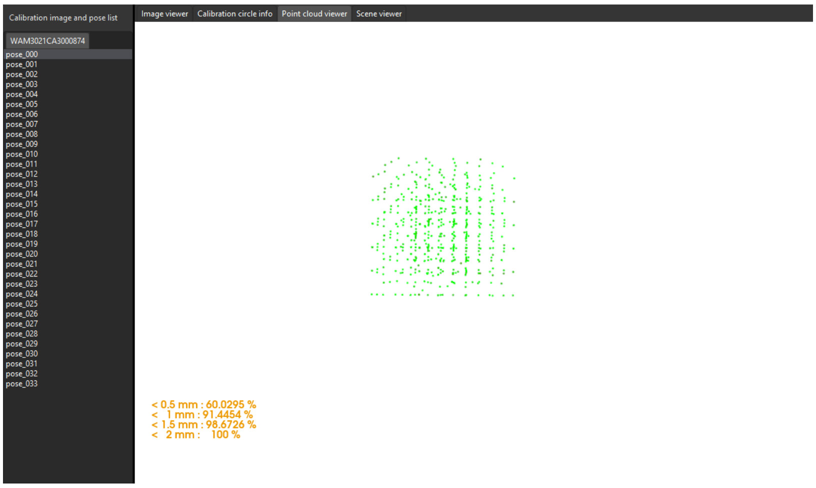
Currently check camera internal parameters

2. In the prompted window indicating calibration success, click the [OK] button.
3. View the calibration error of the point cloud in the right **Point cloud viewer** panel.

> The error point cloud shows the deviation between the calculated value and the actual value of the circles on the calibration board.

4. Confirm that the calibration accuracy meets the project requirements.

   Determine the error value with 100% to obtain the rough calibration accuracy. For example, the calibration accuracy in the following figure is within 2.5 mm.



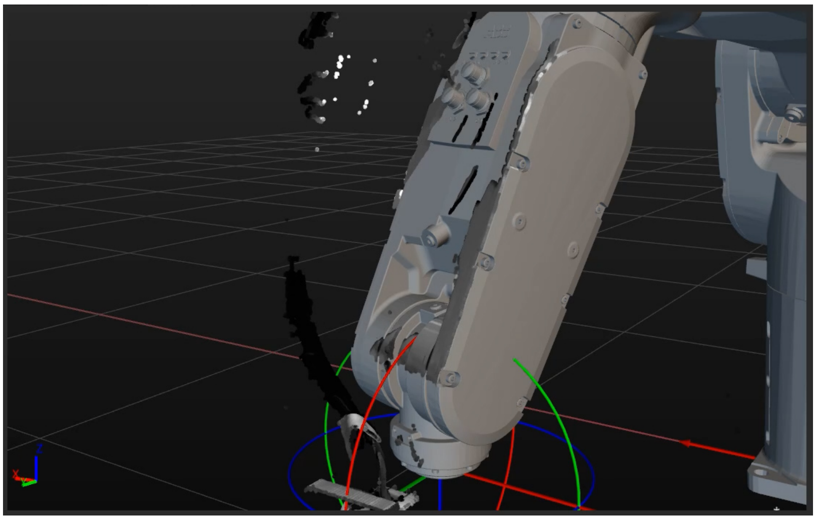
```
< 0.5 mm : 60.0295 %
<   1 mm : 91.4454 %
< 1.5 mm : 98.6726 %
<   2 mm :    100 %
```

   To improve the calibration accuracy, please refer to the section Calibration Result Check and Analysis.

Validate and Save the Calibration Result

1. After calibration, move the robot arms into the FOV of the camera.
2. In the **Calculate camera parameters** step, click the [**Re-calculate camera extrinsic parameters**] button. This operation triggers the camera to capture images.
3. Click **Scene viewer** to check the coincidence degree between the point cloud of the robot and the robot model.

If the point cloud of the robot approximately coincides with the robot model, the calibration is successful.



4. Click the [**Save**] button on the bottom bar. In the prompted **Save Calibration Files** dialog box, click the [**OK**] button. The camera calibration result will be automatically saved in the “calibration” directory of the project.

2.4. Workpiece Locating

Before using this tutorial, you should have created a Mech-Vision solution using the “General Workpiece Recognition” case project in the “Create a Mech-Vision Project and Save It” section.

In this tutorial, you will first learn the project workflow, and then deploy the project by adjusting the Step parameters to recognize the workpiece poses and output the vision result.

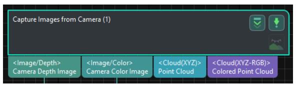
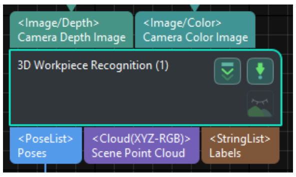
In this tutorial, you will need to convert the model file of the workpiece in the CAD format into a point cloud matching model. Since it takes a long time to prepare the CAD model file, you are recommended to prepare the CAD model file of the workpiece before using this tutorial.

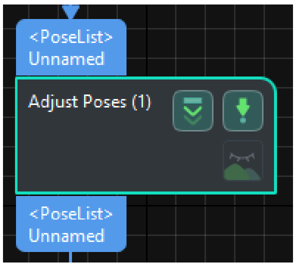
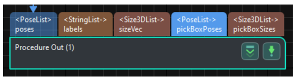
▶ https://www.youtube.com/watch?v=D3On5xx14F4/PLVcMd7cW2rXVtrAejMyVQni2dUDv8bxje *(YouTube video)*

Video Tutorial: Workpiece Locating

Introduction to the Project Workflow

The following table describes each Step in the project workflow.

No.	Phase	Step	Image	Description
1	Capture images	Capture Images from Camera	 Capture Images from Camera (1) <Image/Depth> Camera Depth Image <Image/Color> Camera Color Image <Cloud(XYZ)> Point Cloud <Cloud(XYZ-RGB)> Colored Point Cloud	Connect to the camera and capture images
2	Recognize workpieces	3D Workpiece Recognition	 <Image/Depth> Camera Depth Image <Image/Color> Camera Color Image 3D Workpiece Recognition (1) <PoseList> Poses <Cloud(XYZ-RGB)> Scene Point Cloud <StringList> Labels	Use 3D matching algorithms to calculate the workpieces’ poses (as pick points)

No.	Phase	Step	Image	Description
3	Adjust poses	Adjust poses	 <PoseList> Unnamed Adjust Poses (1) <PoseList> Unnamed	Transform the pick points from the camera reference frame to the robot reference frame
4	Output the vision result	Procedure Out	 <PoseList> poses <StringList-> labels <Size3DList-> sizeVec <PoseList-> pickBoxPoses <Size3DList-> pickBoxSizes Procedure Out (1)	Output the workpieces' poses for the robot to pick

A pick point refers to a point on the workpiece on which the robot can pick the object.

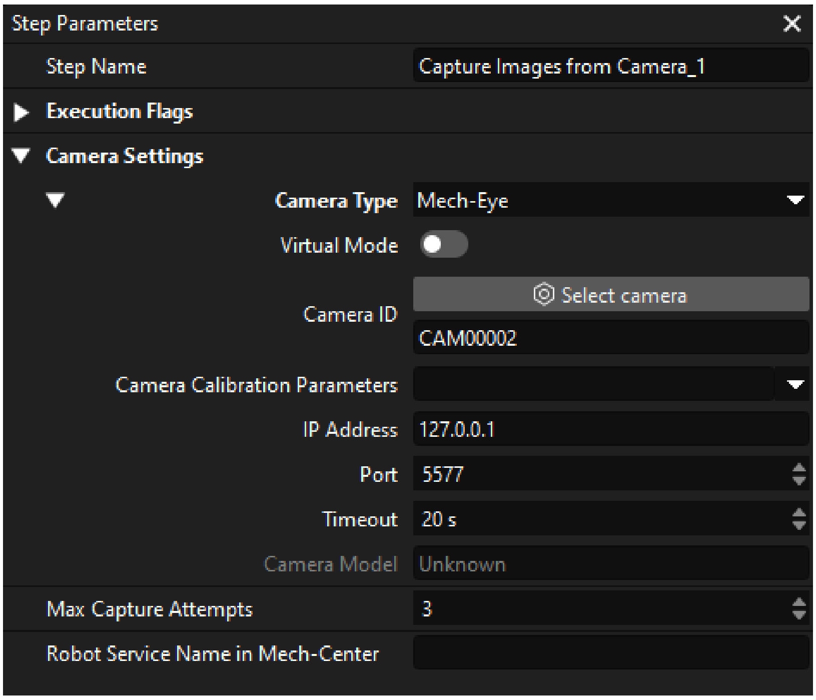
Adjust Step Parameters

In this section, you will deploy the project by adjusting the parameters of each Step.

Capture Images from Camera

You should adjust the parameters of the "Capture Images from Camera" Step to connect to the camera.

1. Select the "Capture Images from Camera" Step, and click **[Select camera]** on the Step parameters tab.



Step Parameters

Step Name: Capture Images from Camera_1

▶ Execution Flags

▼ Camera Settings

Camera Type: Mech-Eye

Virtual Mode: ○

Camera ID: Select camera

CAM00002

Camera Calibration Parameters:

IP Address: 127.0.0.1

Port: 5577

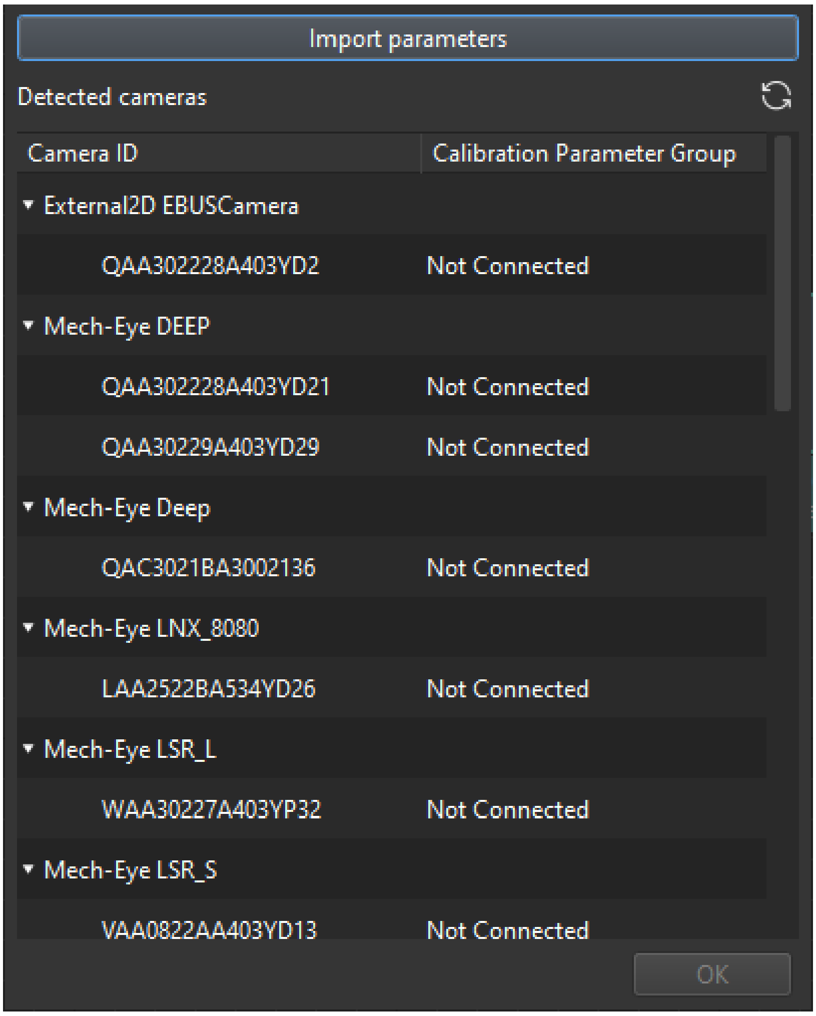
Timeout: 20 s

Camera Model: Unknown

Max Capture Attempts: 3

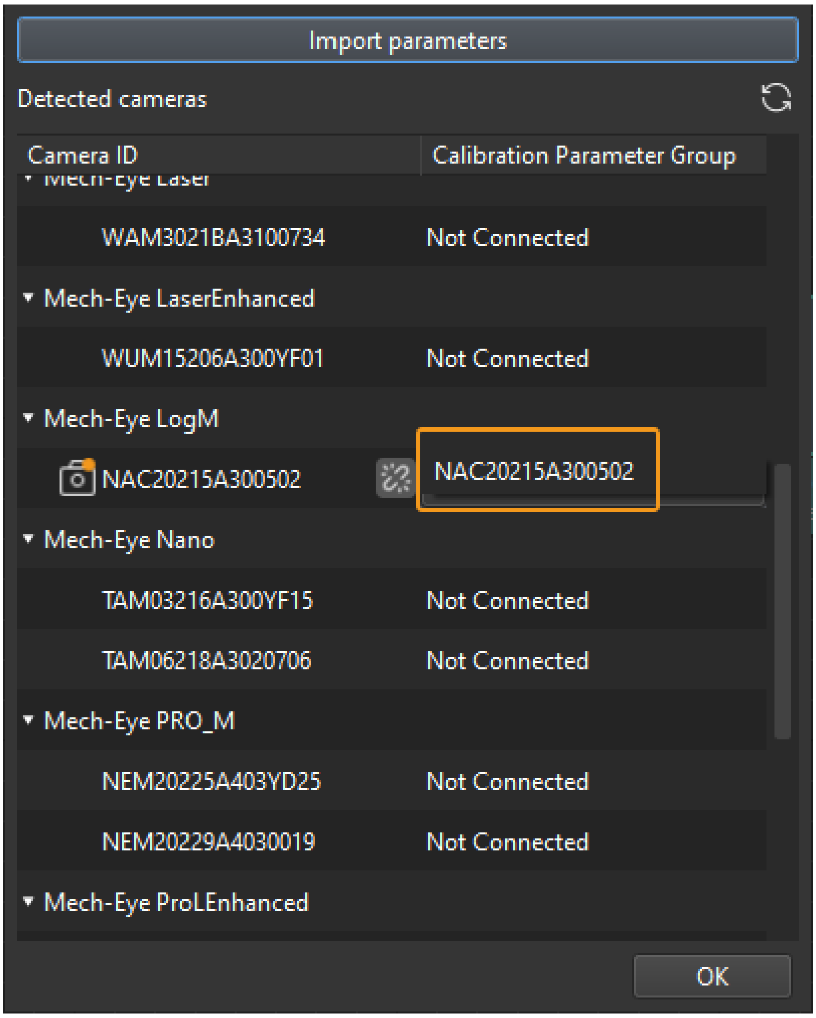
Robot Service Name in Mech-Center:

2. In the pop-up window, click 🔗 on the right of the camera serial No. to connect the camera. After the camera is connected, the icon 🔗 will turn into ⛓.



Camera ID	Calibration Parameter Group
▾ External2D EBUSCamera	
QAA302228A403YD2	Not Connected
▾ Mech-Eye DEEP	
QAA302228A403YD21	Not Connected
QAA30229A403YD29	Not Connected
▾ Mech-Eye Deep	
QAC3021BA3002136	Not Connected
▾ Mech-Eye LNX_8080	
LAA2522BA534YD26	Not Connected
▾ Mech-Eye LSR_L	
WAA30227A403YP32	Not Connected
▾ Mech-Eye LSR_S	
VAA0822AA403YD13	Not Connected

After the camera is connected, select the parameter group. Click **[Select from]** and select the displayed parameter group.

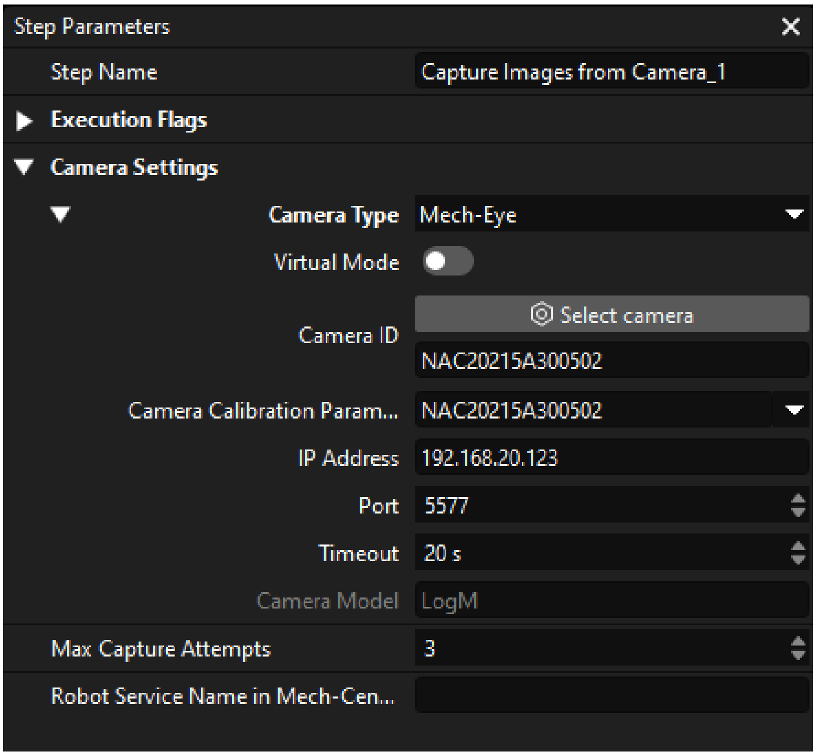


Import parameters

Detected cameras

Camera ID	Calibration Parameter Group
▾ Mech-Eye Laser	
WAM3021BA3100734	Not Connected
▾ Mech-Eye LaserEnhanced	
WUM15206A300YF01	Not Connected
▾ Mech-Eye LogM	
NAC20215A300502	NAC20215A300502
▾ Mech-Eye Nano	
TAM03216A300YF15	Not Connected
TAM06218A3020706	Not Connected
▾ Mech-Eye PRO_M	
NEM20225A403YD25	Not Connected
NEM20229A4030019	Not Connected
▾ Mech-Eye ProLEnhanced	

3. After the camera is connected and the parameter group is selected, the calibration parameter group, IP address, and ports of the camera will be obtained automatically. Just keep the default settings of the other parameters.



Step Parameters

Step Name: Capture Images from Camera_1

▶ Execution Flags

▼ Camera Settings

Camera Type: Mech-Eye

Virtual Mode: ○

Camera ID: Select camera — NAC20215A300502

Camera Calibration Param...: NAC20215A300502

IP Address: 192.168.20.123

Port: 5577

Timeout: 20 s

Camera Model: LogM

Max Capture Attempts: 3

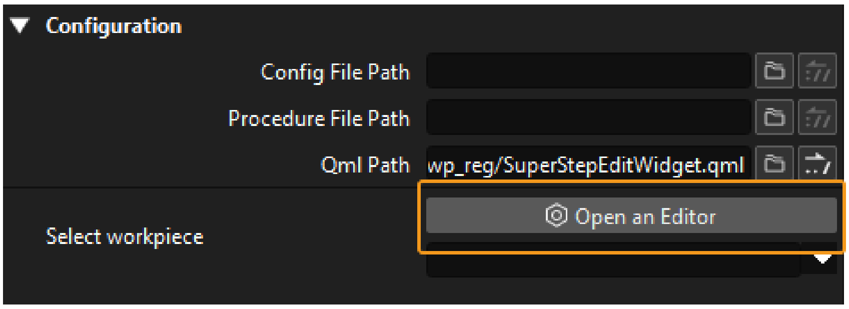
Robot Service Name in Mech-Cen...:

Now, you have connected the software to the camera.

3D Workpiece Recognition

The “3D Workpiece Recognition” Step has integrated a 3D Workpiece Recognition visualized configurator, which provides point cloud preprocessing, model-based matching, and pose (pick point) calculation.

Select the “3D Workpiece Recognition” Step, and click **[Open an Editor]** on the Step parameters tab.



▼ Configuration

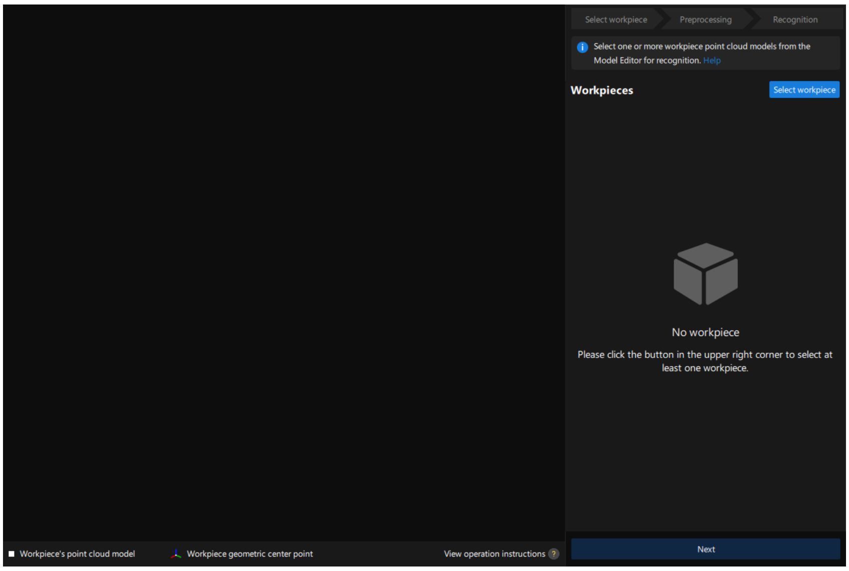
Config File Path:

Procedure File Path:

Qml Path: wp_reg/SuperStepEditWidget.qml

Select workpiece: Open an Editor

The 3D Workpiece Recognition visualized configurator is shown below.



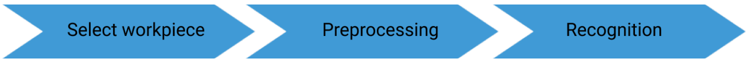
Select workpiece
Preprocessing
Recognition
Workpieces
Select workpiece
Workpiece's point cloud model
Workpiece geometric center point
View operation instructions
Next

Select one or more workpiece point cloud models from the Model Editor for recognition. Help

No workpiece

Please click the button in the upper right corner to select at least one workpiece.

Then follow the procedure to recognize workpieces.



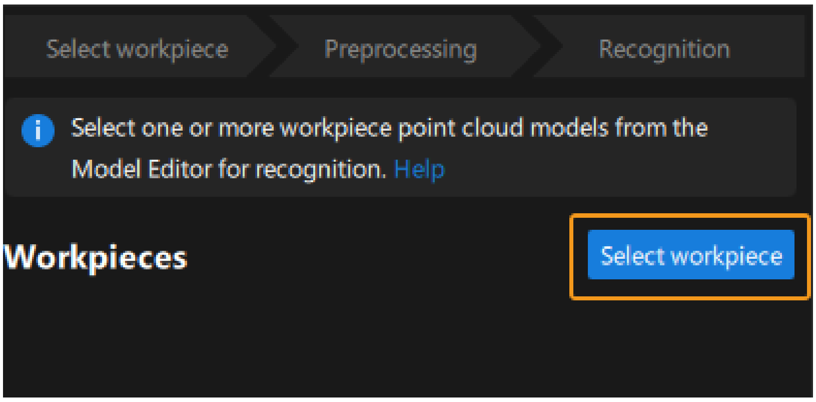
Select workpiece → Preprocessing → Recognition

Select Workpiece

After entering the 3D Workpiece Recognition visualized configurator, you need to make the point cloud model for the workpieces to recognize.

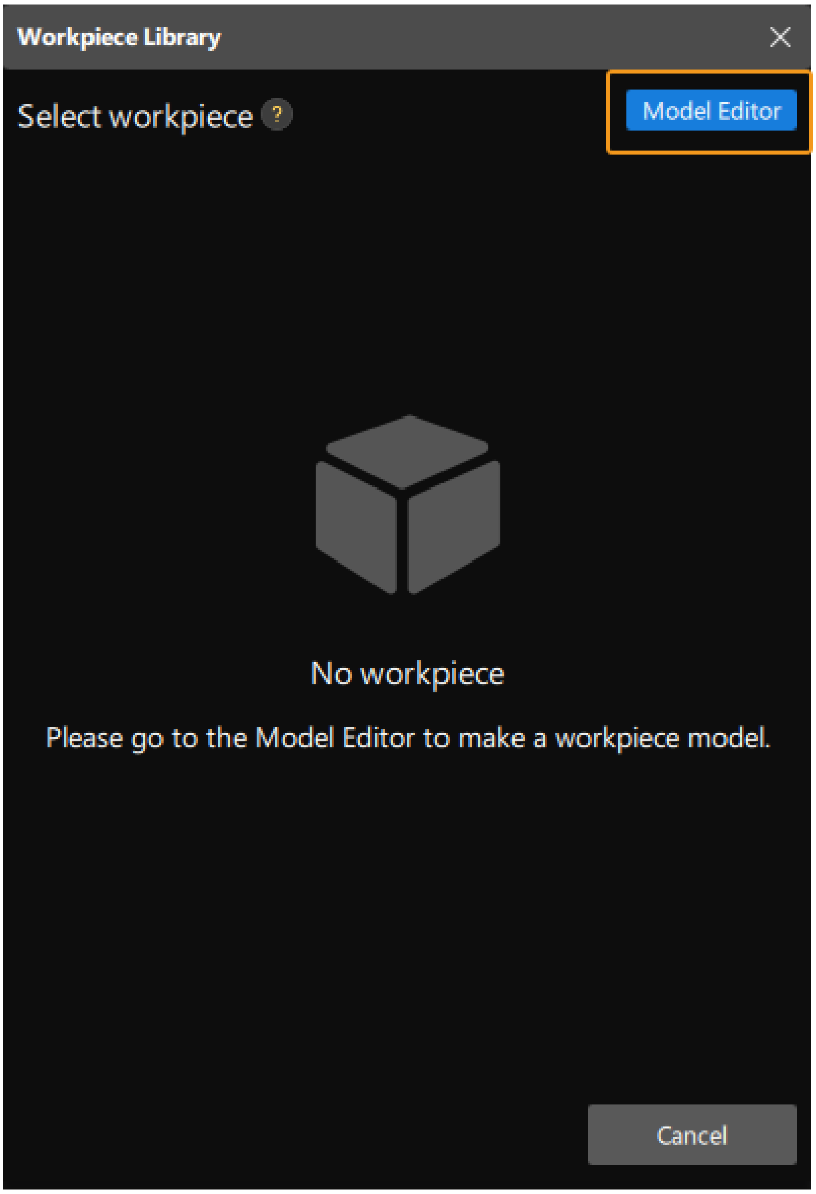
1. Open the Model Editor.

   On the upper part of the 3D Workpiece Recognition visualized configurator. click **[Select workpiece]**.



Select one or more workpiece point cloud models from the Model Editor for recognition. Help

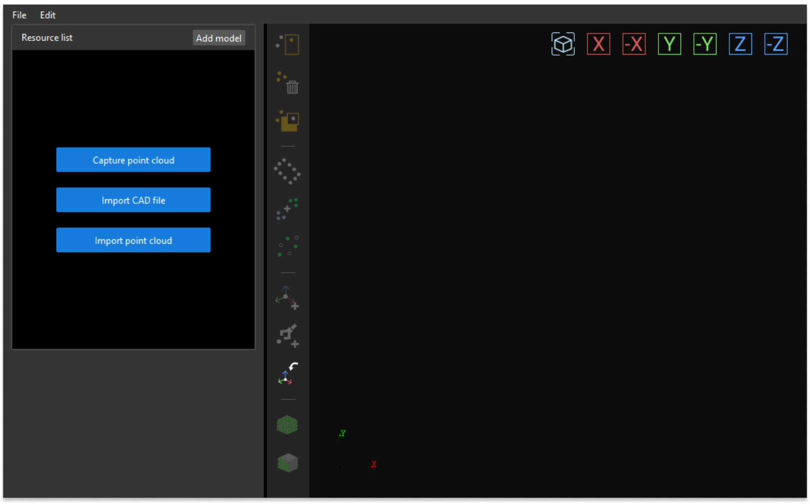
   In the pop-up Workpiece Library window, click **[Model Editor]**.



No workpiece

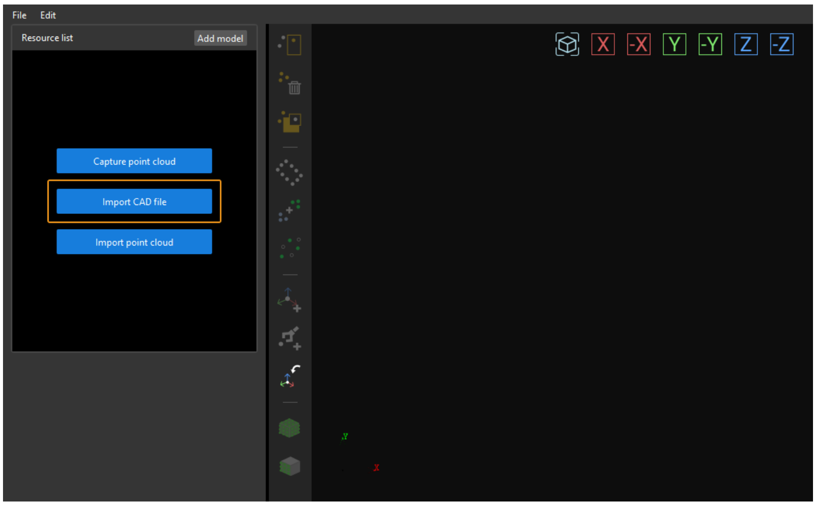
Please go to the Model Editor to make a workpiece model.

The following figure shows the Model Editor.

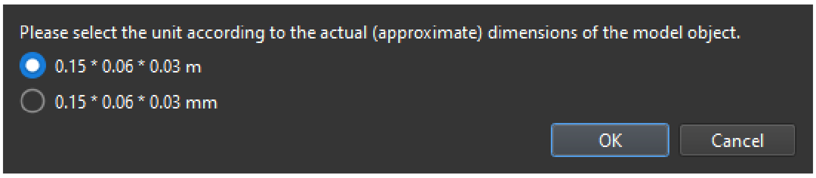


2. Import the CAD file.

On the Matching Model and Pick Point Editor interface, click [**Import CAD file**].



Import the prepared workpiece model in STL format, select the unit of measurement of the model object with which the workpiece was created, and click [**OK**].

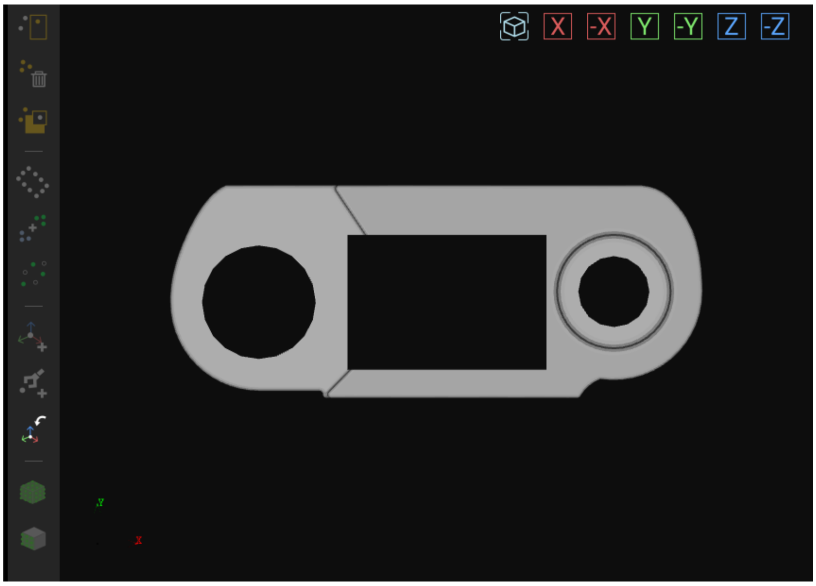


Please select the unit according to the actual (approximate) dimensions of the model object.

● 0.15 * 0.06 * 0.03 m

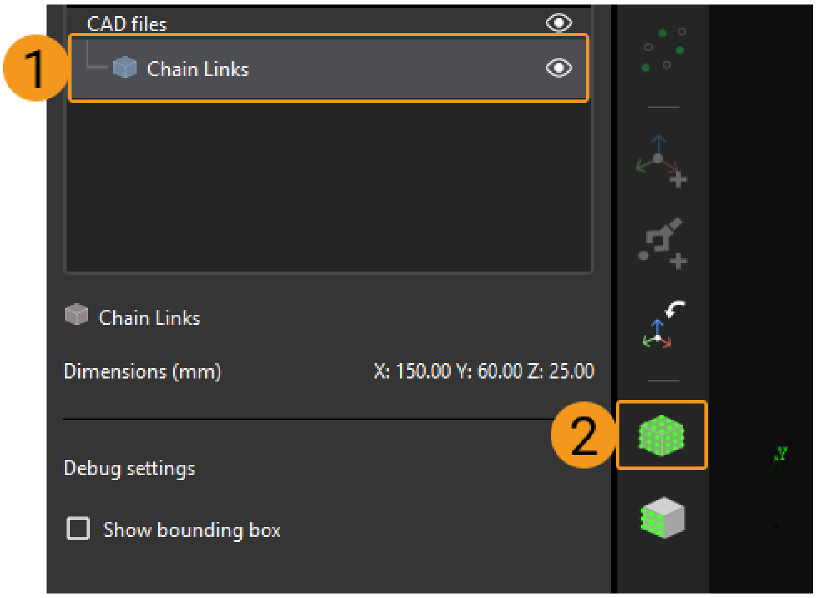
○ 0.15 * 0.06 * 0.03 mm

After the CAD file is imported, it will be displayed in the visualization area of the Matching Model and Pick Point Editor interface.



3. Use the CAD file to make the point cloud model.

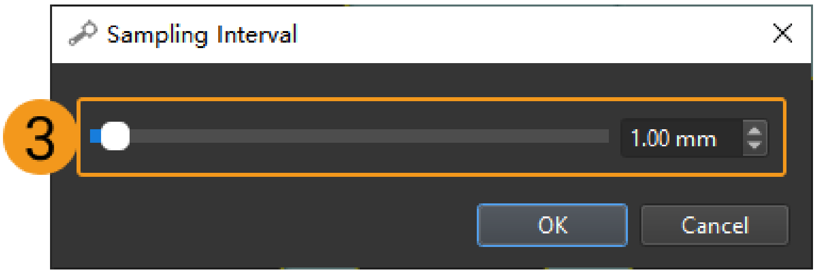
   On the left resource list of the Matching Model and Pick Point Editor interface, select the CAD file, and click the 🟩 button on the toolbar. Then, set the sampling interval on the prompted window to generate the point cloud outside the exterior surface of the CAD model.



Chain Links

Dimensions (mm) X: 150.00 Y: 60.00 Z: 25.00

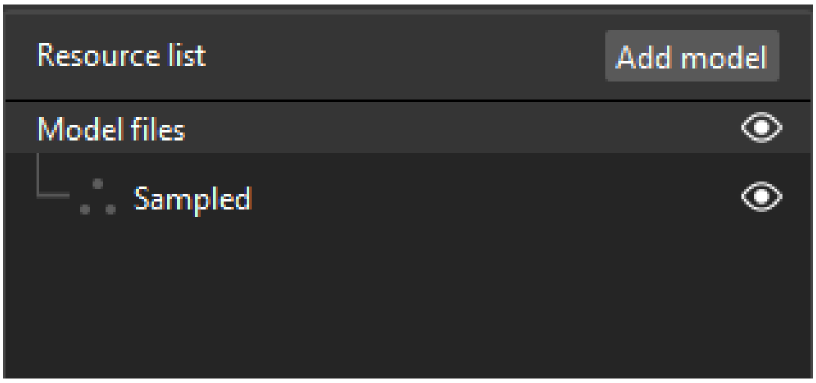
□ Show bounding box



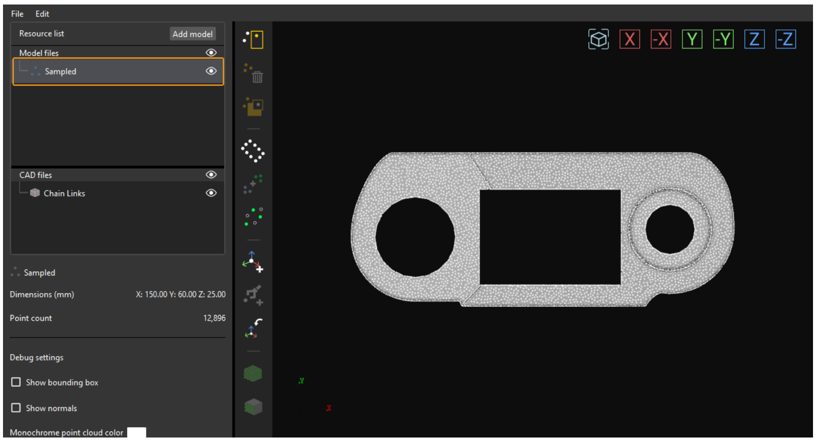
1.00 mm

4. View the generated point cloud model.

   The point cloud model generated based on the CAD file will be displayed in the resource list.



   Select the point cloud model file and you will see the point cloud model in the visualization area of the Matching Model and Pick Point Editor interface.



Sampled

Dimensions (mm) X: 150.00 Y: 60.00 Z: 25.00

Point count 12,896

Debug settings

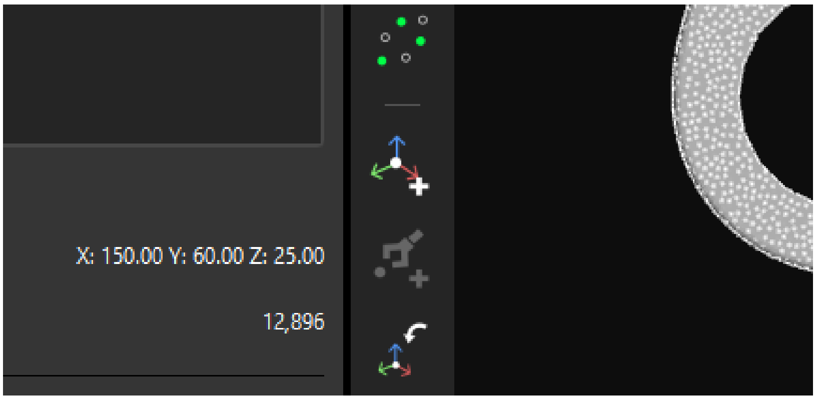
□ Show bounding box

□ Show normals

Monochrome point cloud color

5. Add a pick point.

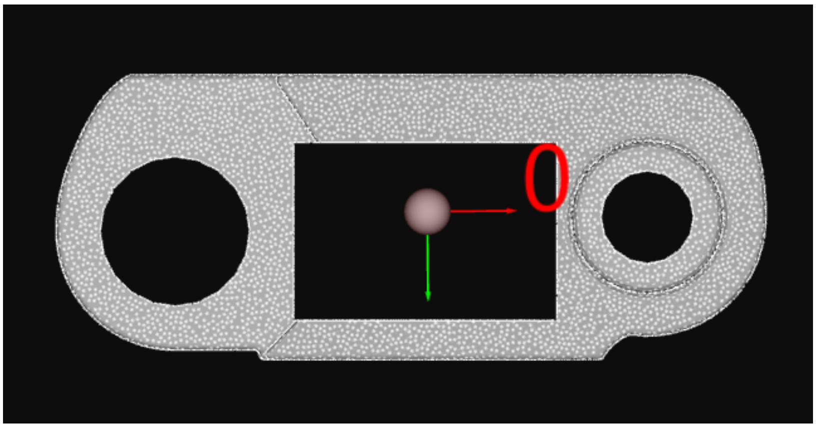
   Click the button on the toolbar to add a pose as a pick point to the point cloud model of the workpiece.



X: 150.00 Y: 60.00 Z: 25.00

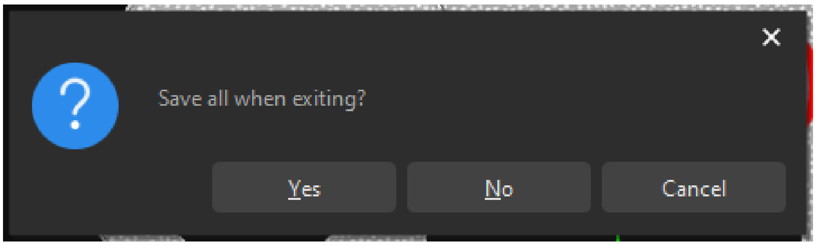
12,896

The following figure shows the added pick point.



6. Save the model and the pick point.

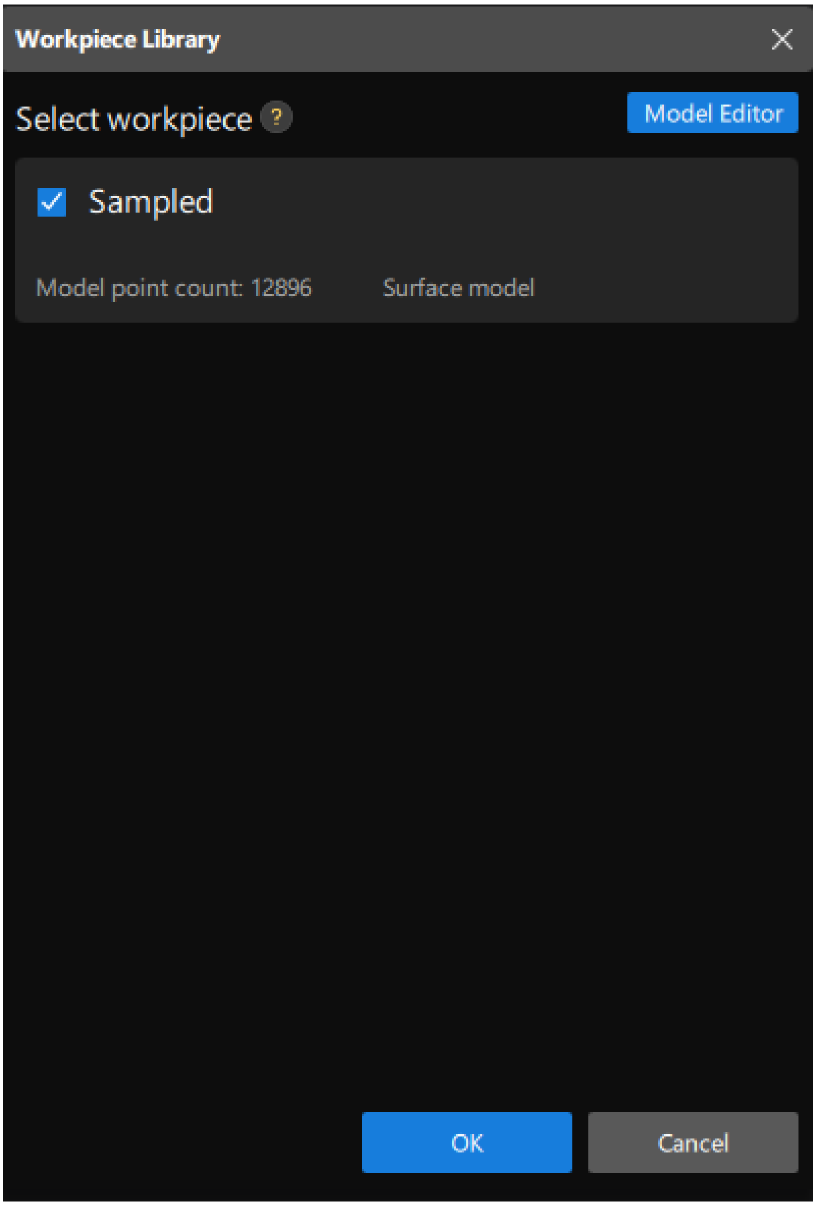
   Close the Matching Model and Pick Point Editor, and click **[Yes]** in the pop-up window.



Save all when exiting?

7. Select this workpiece from the Workpiece Library.

   After closing the Matching Model and Pick Point Editor, select the saved point cloud model of the workpiece, and click [OK].



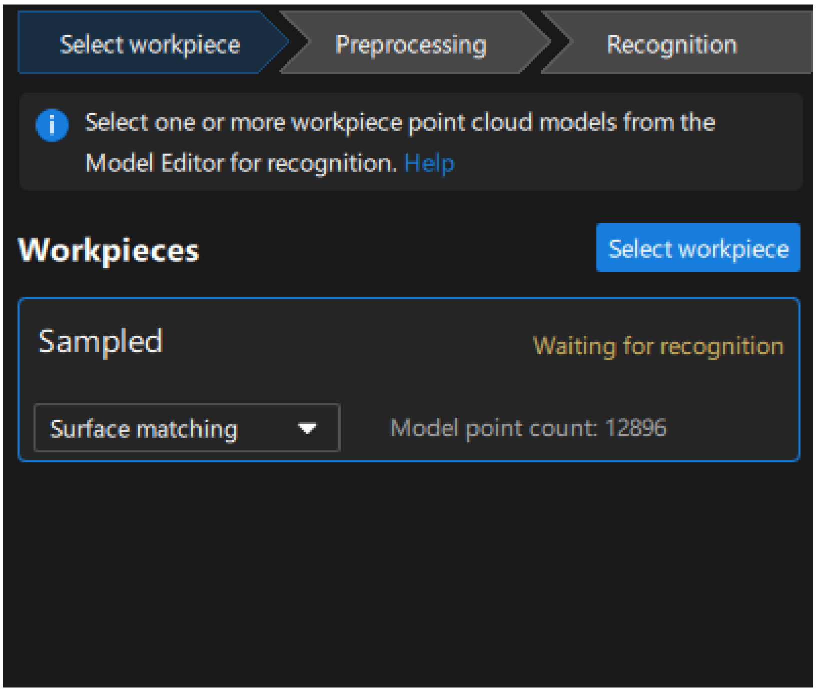
Workpiece Library

Select workpiece

☑ Sampled

Model point count: 12896 Surface model

Subsequently, the target workpiece to recognize is displayed in the upper-right corner of the 3D Workpiece Recognition visualized configurator.



Select one or more workpiece point cloud models from the Model Editor for recognition. Help

Workpieces

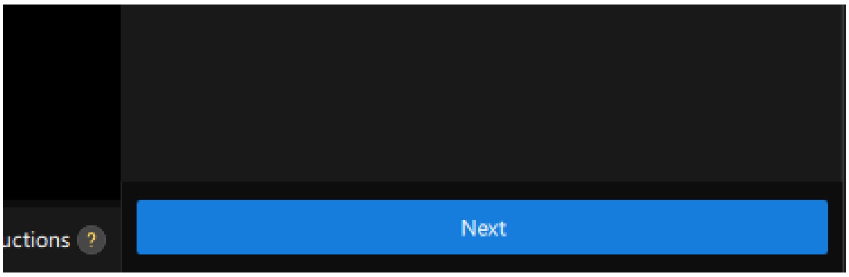
Sampled

Waiting for recognition

Surface matching

Model point count: 12896

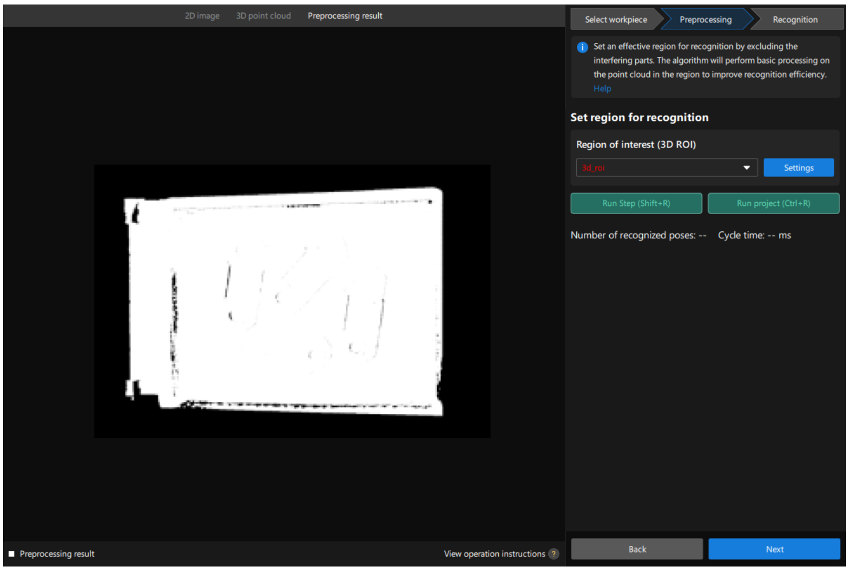
Now, you have selected the workpiece. Click **[Next]** on the bottom of the 3D Workpiece Recognition visualized configurator.



Preprocessing

Preprocessing is used to set an effective region for recognition to exclude the point cloud of unnecessary parts and keep only the point cloud of the workpiece, thus improving recognition efficiency.

The following figure displays the Preprocessing interface.

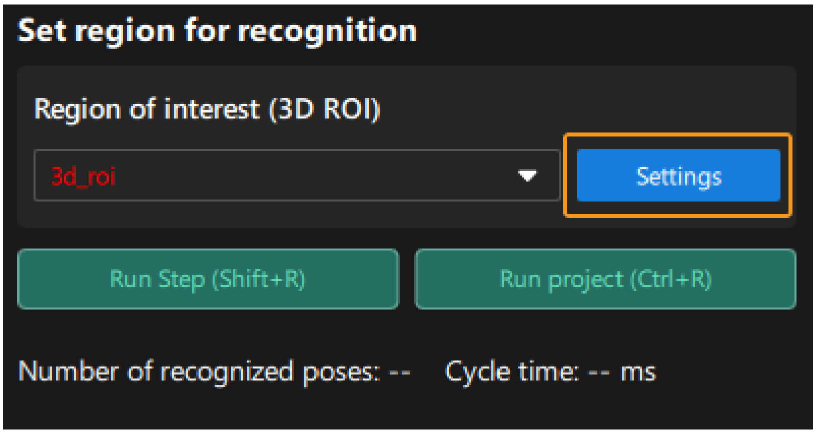


Set an effective region for recognition by excluding the interfering parts. The algorithm will perform basic processing on the point cloud in the region to improve recognition efficiency.

Number of recognized poses: -- Cycle time: -- ms

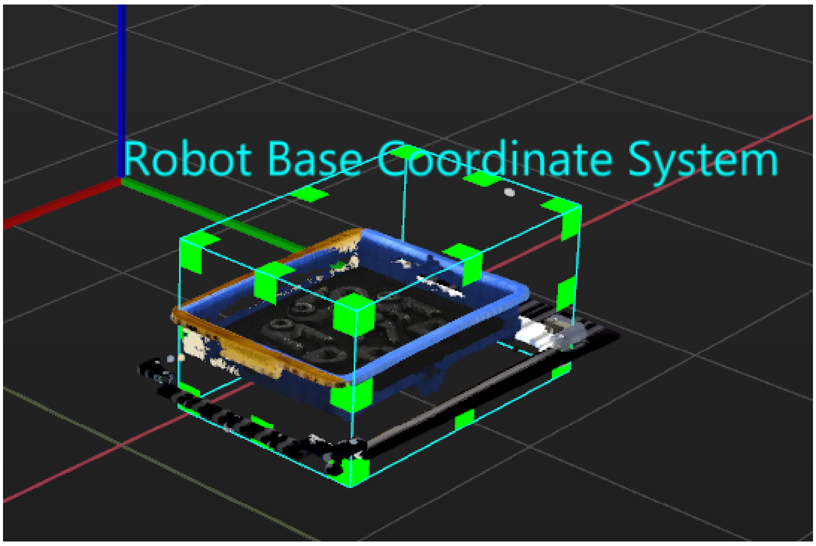
1. Set the region for recognition.

   Click **[Settings]**.

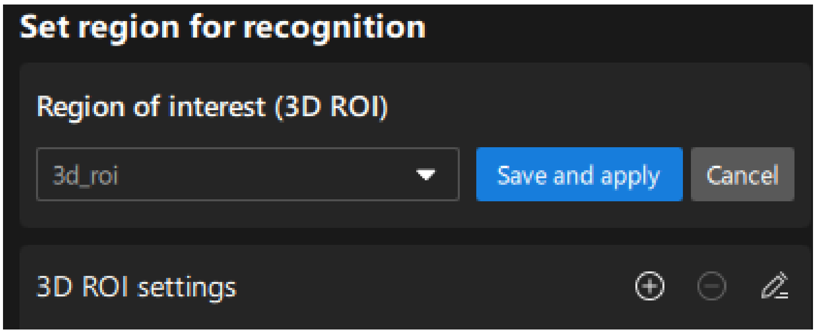
   Number of recognized poses: -- Cycle time: -- ms

   In visualized interface, set the region for recognition (3D ROI). Press and hold the `Ctrl` key, select the vertices of the 3D ROI, and drag the 3D ROI to the proper size. The following figure displays the set 3D ROI.

2. Save the region for recognition.

   Click **[Save and apply]** to save the region for recognition.



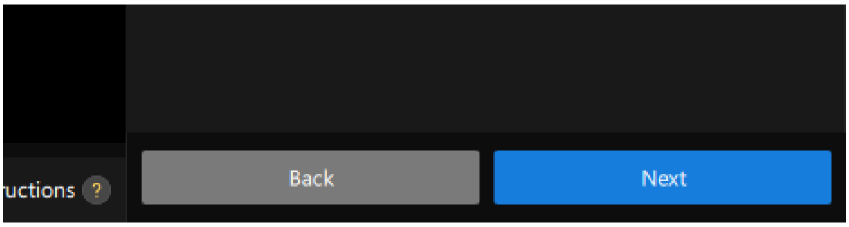
Set region for recognition

Region of interest (3D ROI)

3d_roi

3D ROI settings

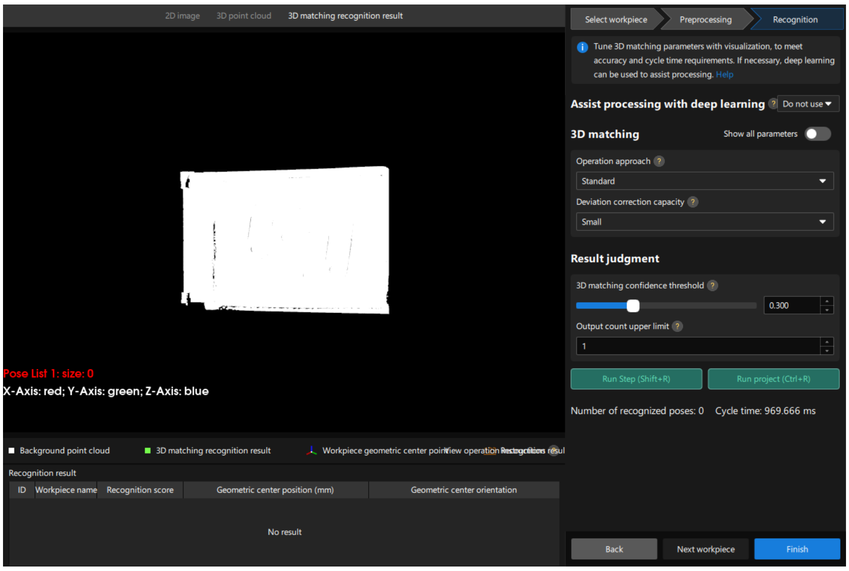
Now, you have finished the preprocessing procedure. Click **[Next]** on the bottom of the 3D Workpiece Recognition visualized configurator to enter the recognition procedure.



Recognize workpieces

In this procedure, you can adjust the 3D matching related parameters in a visualized manner, and output the workpieces' poses.

The following figure displays the Recognition interface.



Tune 3D matching parameters with visualization, to meet accuracy and cycle time requirements. If necessary, deep learning can be used to assist processing. Help

Assist processing with deep learning: Do not use

3D matching

Operation approach: Standard

Deviation correction capacity: Small

Result judgment

3D matching confidence threshold: 0.300

Output count upper limit: 1

Number of recognized poses: 0 Cycle time: 969.666 ms

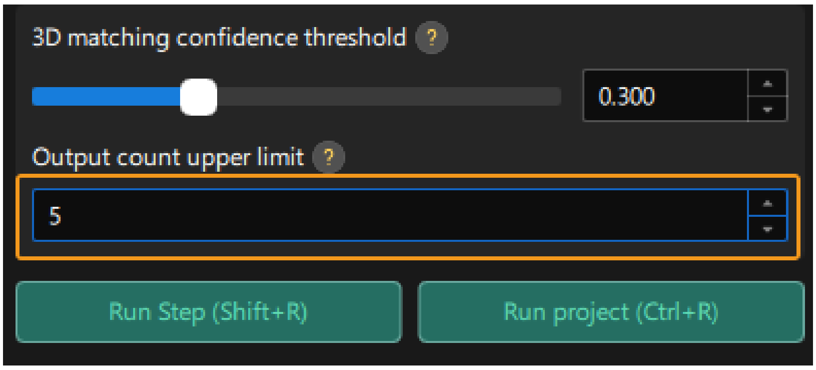
Pose List 1: size: 0

X-Axis: red; Y-Axis: green; Z-Axis: blue

Recognition result

ID	Workpiece name	Recognition score	Geometric center position (mm)	Geometric center orientation
No result				

1. Since this project needs to recognize a maximum of five workpieces, set the **Output count upper limit** to 5.



3D matching confidence threshold

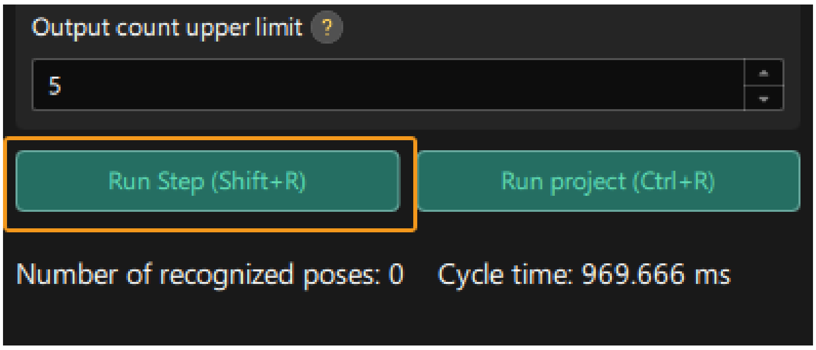
0.300

Output count upper limit

5

2. View the visualized output result.

   Click [Run Step].

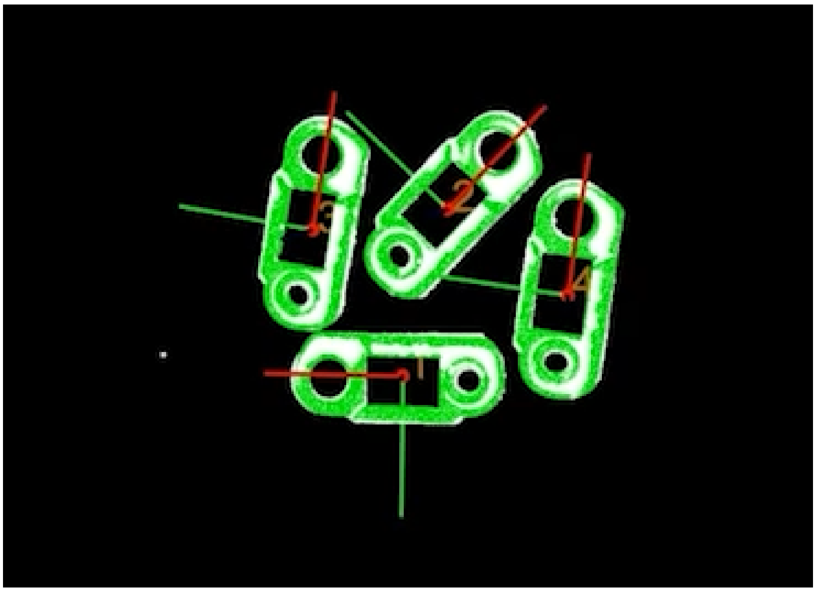


Output count upper limit

5

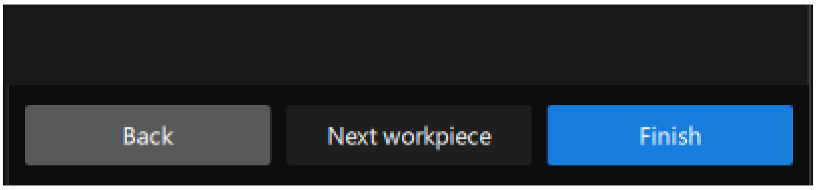
Number of recognized poses: 0 Cycle time: 969.666 ms

   You can view the visualized output result in the visualized area. As the figure below, the poses of four workpieces are output.

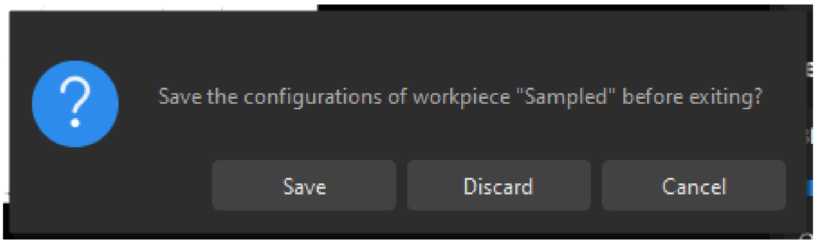


3. Save the configuration.

   Click the [Finish] button on the bottom of the 3D Workpiece Recognition visualized configurator.



In the pop-up window, click [**Save**].



Save the configurations of workpiece "Sampled" before exiting?

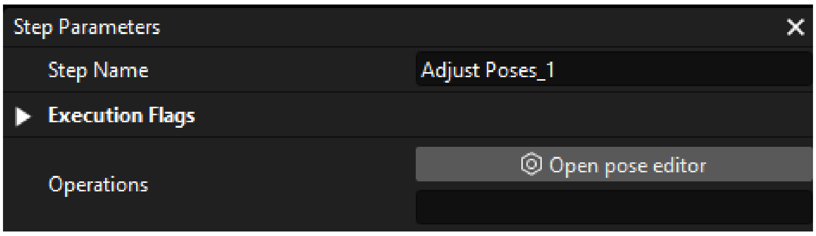
Now, you have recognized the workpiece and calculated its pick point.

Adjust poses

The pick points output by the “3D Workpiece Recognition” Step is in the camera reference frame. To facilitate robot picking, you need to adjust the workpieces’ poses to transform them from the camera reference frame to the robot reference frame.

1. Open the pose editor.

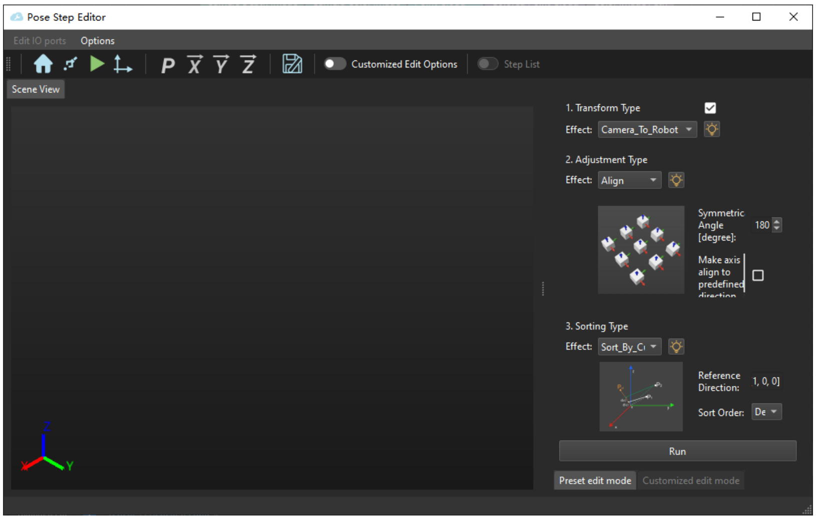
   Select the “Adjust Poses” Step, and click [**Open pose editor**] on the Step Parameters tab.



Step Parameters

Step Name	Adjust Poses_1
▶ Execution Flags	
Operations	Open pose editor

The following figure shows the pose editor.



Pose Step Editor

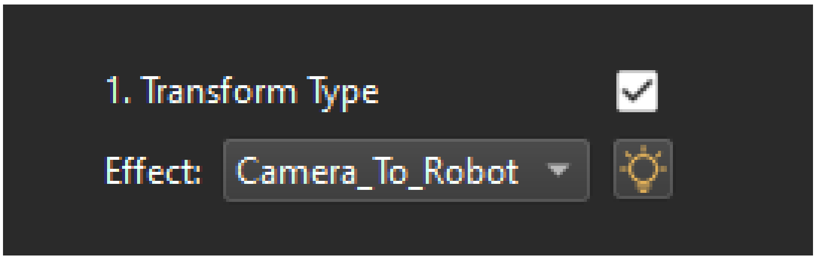
1. Transform Type ☑
Effect: Camera_To_Robot

2. Adjustment Type
Effect: Align
Symmetric Angle [degree]: 180
Make axis align to predefined direction ☐

3. Sorting Type
Effect: Sort_By_C
Reference Direction: 1, 0, 0]
Sort Order: De

2. Adjust the reference frame.

In the upper-left corner of the pose editor, set Effect to **Camera_To_Robot**.

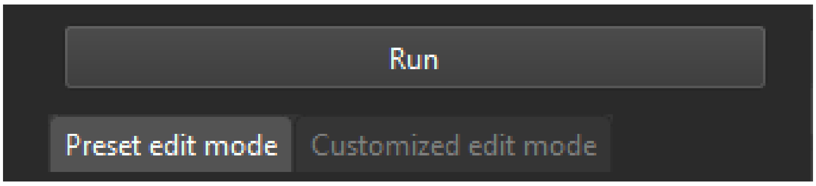


1. Transform Type ☑

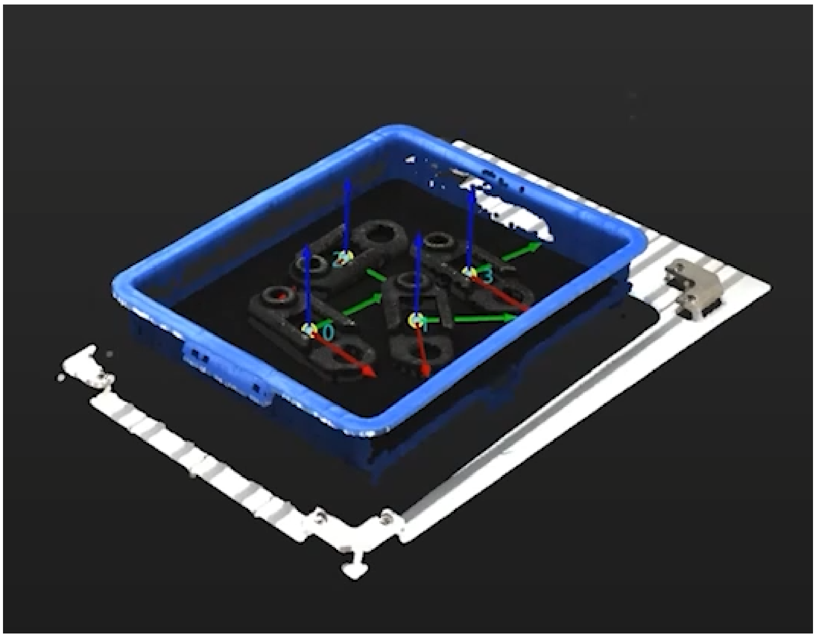
Effect: Camera_To_Robot

3. View the reference frame transformation result.

   On the bottom of the pose editor, click the **[Run]** button.

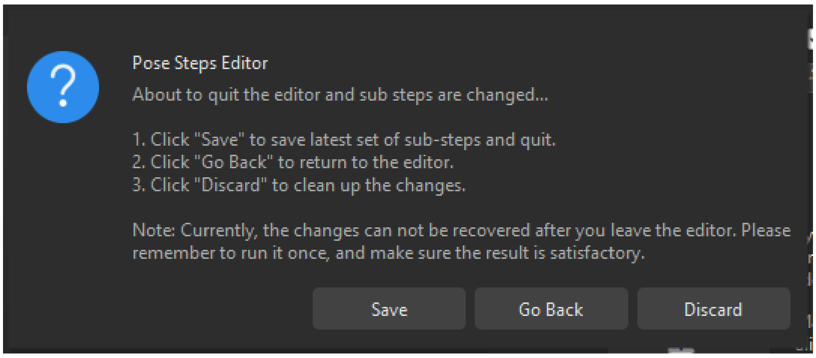


   You can see the transformed pick points in the Scene Viewer of the pose editor.



4. Save the configuration.

   Close the pose editor, and click **[Save]** in the pop-up window.



Pose Steps Editor

About to quit the editor and sub steps are changed...

1. Click "Save" to save latest set of sub-steps and quit.
2. Click "Go Back" to return to the editor.
3. Click "Discard" to clean up the changes.

Note: Currently, the changes can not be recovered after you leave the editor. Please remember to run it once, and make sure the result is satisfactory.

Now, the reference frame of the pick points has been transformed.

Procedure Out

The “Procedure Out” Step sends the results of the current project to the backend service.

Up to now, you have deployed the Mech-Vision project.

2.5. Pick and Place

After having obtained the poses of workpieces using a Mech-Vision case project, you can then build a workflow for the Mech-Viz project to guide the robot to repeatedly pick and place workpieces.

In this tutorial, the model file of the end tool in the OBJ format is required for collision detection. Please prepare the OBJ model file of the end tool before using this tutorial.

▶ https://www.youtube.com/watch?v=at95dOGlRcg/PLVcMd7cW2rXVtrAejMyVQni2dUDv8bxje *(YouTube video)*

Video tutorial: Perform Picking

The following figure shows the workflow to build.

Configure robot and scene
Build workflow
Simulate and run

Configure the Robot and Scene

To avoid collision with surrounding objects during robot picking and placing, you should add the tool model, scene model, and bin models to the project for collision detection.

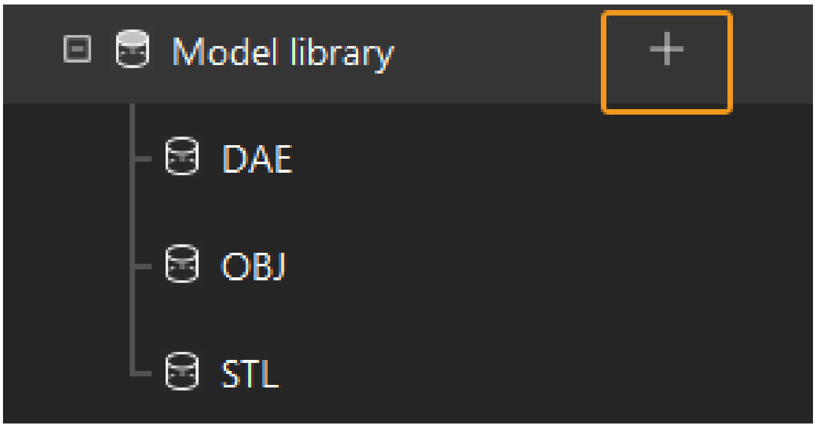
Import and Configure the Tool Model

Tool refers to the device mounted on the robot end that performs processing/picking jobs.

The end tool should be imported and configured so that its model can be displayed in the 3D simulation space and used for collision detection.

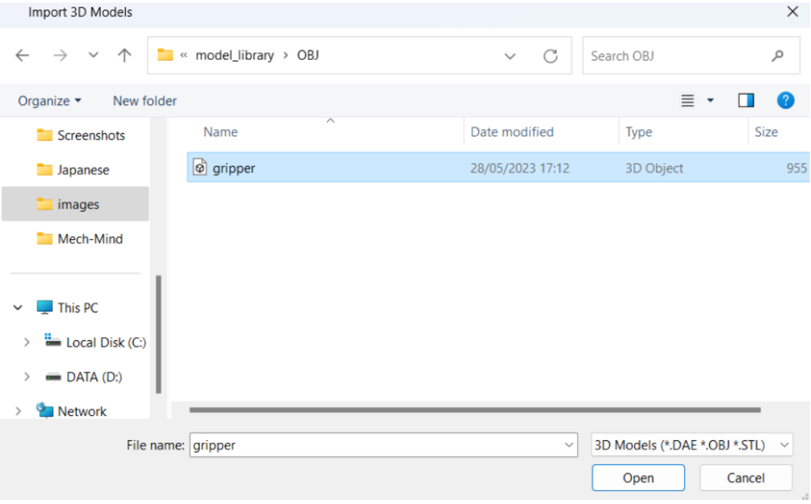
Import the Tool Model

1. Click [+] in Resources › Model Library.



Project resources refer to various fundamental resources used in the project, including the robot, tools, workobjects, and scene objects.

2. In the prompted window, select the collision model file in .obj format, and then click [Open].

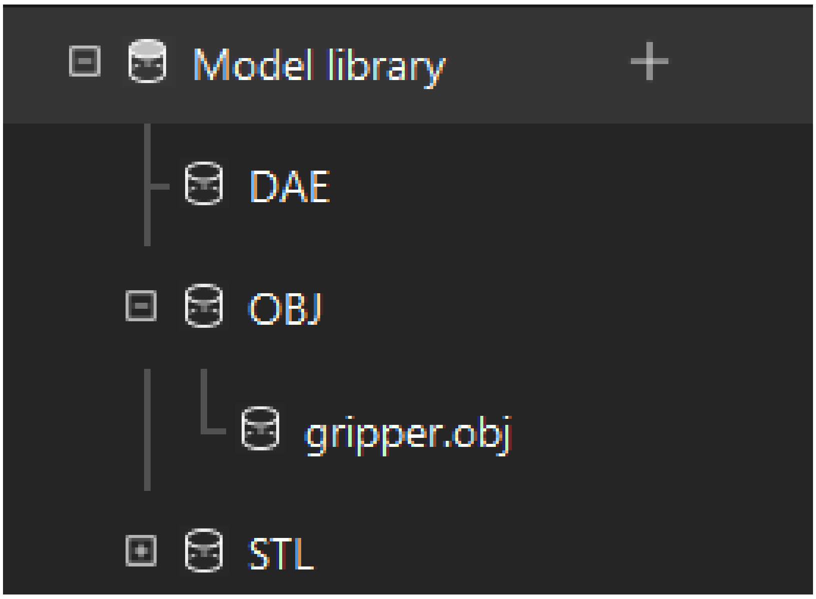


Name	Date modified	Type	Size
gripper	28/05/2023 17:12	3D Object	955

File name: gripper

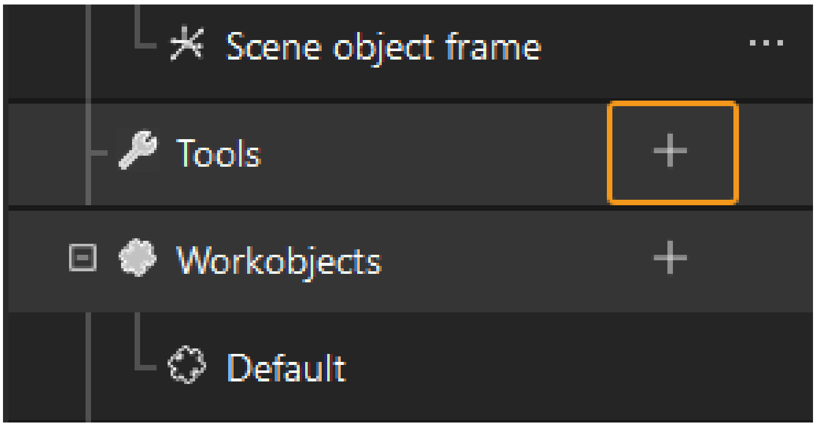
3D Models (*.DAE *.OBJ *.STL)

3. After the model is imported, you can see the imported model in the model library.

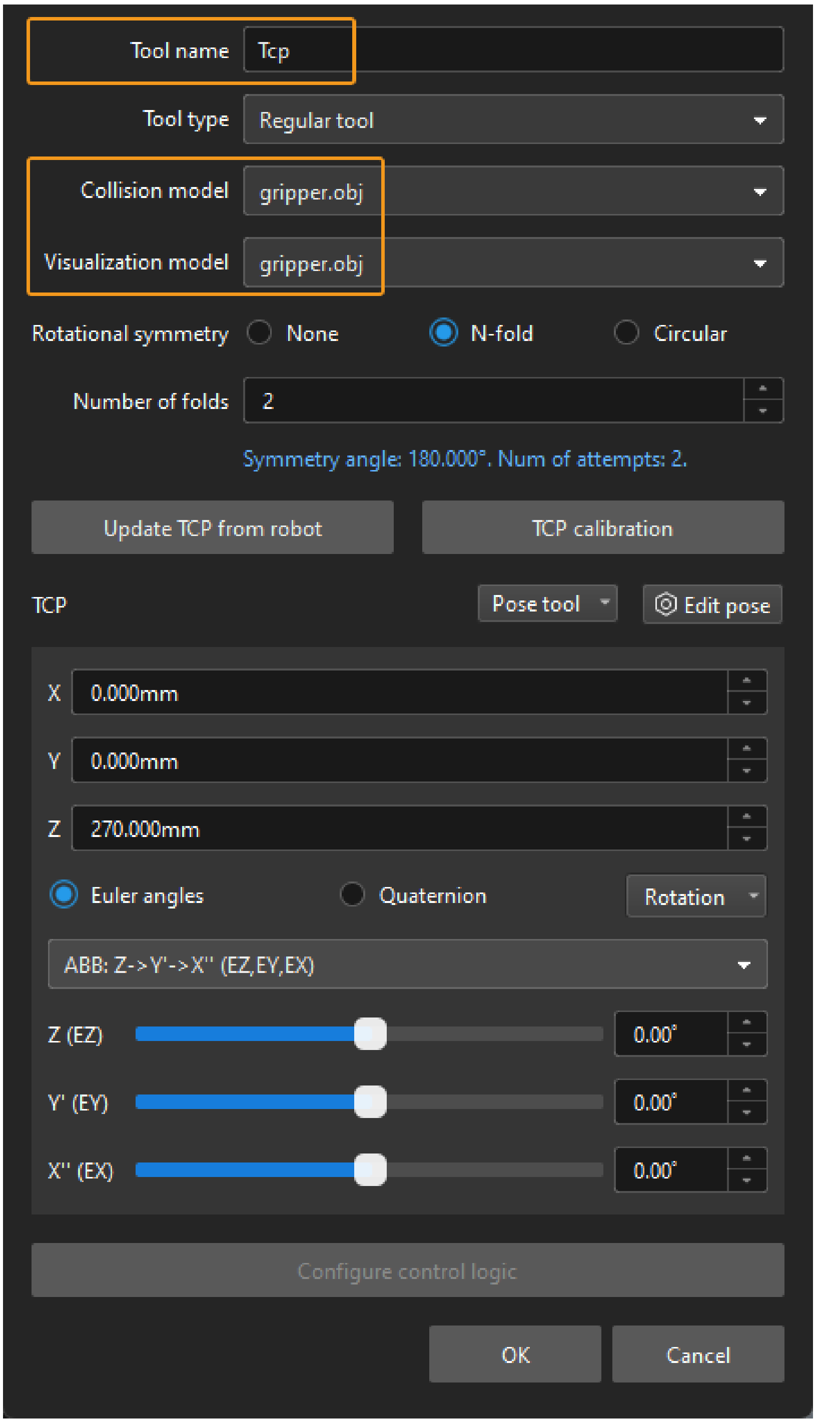


Configure the Tool

1. Click [+] in Resources › Tools.



2. In the prompted window, enter the tool name, select the imported tool model as the collision model and visualization model, and then click [OK].



Tool name: Tcp

Tool type: Regular tool

Collision model: gripper.obj

Visualization model: gripper.obj

Rotational symmetry: ○ None ● N-fold ○ Circular

Number of folds: 2

Symmetry angle: 180.000°. Num of attempts: 2.

TCP

X: 0.000mm

Y: 0.000mm

Z: 270.000mm

● Euler angles ○ Quaternion

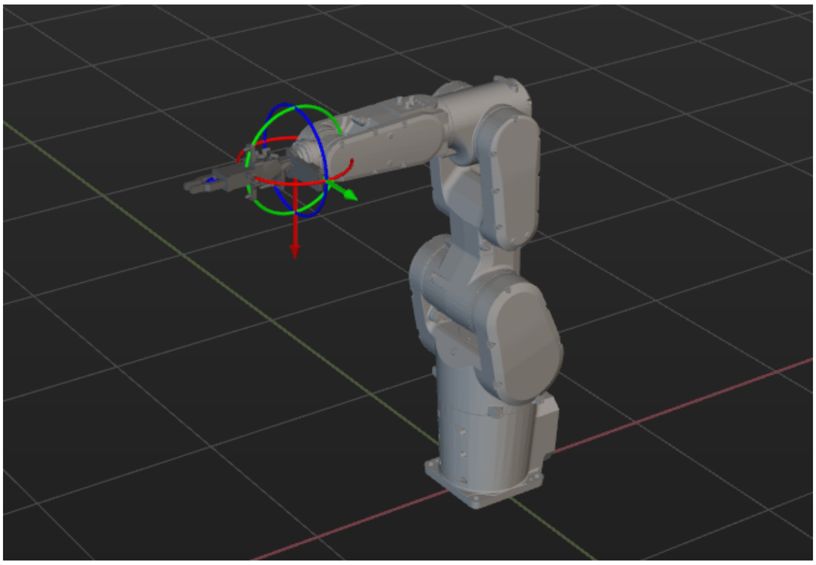
ABB: Z->Y'->X'' (EZ,EY,EX)

Z (EZ): 0.00°

Y' (EY): 0.00°

X'' (EX): 0.00°

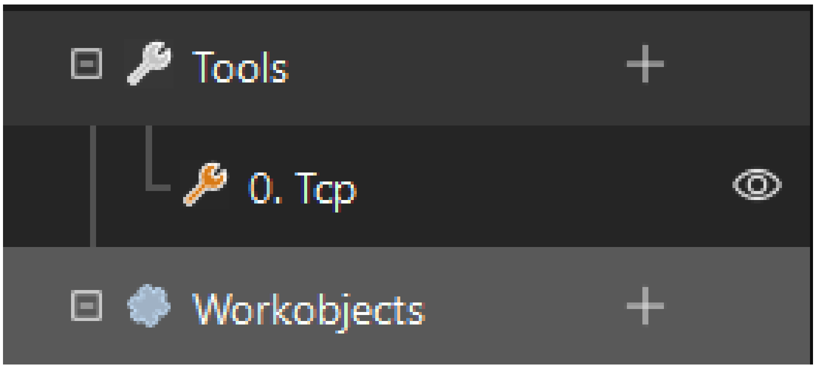
3. After the tool is configured, you can view the configured tool in the 3D simulation space, as shown below.



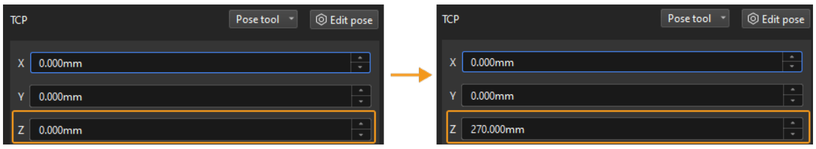
Adjust the Tool

As shown above, the position of the Tool Center Point (TCP) in relation to the robot is obviously wrong. So, you should adjust the position of the TCP so that it is at the tip of the tool.

1. Double-click the tool model in the model library.



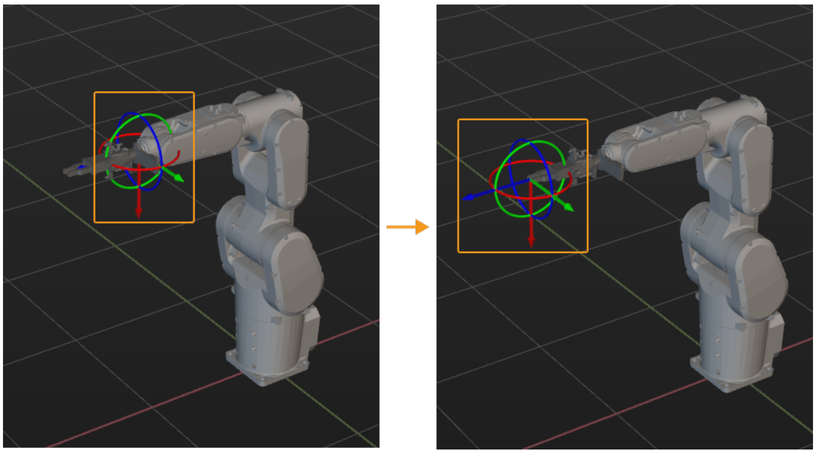
2. In the prompted windows, adjust the settings as shown below.



TCP
X 0.000mm
Y 0.000mm
Z 0.000mm

TCP
X 0.000mm
Y 0.000mm
Z 270.000mm

3. The poses of the tool before and after fitting are shown below.

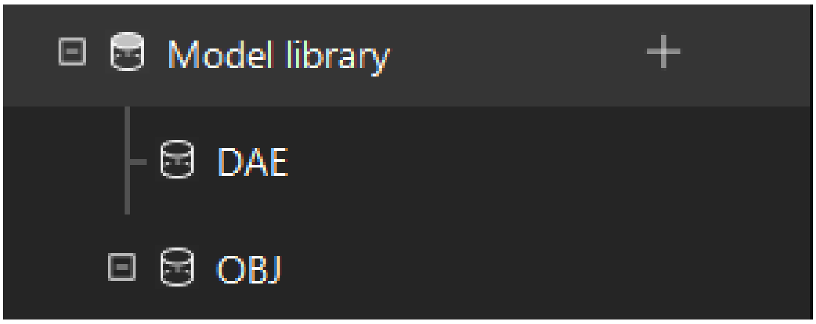


Import and Configure the Scene Model

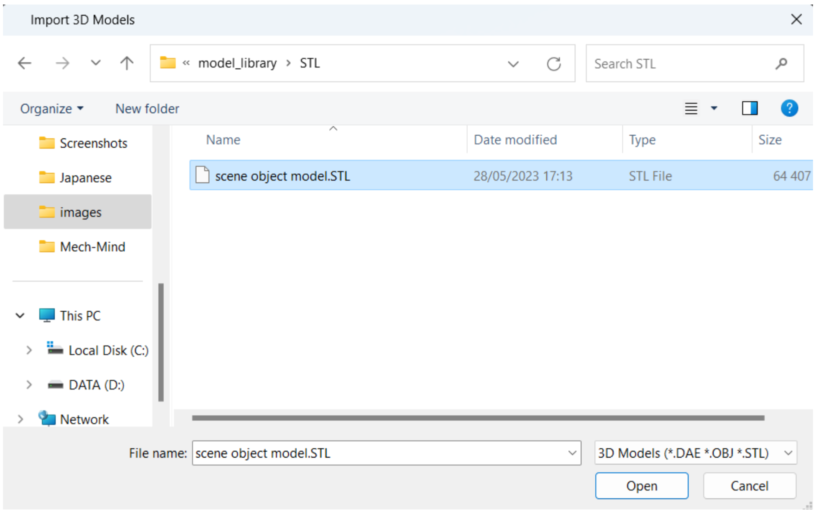
Scene objects are imported and configured to make the scene in the software closer to the real scenario, which facilitates the planning of the collision-free robot motion path.

Import the Scene Model

1. Click [+] in Resources › Model Library.



2. In the prompted window, select the scene object model file, and then click [Open].



Import 3D Models

« model_library › STL

Search STL

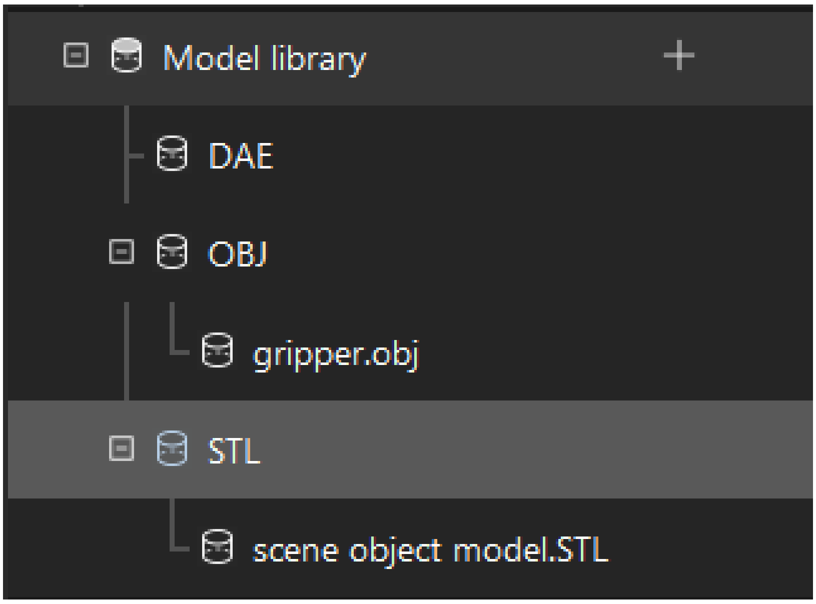
Organize ▾ New folder

Name	Date modified	Type	Size
scene object model.STL	28/05/2023 17:13	STL File	64 407

File name: scene object model.STL

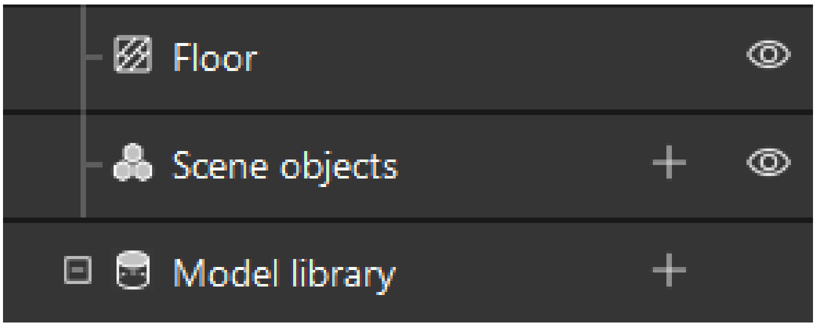
3D Models (*.DAE *.OBJ *.STL)

3. After the model is imported, you can see the imported model in the model library.

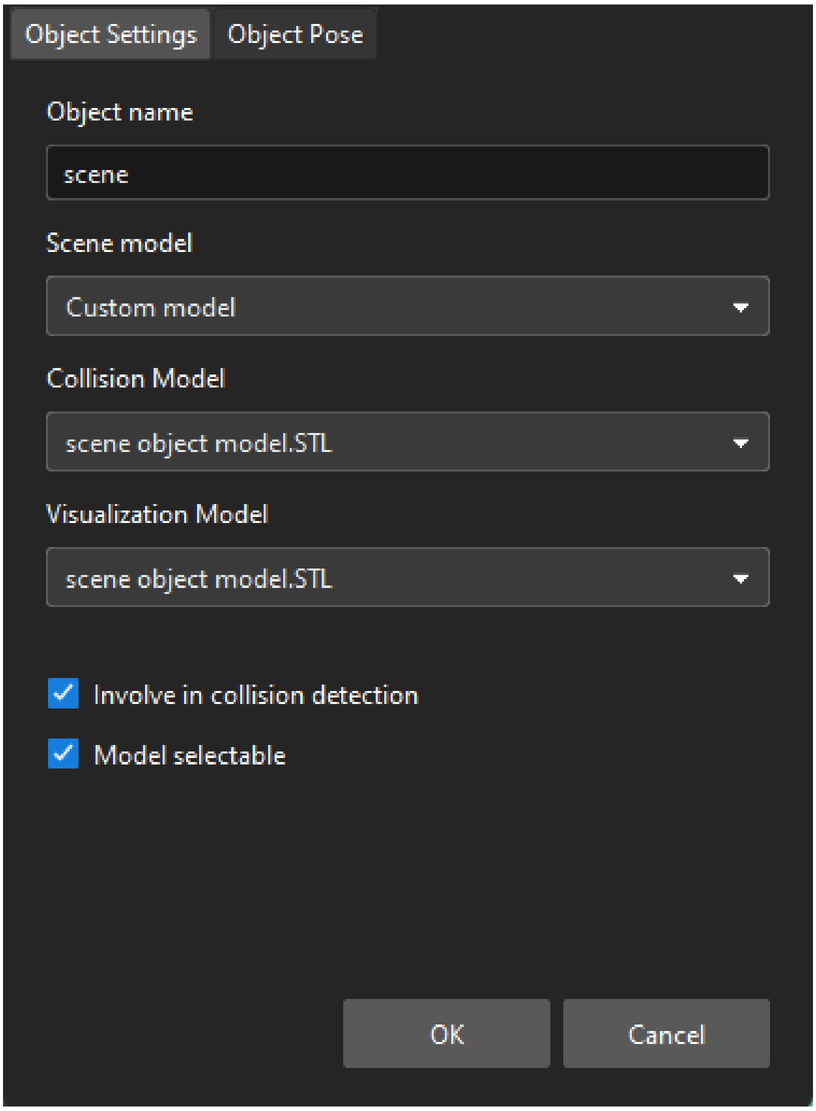


Configure the Scene Model

1. Click [+] in Resources › Scene Objects.



2. In the prompted window, enter the object name, select “Custom model” as the “Scene model”, select the imported scene object model as the collision model and visualization model, and then click [OK].



Object name

scene

Scene model

Custom model

Collision Model

scene object model.STL

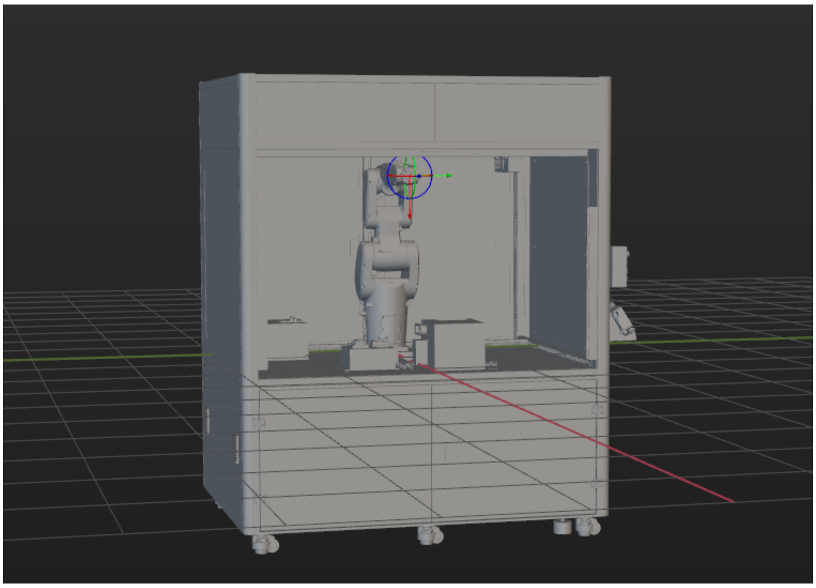
Visualization Model

scene object model.STL

☑ Involve in collision detection

☑ Model selectable

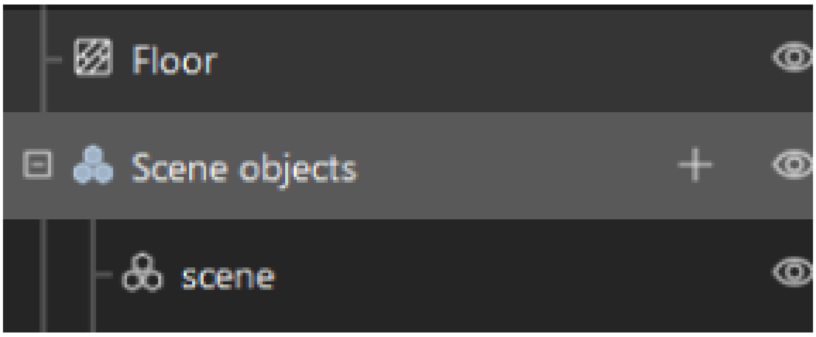
3. After configuration, the scene object is displayed in the 3D simulation space.



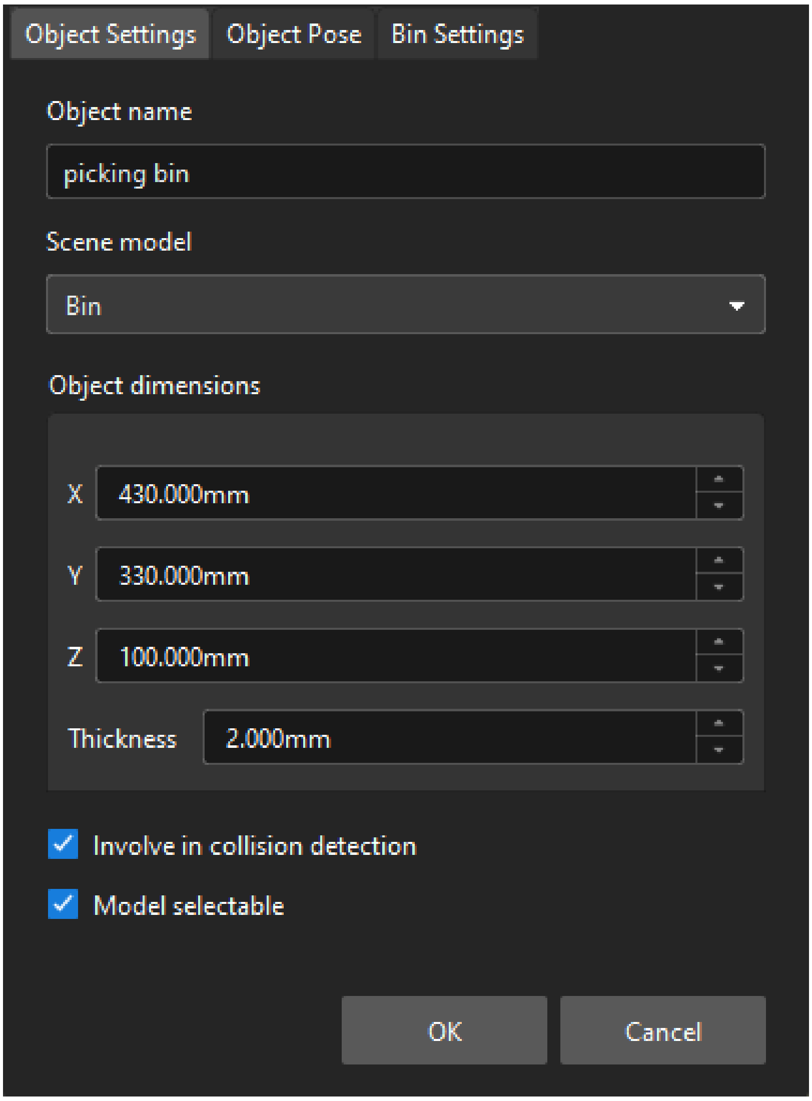
Add and Configure the Bin Models

1. Add the “picking bin” model.

   Click [+] in Resources › Scene Objects.



   In the prompted window, set the “Object name” parameter to “picking bin”, and the “Scene model” parameter to “Bin”. Then set the “Object dimensions” parameter according to the measured size of the real bin, and then click [OK].



Object name: picking bin

Scene model: Bin

Object dimensions

X: 430.000mm

Y: 330.000mm

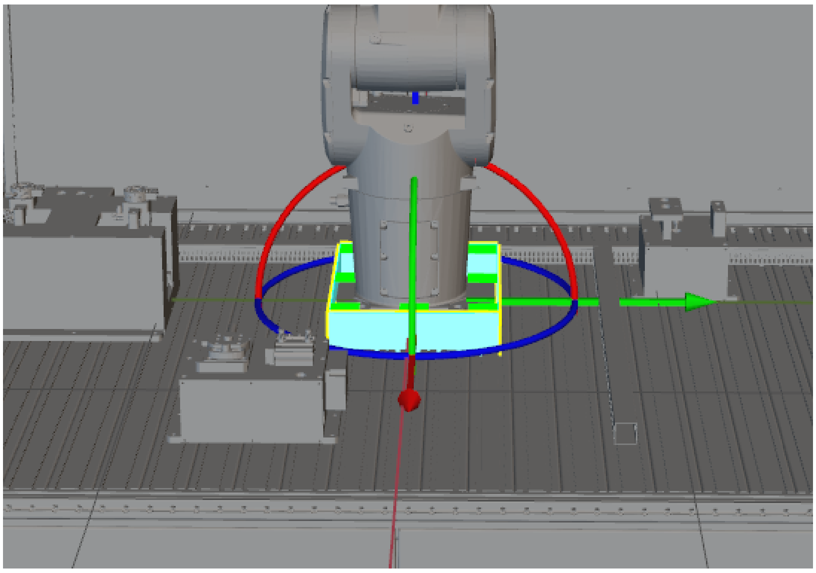
Z: 100.000mm

Thickness: 2.000mm

☑ Involve in collision detection

☑ Model selectable

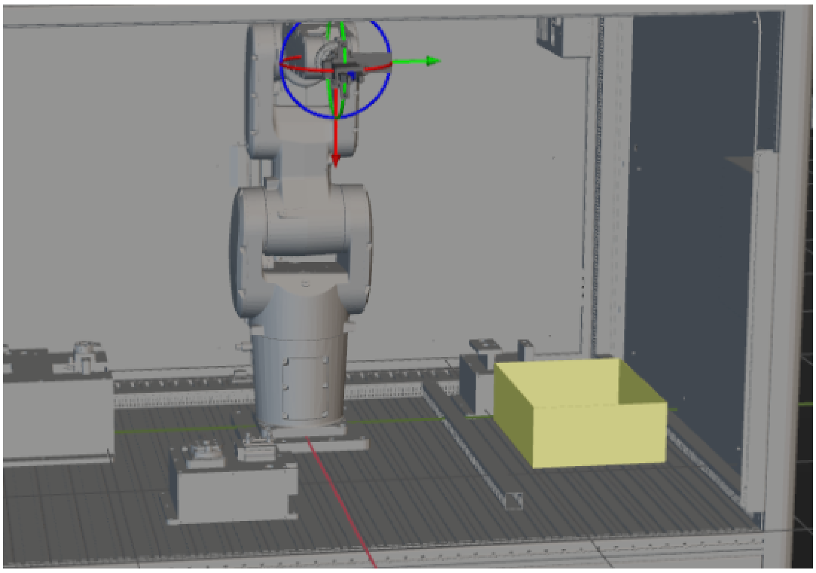
   After configuration, the “picking bin” model is displayed in the 3D simulation space.



2. Adjust the “picking bin” model.

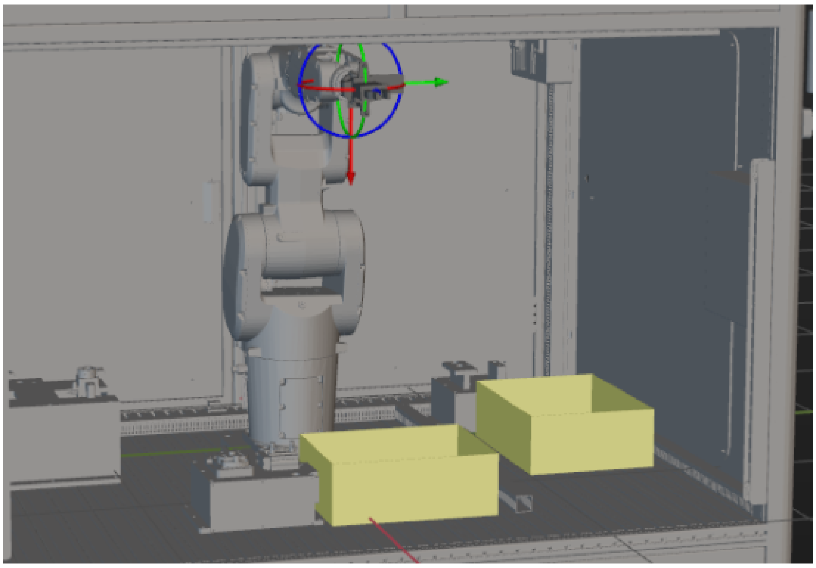
   To determine the position of the real bin, you can run the Mech-Vision project to send the point cloud of the real bin to Mech-Viz. Then, you can adjust the pose of the “picking bin” based on the point cloud position of the real bin.

   Press the `Ctrl` key and press the left mouse button to drag the “picking bin” model to coincide with the point cloud position of the real bin. The position of the “picking bin” model after adjustment is shown below.



3. Add and adjust the “placing bin” model.

   Use the same method to add and adjust the “placing bin” model, as shown below.



So far, you have added all the resources required by the Mech-Viz project.

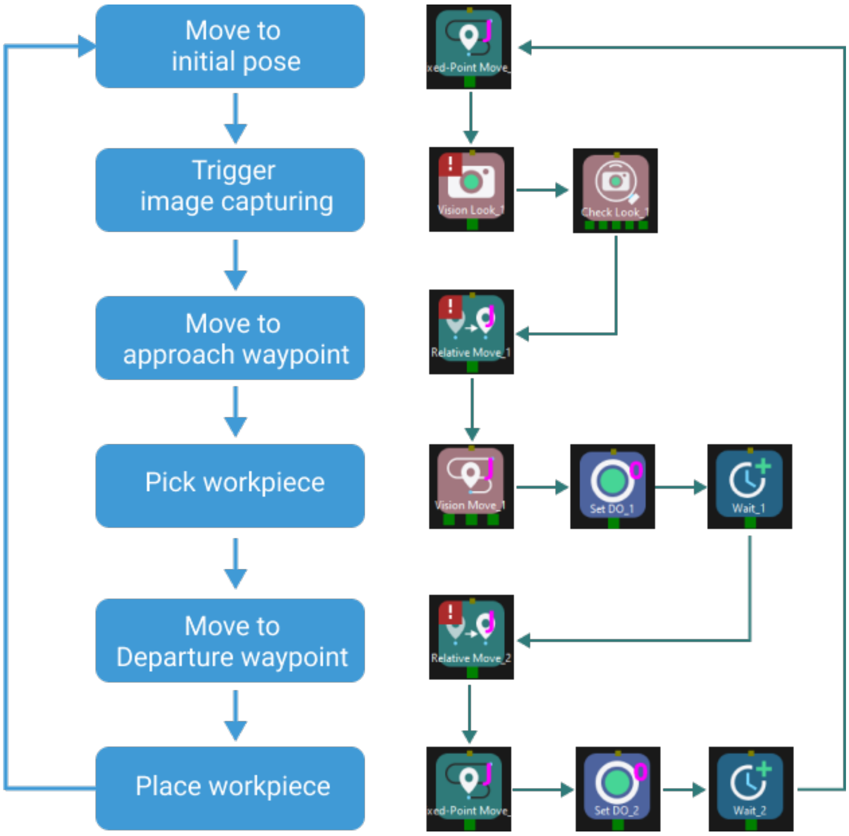
Create a Workflow

Now that you have configured the models, you can start creating a workflow. Drag a step from the step library to the graphical programming workspace, set the parameters of the Steps, and connect Steps to achieve preset program functions.

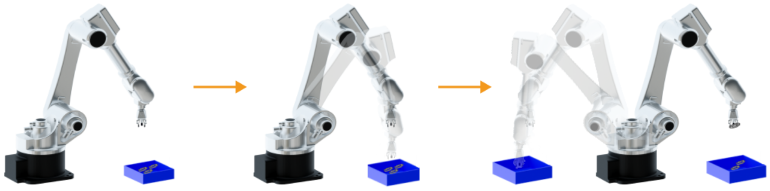
- The workflow refers to the robot motion control program created in Mech-Viz in the form of a flowchart.
- A Step refers to a function module of robot programming.

Introduction to the Project Workflow

The created workflow is shown below.



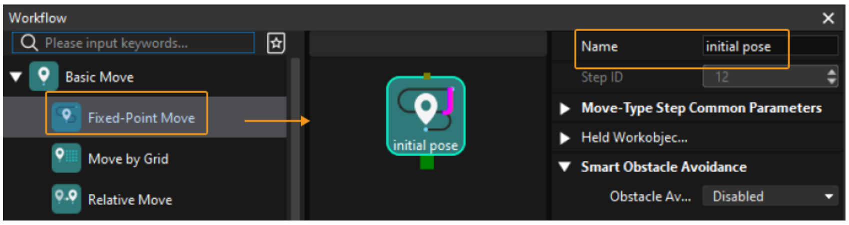
The created workflow can achieve the picking and placing as shown below.



Define the "Home Position"

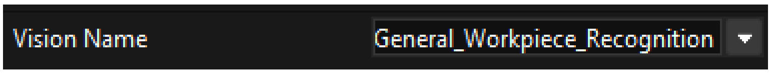
The home position is the starting point of the robot movement and should also be a safe position. At this point, the robot should be away from the objects to pick and the surrounding devices, and should not block the camera's FOV.

After the robot arrives at the self-defined home position, select **Fixed-Point Move** in the Step Library, and drag it to the graphical programming workspace, rename it as "initial pose". Then click **[Sync Robot]** on the toolbar to record the current pose of the robot for this Step.



Name: initial pose
Step ID: 12
Obstacle Av...: Disabled

Trigger the Camera to Capture Images

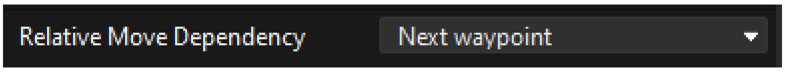
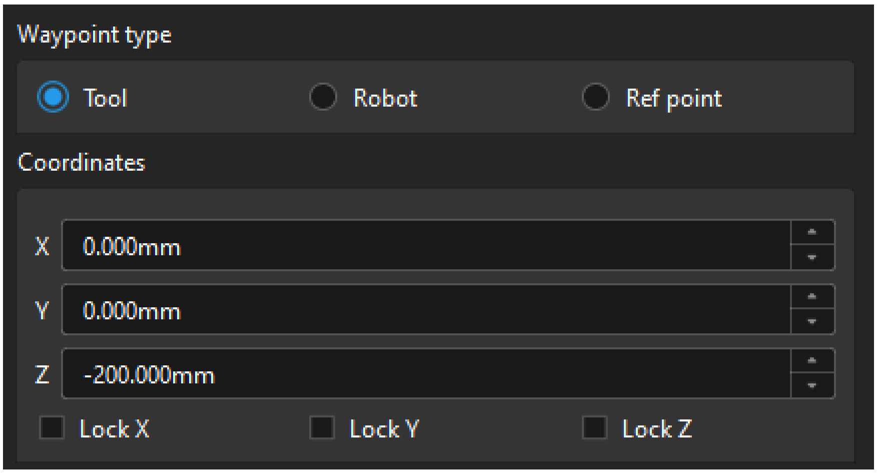
Step	Vision Look
Description	Start the Mech-Vision project to obtain the vision recognition result.
Operation	Find the “Vision Look” Step in the Step Library, and drag it to the graphical programming workspace.
Parameter Settings	Select **General_Workpiece_Recognition** from the **Vision Name** drop-down list.
Illustration	 Vision Name General_Workpiece_Recognition

After the “Vision Look” Step, add the **Check Look** Step to check whether there is any vision result.

Step	Check Look
Description	Check whether there is any vision result.
Operation	Find the “Check Look” Step in the Step Library, and drag it to the graphical programming workspace.
Parameter Settings	Keep the default parameter settings.

Move the Robot to the Approach Waypoint

After obtaining the vision recognition result, you can then use the **Relative Move** step to move the robot to the approach waypoint.

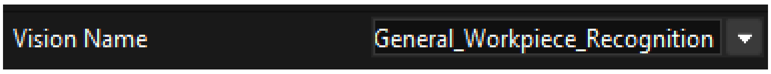
Step	Relative Move
Description	The robot moves according to the vision recognition result.
Operation	Find the “Relative Move” Step in the Step Library, drag it to the graphical programming workspace, and rename it as “approach waypoint”.
Parameter Settings	Set **Relative Move Dependency** to **Next waypoint**; set **Waypoint type** to **Tool**, and set the **Z** coordinate to **-200mm**.
Illustration	 Relative Move Dependency Next waypoint  Waypoint type ● Tool ○ Robot ○ Ref point Coordinates X 0.000mm Y 0.000mm Z -200.000mm □ Lock X □ Lock Y □ Lock Z

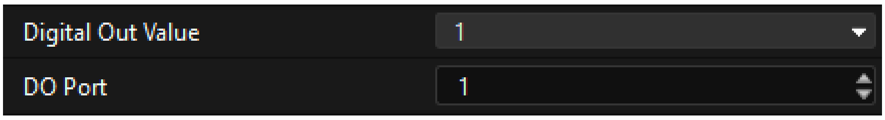
Pick the Workpiece

After the robot has arrived at the approach waypoint, you can control the robot the pick workpieces. The picking process can be divided into two steps.

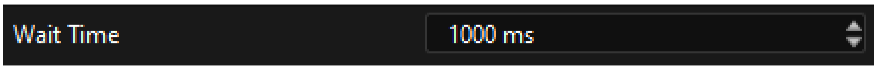
1. Step 1: Use the **Vision Move** Step to control the robot to reach the picking waypoint of the workpiece.
2. Step 2: Use the **Set DO** Step to control the robot to expand the gripper to pick the workpiece.

See the following for details.

Step	Vision Move
Description	The robot moves according to the vision recognition result.
Operation	Find the "Vision Move" Step in the Step Library, and drag it to the graphical programming workspace.
Parameter Settings	Select **General_Workpiece_Recognition** from the **Vision Name** drop-down list.
Illustration	 Vision Name General_Workpiece_Recognition

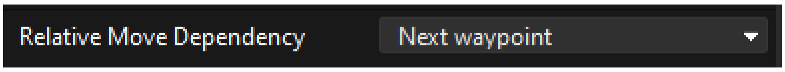
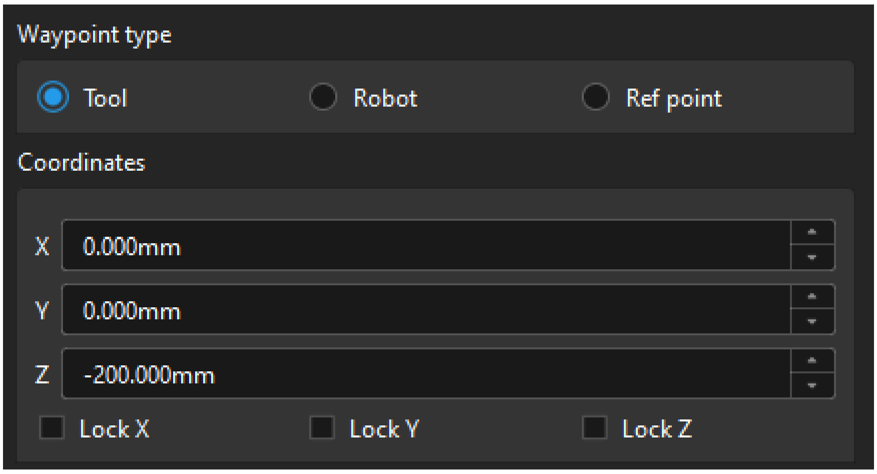
Step	Set DO
Description	Control the robot to expand the gripper to pick the workpiece.
Operation	Find the "Set DO" Step in the Step Library, drag it to the graphical programming workspace, and rename it as "expand the gripper".
Parameter Settings	Set the **Digital Out Value** and **DO Port** parameters to **1**.
Illustration	 Digital Out Value 1 DO Port 1

Since it takes time for the gripper to expand, you should add the **Wait** Step to avoid empty picking by the robot.

Step	Wait
Description	Avoid an empty pick by the robot.
Operation	Find the "Wait" Step in the Step Library, drag it to the graphical programming workspace, and rename it as "Wait for firm gripping".
Parameter Settings	Set the **Wait Time** parameter to **1000ms**.
Illustration	 Wait Time 1000 ms

Move the Robot to the Departure Waypoint

After the robot picks the workpiece, you should add the **Relative Move** Step to control the robot to arrive at the departure waypoint.

Step	Relative Move
Description	The robot moves according to the vision recognition result.
Operation	Find the “Relative Move” Step in the Step Library, drag it to the graphical programming workspace, and rename it as “Departure waypoint”.
Parameter Settings	Set **Relative Move Dependency** to **Next waypoint**; set **Waypoint type** to **Tool**, and set the **Z** coordinate to **-200mm**.
Illustration	 Relative Move Dependency: Next waypoint  Waypoint type ● Tool ○ Robot ○ Ref point Coordinates X 0.000mm Y 0.000mm Z -200.000mm □ Lock X □ Lock Y □ Lock Z

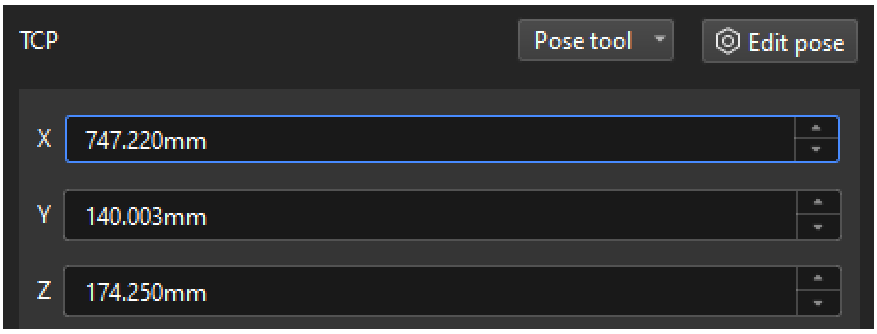
Place the Workpiece

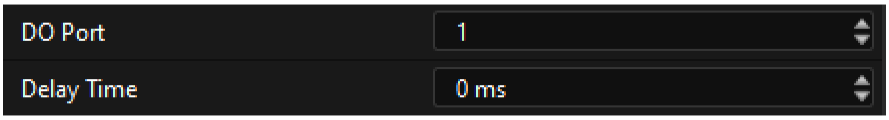
After the robot picks the workpiece, the robot needs to place the workpiece into the placing bin. The placing process can be divided into two steps.

1. Step 1: Use the **Fixed-Point Move** Step to control the robot to arrive at the position of the placing bin.
2. Step 2: Use the **Set DO** Step to control the robot to close the gripper to place the workpiece.

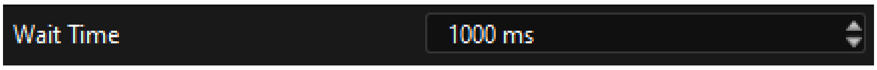
See the following for details.

Step	Fixed-Point Move
Description	Control the robot to arrive at the position of the placing bin.
Operation	Find the “Fixed-Point Move” Step in the Step Library, drag it to the graphical programming workspace, and rename it as “Placing waypoint”.
Parameter Settings	Set a suitable **TCP** as the placing waypoint.

Illustration	 TCP Pose tool Edit pose X 747.220mm Y 140.003mm Z 174.250mm

Step	Set DO
Description	Control the robot to close the gripper to place the workpiece.
Operation	Find the “Set DO” Step in the Step Library, drag it to the graphical programming workspace, and rename it as “retract the gripper”.
Parameter Settings	Set the **Digital Out Value** parameter to **0** and the **DO Port** parameter to **1**.
Illustration	 DO Port 1 Delay Time 0 ms

Since it takes time for the gripper to close, you should add the **Wait** Step to avoid that the robot cannot place the workpiece.

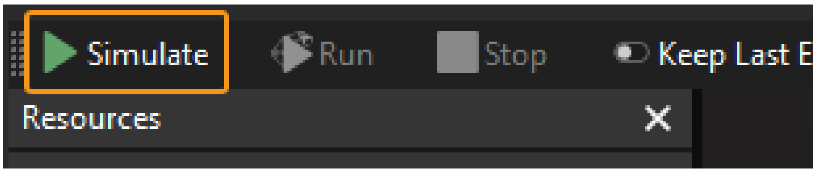
Step	Wait
Description	Avoid the failure of placing the workpiece by the robot.
Operation	Find the “Wait” Step in the Step Library, drag it to the graphical programming workspace, and rename it as “Wait for release”.
Parameter Settings	Set the **Wait Time** parameter to **1000ms**.
Illustration	 Wait Time 1000 ms

Connect Steps

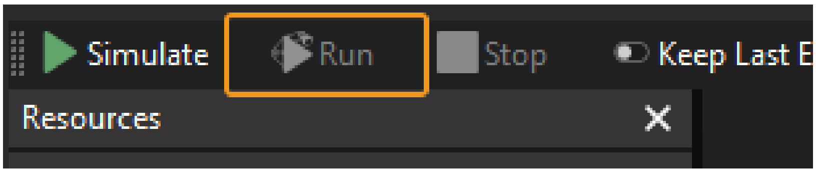
After adding the previous steps, connect them one by one. To control the robot to repeatedly pick up and place workpieces, you should connect the exit port of “Placing Wait” to the entry port of “Home Position”.

Simulate and Run the Project

1. After all the Steps are connected properly, click the **[Simulate]** button on the toolbar to simulate the Mech-Viz project.



2. If the project runs as expected in simulation, click the [Run] button on the Mech-Viz toolbar to run the project with the real robot.



It is recommended that the robot should move at a low speed and that you pay attention to the robot motion trajectory. In an emergency, press the emergency stop key on the teach pendant.

3. Getting Started Tutorial: Vision-Guided Robotic Picking of Metal Parts (Standard Interface Communication)

In this tutorial, you will learn how to deploy a simple 3D vision-guided robotic application of picking small metal parts in the Standard Interface communication mode.

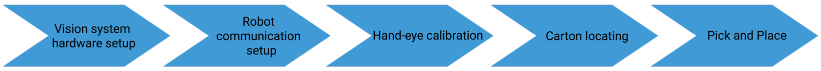
Application Overview

- Camera: Mech-Eye PRO M camera, mounted in eye to hand (ETH) mode
- Robot: YASKAWA_GP8
- Workpiece: track links (made of metal)
- End tool: gripper
- Used software: Mech-Vision 1.7.4, Mech-Viz 1.7.4, Mech-Center 1.7.4, and Mech-Eye Viewer 2.1.0
- Communication mode: Standard Interface communication

If you are using a different camera model, robot brand, or workpiece than in this example, please refer to the reference information provided in the corresponding steps to make adjustments.

How to Deploy a Vision-Guided Robotic Application?

The deployment of the vision-guided robotic application can be divided into five phases, as shown in the figure below:



The following table describes the five phases of deploying a vision-guided robotic application.

No.	Phase	Description
1	Vision system hardware setup	Install and connect hardware of the Mech-Mind Vision System.
2	Robot communication setup	Load the robot Standard Interface program and the configuration files to the robot system and set up the communication between the vision side and the robot.
3	Hand-eye calibration	Perform the automatic hand-eye calibration in the eye-to-hand setup, to establish the transformation relationship between the camera reference frame and the robot reference frame.
4	Workpiece locating	Use the “General Workpiece Recognition” case project to calculate the workpiece poses and output the vision result.

No.	Phase	Description
5	Pick and place	Prepare a robot pick-and-place program that allows the robot to pick and place workpieces under the guidance of the obtained vision result.

Next, follow subsequent sections to complete the application deployment.

3.1. Vision System Hardware Setup

In this tutorial, you will learn how to build the Mech-Mind Vision System.

Follow the steps below to build the Mech-Mind Vision System: **Check the contents of the package** → **Install the hardware** → **Connect the network** → **Upgrade the software (optional)** → **Confirm that the vision system can capture images normally**.

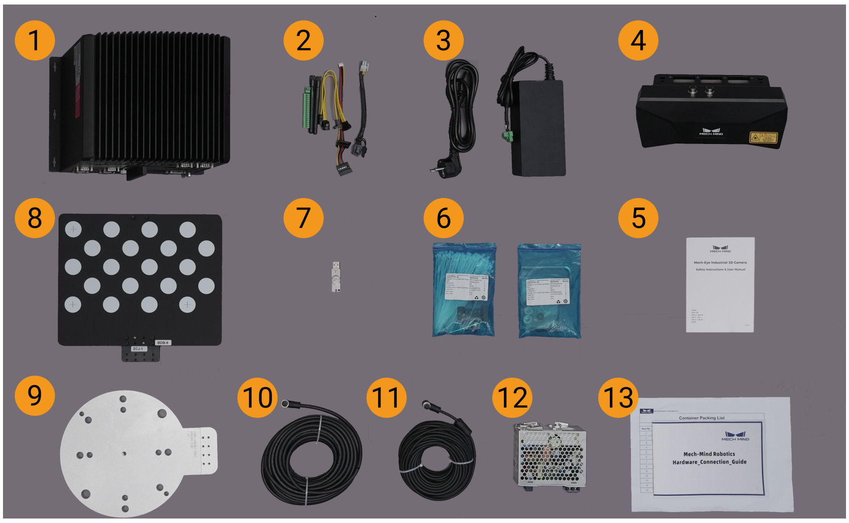
▶ https://www.youtube.com/watch?v=Utnjo0l0UAU/PLVcMd7cW2rXVtrAejMyVQni2dUDv8bxje *(YouTube video)*

Video Tutorial: Vision System Hardware Setup

Check Contents of the Package

1. Make sure that the package is intact when you receive it.
2. Check the contents against the “packing list” in the package to ensure that no devices or accessories are missing or damaged.

The following figure shows the devices and accessories included in a vision system shipment. The table below is for reference only. Please take the “packing list” in the package as final.



No.	Category	Name	Function
1	IPC and accessories	IPC	Provides running environment for Mech-Mind software
2		IPC accessories	Provides IPC accessories, such as antennas for WIFI connection
3		IPC power cable and adapter	Supplies power to IPC
4	Camera and accessories	Mech-Eye Industrial 3D Camera	Captures images
5		User manual	Mech-Eye Industrial 3D Camera User Manuals
6		Camera accessory box	Mounts the camera
7	Project accessories	License dongle	Authorizes the Software Suite license
8		Calibration board	Calibrates the camera
9		Flange adapter	Connects the calibration board to the robot flange
10		Camera DC power cable (standard: 20 meters)	Connects the camera and the DIN rail power supply; Longer power cables are an option
11		Camera Ethernet cable (standard: 20 meters)	Connects the camera and the IPC; longer camera Ethernet cables are an option
12		DIN rail power supply (standard)	Supplies power to Mech-Eye Industrial 3D Camera; The camera power adaptor is an option
13	Packing list		Lists all the devices and accessories in the package

Contact Mech-Mind if any items are missing or damaged.

Prepare Other Materials

In this tutorial, besides the items in the package, you still need to prepare the materials shown in the following table by yourself.

Item	Function
Monitor	Provides display for the IPC
HDMI cable	Connects the monitor and the IPC
RJ45 Ethernet cable	Connects the IPC and the robot controller

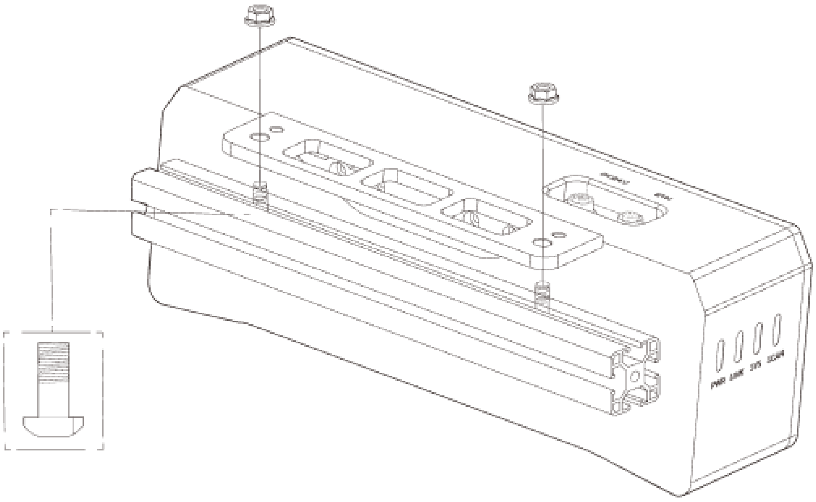
In this tutorial, the IPC and robot controller are directly connected through an RJ45-to-RJ45 Ethernet cable, and the IPC and the camera are directly connected through the camera Ethernet cable. Alternatively, you can use a router to connect the IPC and the robot controller, and the IPC and the camera, which is not covered in this topic.

Install Hardware

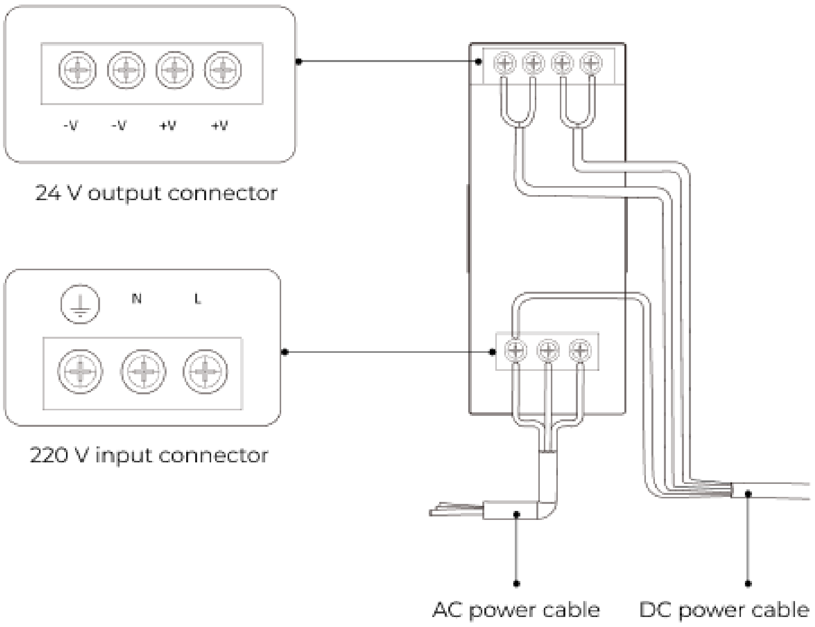
Mount the Camera

In this tutorial, the camera is mounted on the camera mounting frame (that is the eye to hand (ETH) mounting mode). In addition, the camera can also be mounted onto the end terminal of the robot (that is the eye in hand (EIH) mounting mode).

1. Get the camera mounting bolts and wrench from the camera accessory box.
2. Tighten the two bolts with the wrench as shown below.

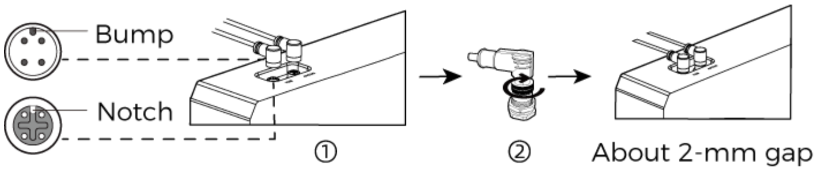


3. Please remove the lens protection film after mounting the camera.
4. Power the camera with the DIN rail power supply.
   - Connect the DC power cable:
     - Connect the +V wire to the +V connectors of the 24 V output connectors;
     - Connect the -V wire to the -V connectors of the 24 V output connectors;
     - Connect the PE wire to the 220 V input connector ⏚.



5. Install the Ethernet cable of the camera.

   Make sure the bump of the M12 connector and the notch of the ETH port, and tighten the nut after plugging in the cable.

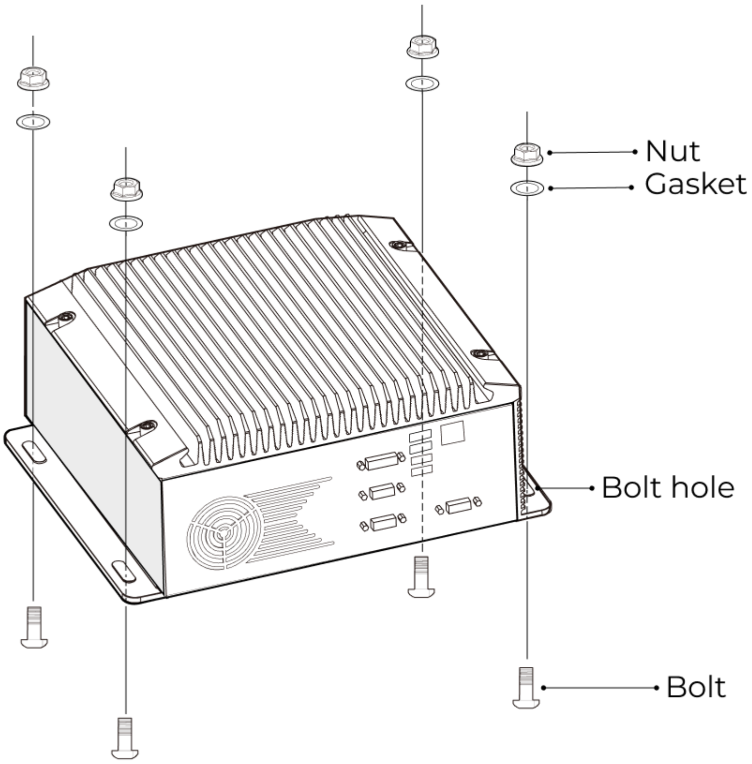


Mount the IPC

The IPC is normally mounted in the control cabinet of the robot. The environment in which the IPC is mounted requires good heat dissipation, ventilation, and dust protection. It should be mounted at a location where Ethernet cables, HDMI cable, and USB ports can be easily mounted and serviced.

To mount the IPC, follow these steps:

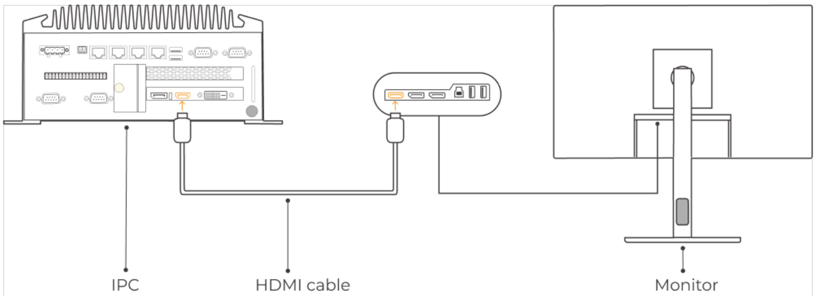
1. Prepare the wrench, bolts, nuts, and gaskets that are not included in the package beforehand.
2. If the robot controller is designed with mounting holes for the IPC in it, secure the IPC in the controller: place the bolt, gasket, and nut one by one, and tighten the two bolts with a wrench, as shown below.



If the location of the robot controller is already fixed, skip this step and just place the IPC inside the controller.

3. Connect the IPC and the monitor with the HDMI cable.

Plug one end of the HDMI cable into the HDMI port of the monitor, and the other end into the HDMI port of the IPC, as shown below.



4. Connect the IPC to the power supply unit with the power adaptor.

   Plug the power cable of the power adaptor into the power connector of the IPC. Connect the adaptor to the power supply on the other end.

5. Insert the license dongle.

   Plug the license dongle into a USB port of the IPC.

6. After the IPC is connected to the power supply, switch on the IPC.
   - If the IPC is started normally, the power indicator should be solid on.
   - If the IPC cannot be started, contact Mech-Mind Technical Support.

Connect the Network

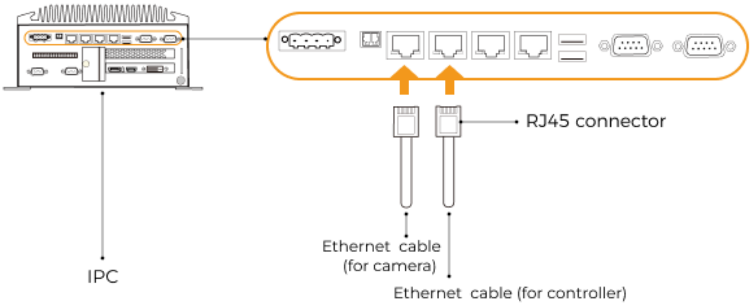
In this section, you will learn how to connect the network between the IPC and the camera, and between the IPC and the robot.

In the following sections, the following IP addresses will be used for network settings. Please adjust the network settings according to your actual network environment.

Device		IP Address
IPC	Ethernet port connecting to the camera	192.168.100.10
	Ethernet port connecting to the robot controller	192.168.200.10
Camera		192.168.100.20
Robot		192.168.200.20 (already set on the robot)

Connect the IPC and the Camera, and the IPC and the Robot Controller

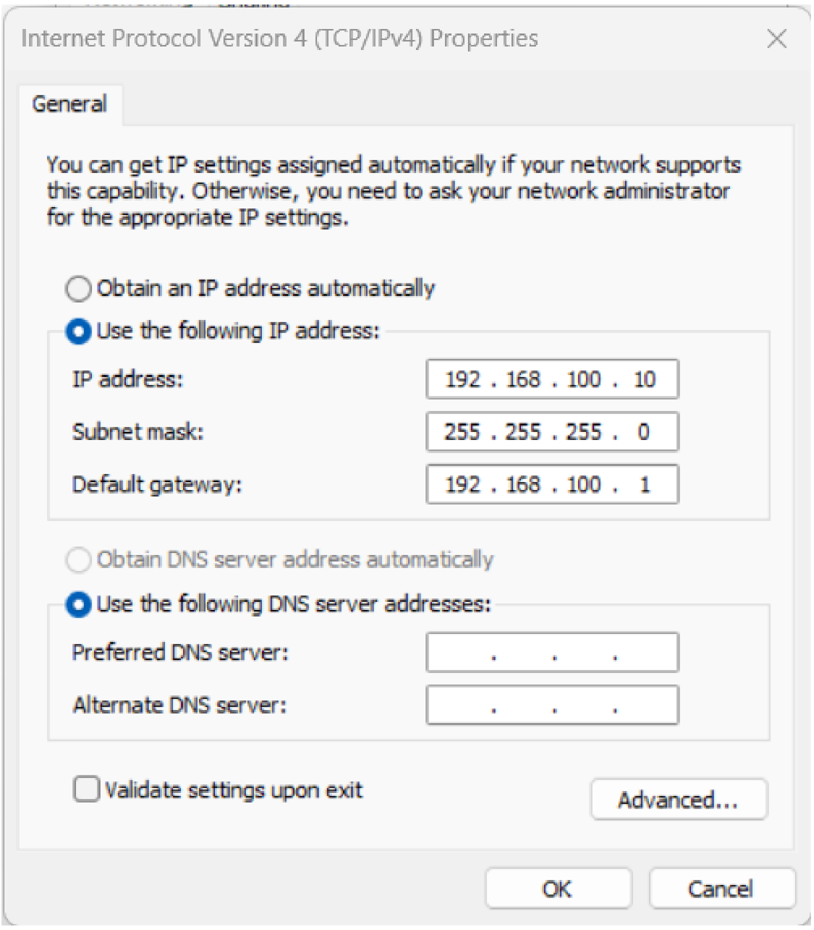
1. Plug the other end of the Ethernet cable connected to the camera into an Ethernet port of the IPC.



2. Use an RJ45-to-RJ45 Ethernet cable to plug one end into the Ethernet port of the robot controller and the other end into an Ethernet port of the IPC.

Set the IP Addresses on the IPC

1. Select Control Panel › Network and Internet › Network and Sharing Center › Change adapter settings on the IPC. The **Network Connections** page will be displayed.
2. Right-click the Ethernet port connected to the camera, and select **Rename** to rename the Ethernet port, such as "To_camera".
3. Right-click the Ethernet port connected to the camera, and select **Properties** to enter the **Ethernet Properties** page.
4. Select the **Internet Protocol Version 4 (TCP/IPv4)** checkbox, and then click the **[Property]** button to enter the **Internet Protocol Version 4 (TCP/IPv4) Properties** page.
5. Select the **Use the following IP address** radio button, set the **IP address** parameter to "192.168.100.10", **Subnet mask** to "255.255.255.0", and **Default gateway** to "192.168.100.1", and then click the **[OK]** button.



Internet Protocol Version 4 (TCP/IPv4) Properties

You can get IP settings assigned automatically if your network supports this capability. Otherwise, you need to ask your network administrator for the appropriate IP settings.

○ Obtain an IP address automatically

● Use the following IP address:

IP address: 192 . 168 . 100 . 10

Subnet mask: 255 . 255 . 255 . 0

Default gateway: 192 . 168 . 100 . 1

○ Obtain DNS server address automatically

● Use the following DNS server addresses:

Preferred DNS server: . . .

Alternate DNS server: . . .

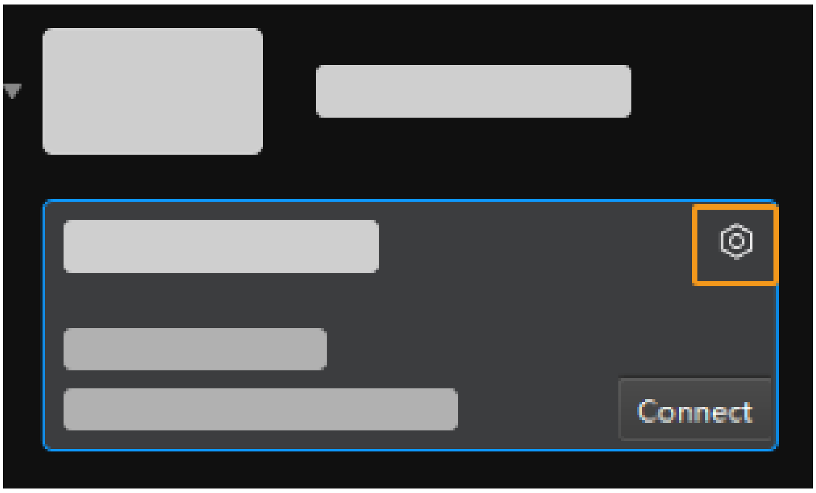
□ Validate settings upon exit

6. Repeat steps 2 to 5 to rename the Ethernet port connected to the robot controller (for example, “To_robot”), and set the IP address for this Ethernet port. For example, set the IP address of this Ethernet port to “192.168.200.10”.

> The IP address of the robot and that of the IPC Ethernet port connected to the robot controller must be on the same subnet.

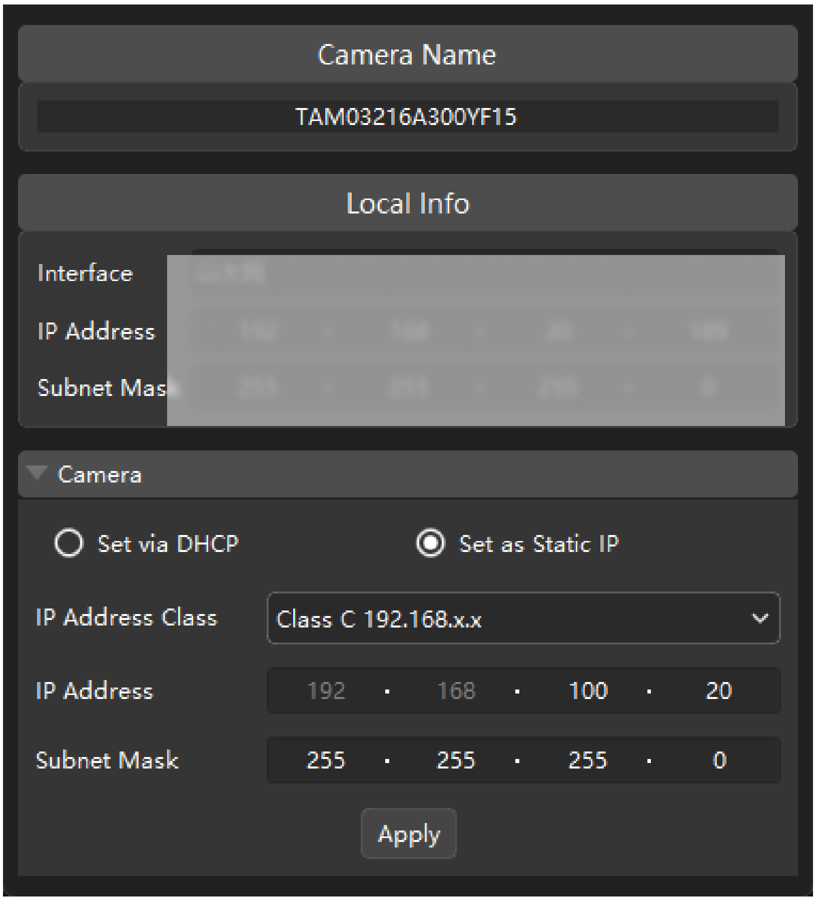
Set the Camera IP Address

1. Double-click EYE on the desktop of the IPC to open and run Mech-Eye Viewer.
2. Select the camera in the camera list, and hover the cursor on the camera. Click ⚙ to open the **Config Camera Network** dialog box.



If the camera cannot be found or connected, please refer to Camera Troubleshooting.

3. In the **Camera** area, select the **Set as Static IP** radio button, set **IP Address Class** to “Class C 192.168.x.x”, **IP Address** to **192.168.100.20**, and **Subnet Mask** to “255.255.255.0” and then click **[Apply]**.



Camera Name

TAM03216A300YF15

Local Info

Interface [redacted]

IP Address

Subnet Mas

Camera

○ Set via DHCP ● Set as Static IP

IP Address Class: Class C 192.168.x.x

IP Address: 192 . 168 . 100 . 20

Subnet Mask: 255 . 255 . 255 . 0

The IP address of the camera and that of the IPC Ethernet port connected to the camera must be on the same subnet.

Test the Network Connectivity

1. Press `Win` + `R` to open the **Run** dialog box.

2. Type cmd in the **Run** dialog box, and then click **[OK]**.
3. Type **ping XXX.XXX.XX.XX** in the command prompt window and press **[Enter]** to execute the command.

💡 Replace XXX.XXX.XX.XX with the actual IP address of the camera/robot.

If the network connectivity is normal, you should receive the following response:

```
Pinging XXX.XXX.XX.XX with 32 bytes of data:
Reply from XXX.XXX.XX.XX: bytes=32 time<1ms TTL=128
Reply from XXX.XXX.XX.XX: bytes=32 time<1ms TTL=128
Reply from XXX.XXX.XX.XX: bytes=32 time<1ms TTL=128
Reply from XXX.XXX.XX.XX: bytes=32 time<1ms TTL=128
```

Upgrade the Software (Optional)

The IPC purchased from Mech-Mind already has the latest version of Mech-Mind software installed.

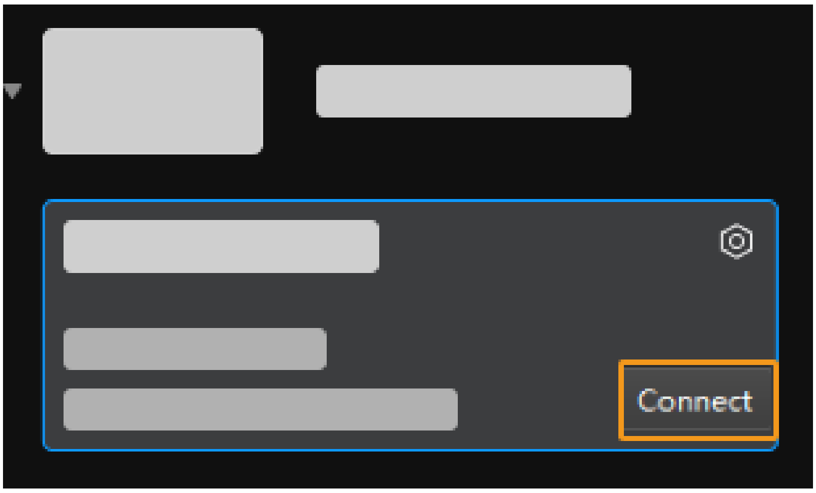
Please check if all software on the IPC is running with the latest version. If so, skip this section; if not, follow the sections below to update the software to the latest versions.

- Downloading and installing Mech-Eye SDK
- Downloading and installing Mech-Vision

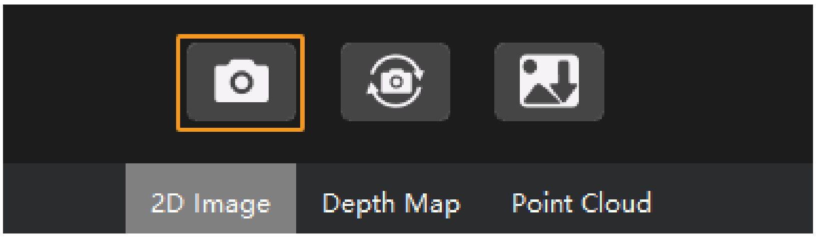
Confirm That the Vision System Can Capture Images Normally

After testing the network connectivity between the IPC and the camera/robot, please confirm that the vision system can capture images normally:

1. Place workpieces at the center of the camera’s field of view (FOV), and make sure that the workpieces on the edge and top are all within the FOV.
2. Open and start Mech-Eye Viewer by double-clicking EYE on the desktop of the IPC.
3. Select the camera in the camera list and click **[Connect]**.

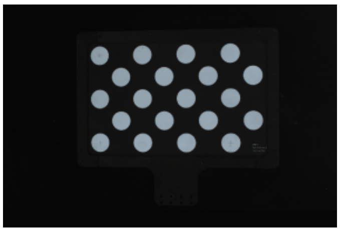
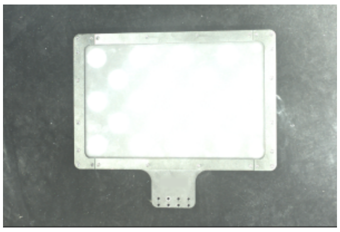
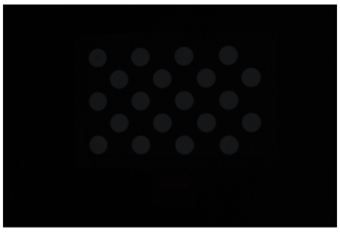
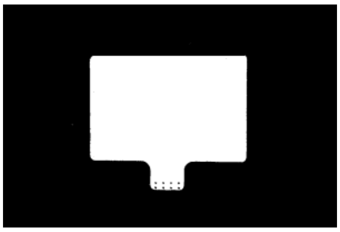
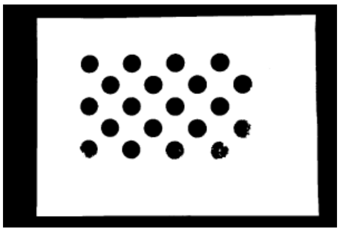
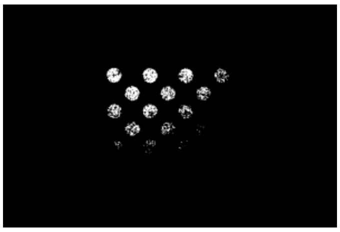


4. After the camera is connected, click **[Capture once]**.



5. Make sure that the captured images meet the following standard.

   - 2D image: the captured 2D image is not significantly overexposed (too white to recognize objects) or underexposed (too dark to recognize object details).
   - Depth map and point cloud: All relevant parts of the object can be recognized.

	Normal	Overexposed	Underexposed
2D image			
Point cloud			

If the captured images do not meet the required standard, please use Mech-Eye Viewer to adjust camera parameters.

Up to now, you have learned how to build the vision system.

3.2. Robot Communication Setup

In this tutorial, you will learn how to load the Standard Interface program files to the YASKAWA robot, and set up the Standard Interface communication between the Mech-Mind Vision System and the robot.

- In this section, you will load the robot Standard Interface program and the configuration files to the robot system to establish the Standard Interface communication between the vision system and the robot.
- If you are using the robots of other brands, you can find instructions on setting up the Standard Interface communication with the desired robot from the section Standard Interface Communication.

▶ https://www.youtube.com/watch?v=2ERGMtWkZV8/PLVcMd7cW2rXVtrAejMyVQni2dUDv8bxje *(YouTube video)*

Video Tutorial: Robot Interface Configuration (Standard Interface Communication)

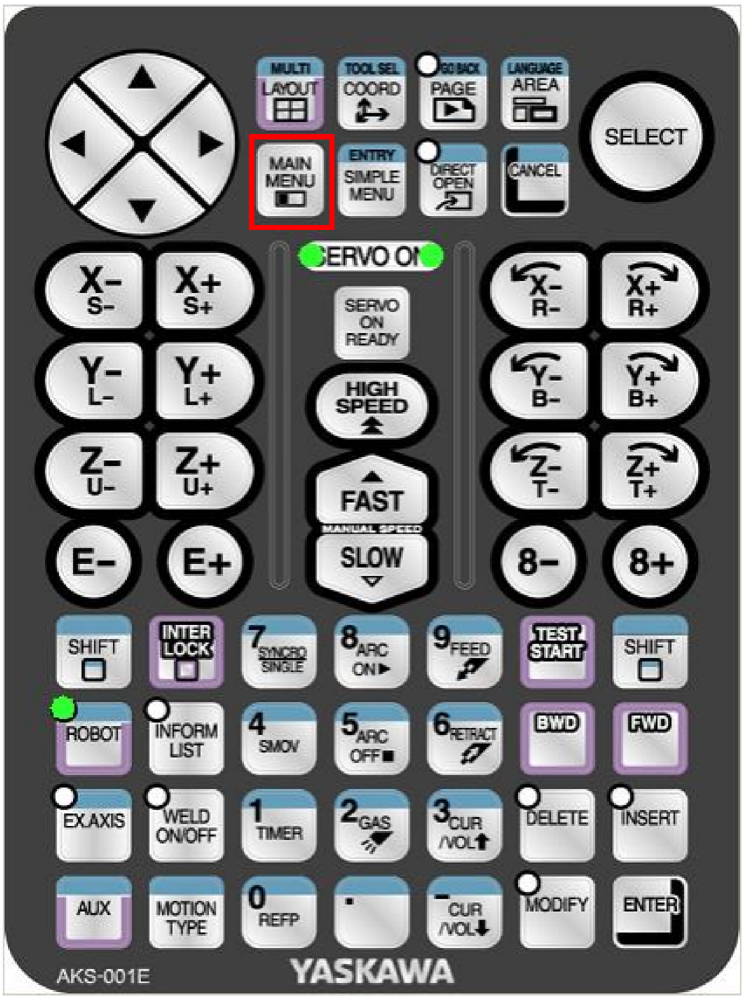
Preparation Before Loading

Check Hardware and Software Versions

- Confirm that the robot is a 6-axis YASKAWA robot. This tutorial uses YASKAWA_GP8 as an example.
- Confirm that the robot controller is YRC1000, YRC1000micro or DX200. This tutorial uses the controller YRC1000 as an example.
- Confirm that the Ethernet option has been enabled for the YASKAWA robot.

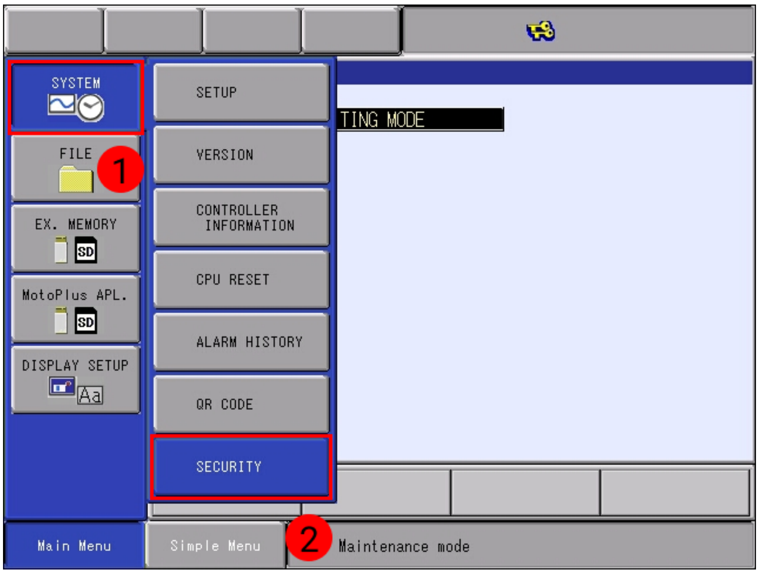
▼ *Operation instructions*

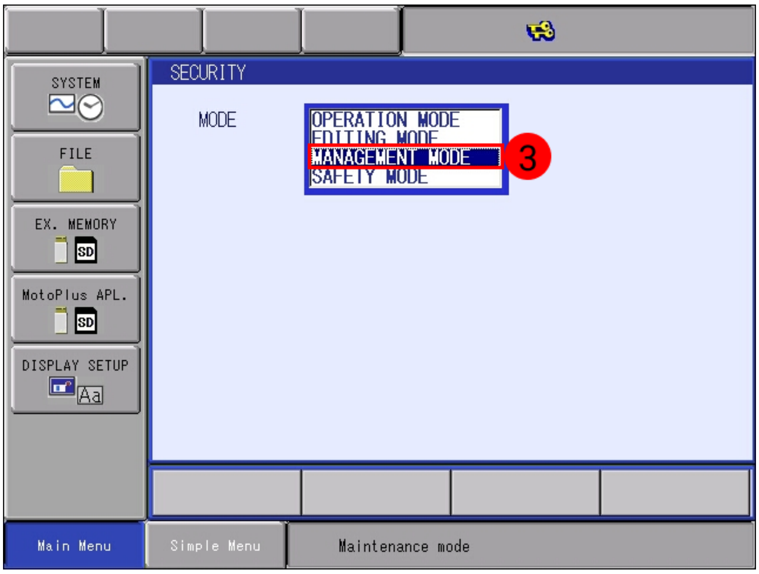
a. Press and hold the `MAIN MENU` key on the teach pendant, and power on the robot to enter the **maintenance** mode.



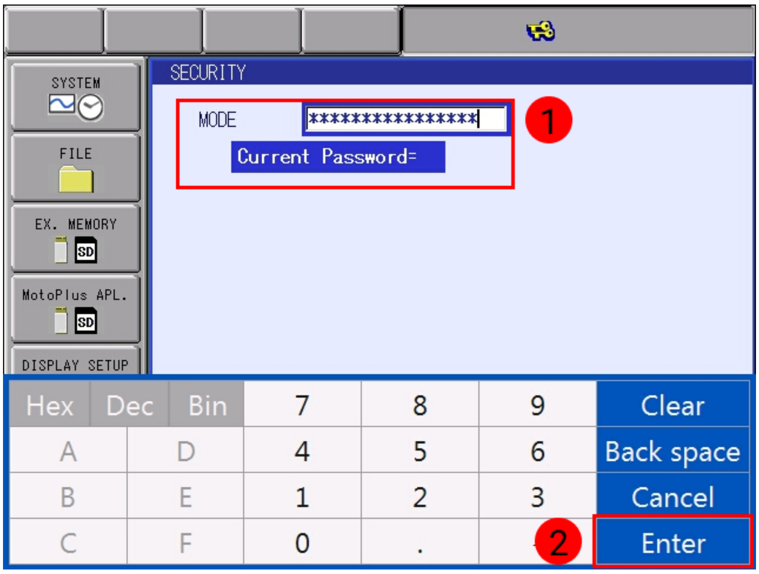
> If the robot is already started, please restart the robot while pressing the `MAIN MENU` key on the teach pendant.

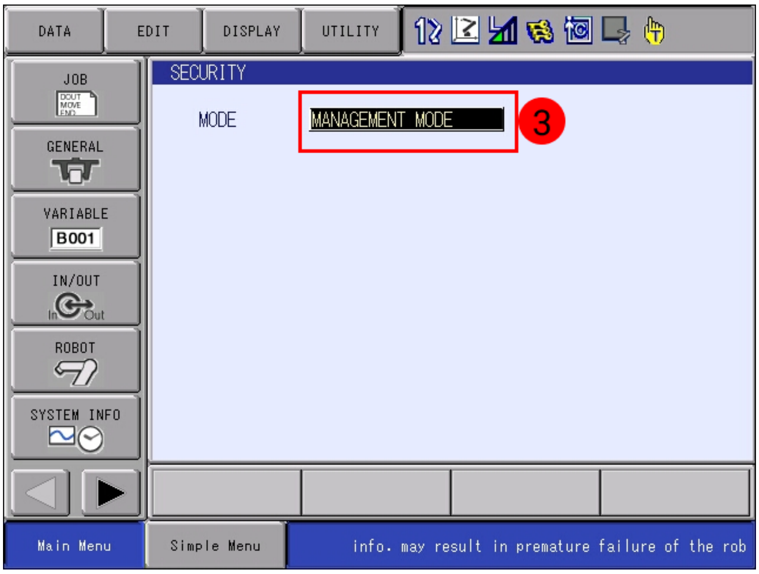
b. In the **maintenance** mode, select SYSTEM › SECURITY › MANAGEMENT MODE.





c. Enter the password (the default password is sixteen 9s), and then click [Enter] to enter the MANAGEMENT MODE.



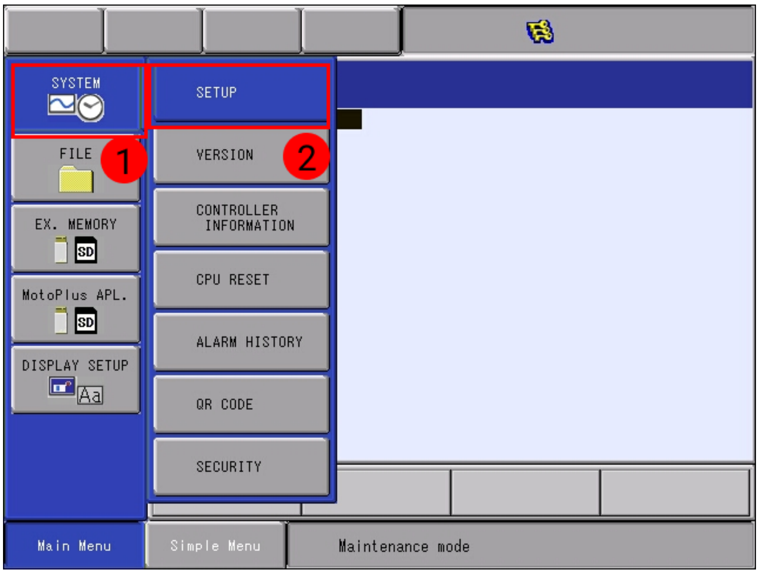


SECURITY

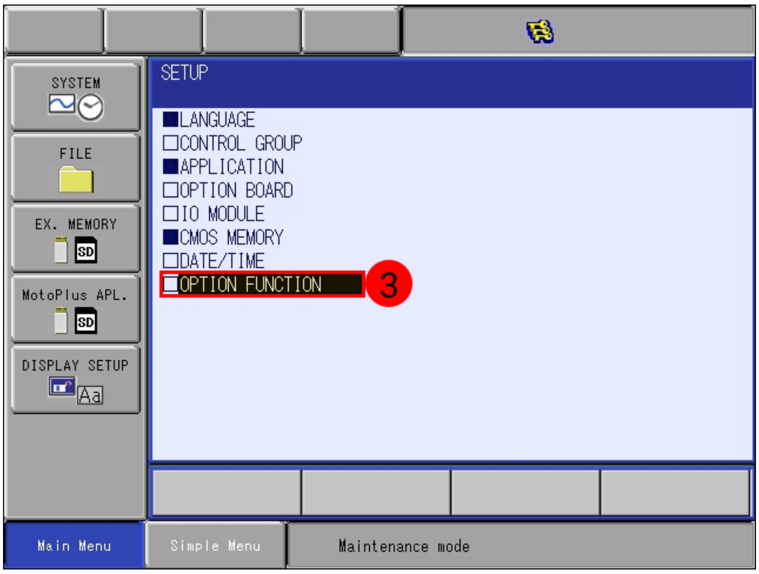
MODE MANAGEMENT MODE

info. may result in premature failure of the rob

d. On the **Main Menu**, select SYSTEM › SETUP › OPTION FUNCTION. Select **DETAIL** of the **NETWORK FUNCTION SETTING** option, and press the SELECT key on the teach pendant to enter the **NETWORK FUNCTION SETTING** interface.

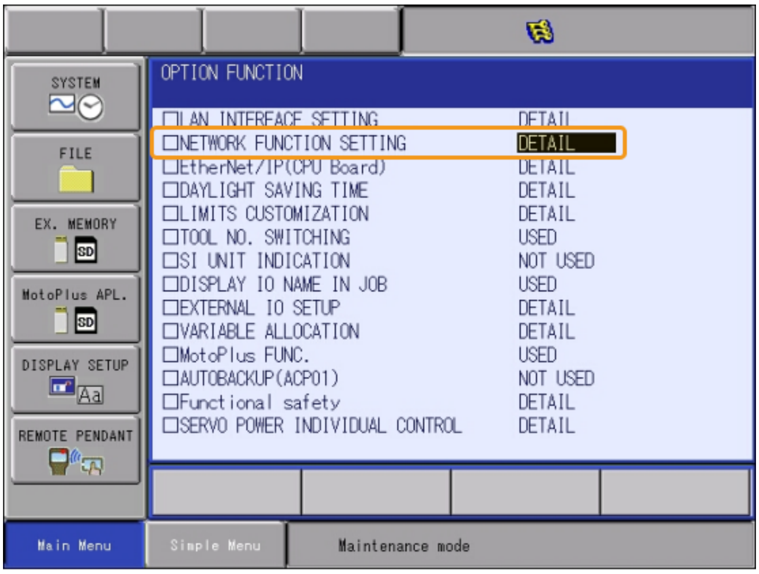


Maintenance mode



SETUP

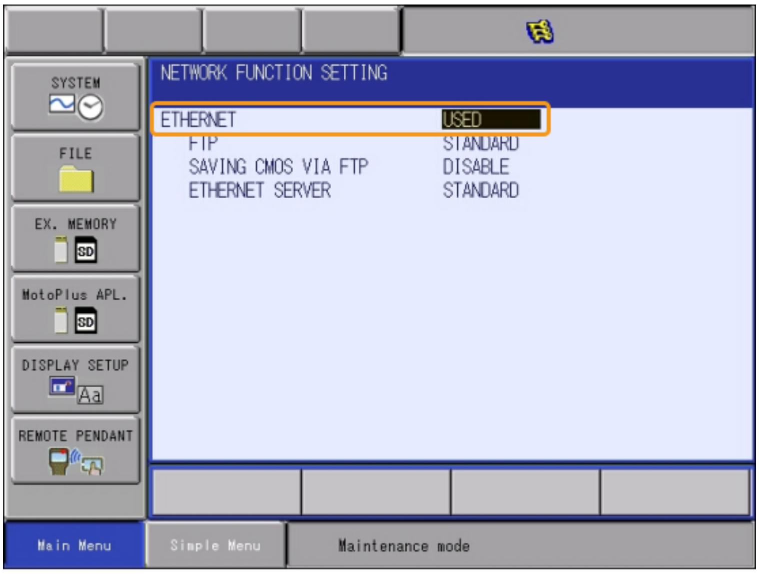
■LANGUAGE
□CONTROL GROUP
■APPLICATION
□OPTION BOARD
□IO MODULE
■CMOS MEMORY
□DATE/TIME
□OPTION FUNCTION



OPTION FUNCTION

```
□LAN INTERFACE SETTING              DETAIL
□NETWORK FUNCTION SETTING           DETAIL
□EtherNet/IP(CPU Board)             DETAIL
□DAYLIGHT SAVING TIME               DETAIL
□LIMITS CUSTOMIZATION               DETAIL
□TOOL NO. SWITCHING                 USED
□SI UNIT INDICATION                 NOT USED
□DISPLAY IO NAME IN JOB             USED
□EXTERNAL IO SETUP                  DETAIL
□VARIABLE ALLOCATION                DETAIL
□MotoPlus FUNC.                     USED
□AUTOBACKUP(ACP01)                  NOT USED
□Functional safety                  DETAIL
□SERVO POWER INDIVIDUAL CONTROL     DETAIL
```

e. Check whether the status of **ETHERNET** is displayed as **USED**.



NETWORK FUNCTION SETTING

```
ETHERNET                    USED
    FTP                     STANDARD
    SAVING CMOS VIA FTP     DISABLE
    ETHERNET SERVER         STANDARD
```

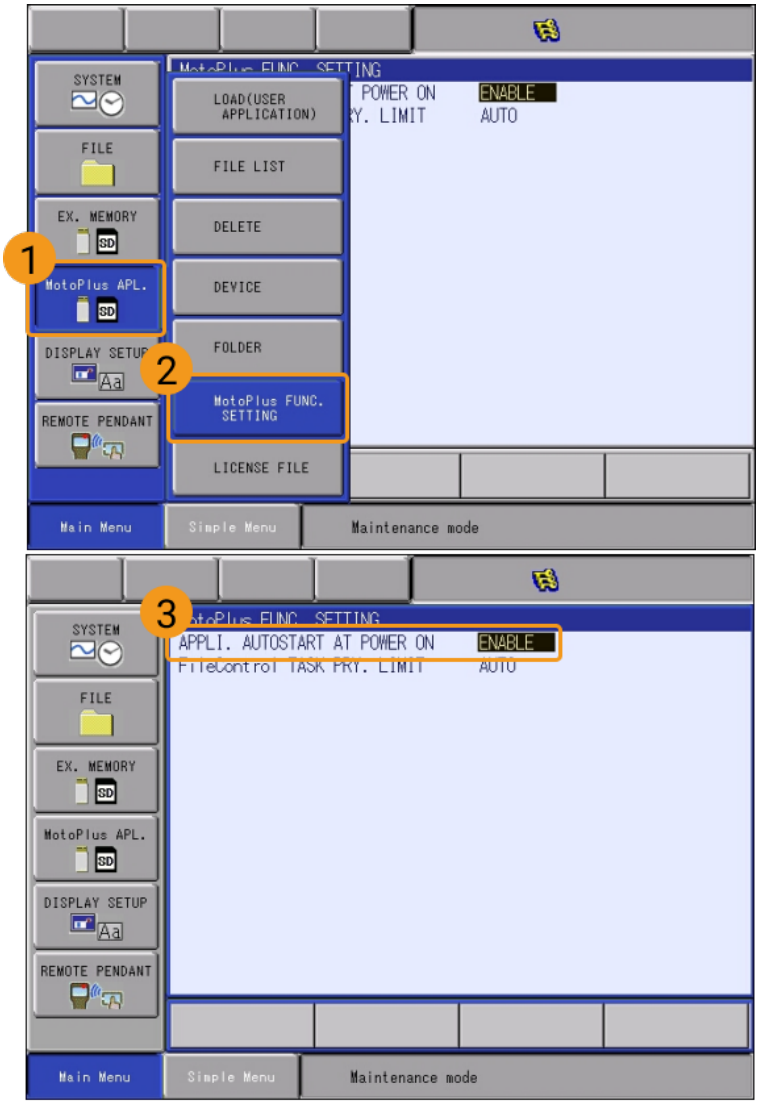
If the status is “UNUSED”, please contact the robot manufacturer for support.

- Confirm that the MotoPlus option has been enabled for the YASKAWA robot.

▼ *Operation instructions*

a. In the **MANAGEMENT MODE** of the **maintenance** mode, select MotoPlus APL. › MotoPlus FUNC. SETTING on the **Main Menu**.

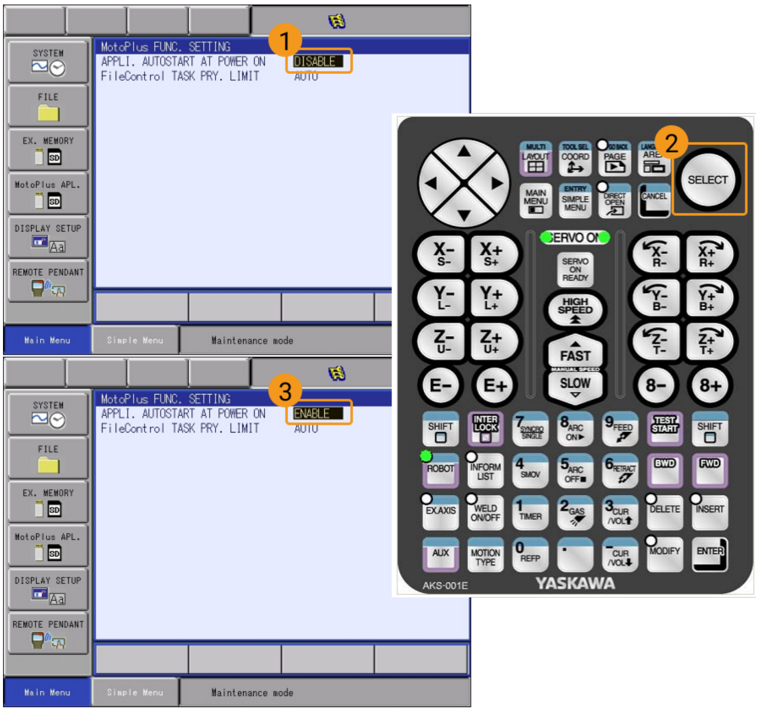
b. On the **MotoPlus FUNC. SETTING** interface, check whether the status of **APPLI. AUTOSTART AT POWER ON** is **ENABLE**.



MotoPlus FUNC. SETTING

APPLI. AUTOSTART AT POWER ON ENABLE
FileControl TASK PRY. LIMIT AUTO

c. If the status is “UNABLE”, select **UNABLE**, and press the SELECT key on the teach pendant to switch the status to “ENABLE”.



MotoPlus FUNC. SETTING
APPLI. AUTOSTART AT POWER ON DISABLE
FileControl TASK PRY. LIMIT AUTO

MotoPlus FUNC. SETTING
APPLI. AUTOSTART AT POWER ON ENABLE
FileControl TASK PRY. LIMIT AUTO

If the preceding conditions cannot be met, the vision system cannot communicate with the robot through the Standard Interface. Please contact the robot manufacturer for support.

Connect the Network

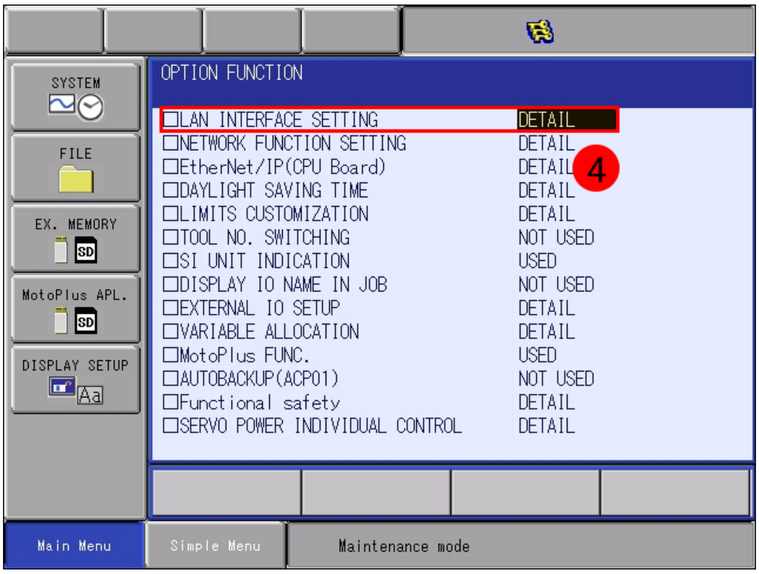
1. Plug the Ethernet cable of the IPC into the LAN2 (CN106) port on the CPU board of the robot controller.

   If you are using the DX200 controller, plug the Ethernet cable of the IPC into the CN104 port on the CPU board of the robot controller.

2. Make sure that the IP address of the YASKAWA robot and that of the IPC are in the same subnet.

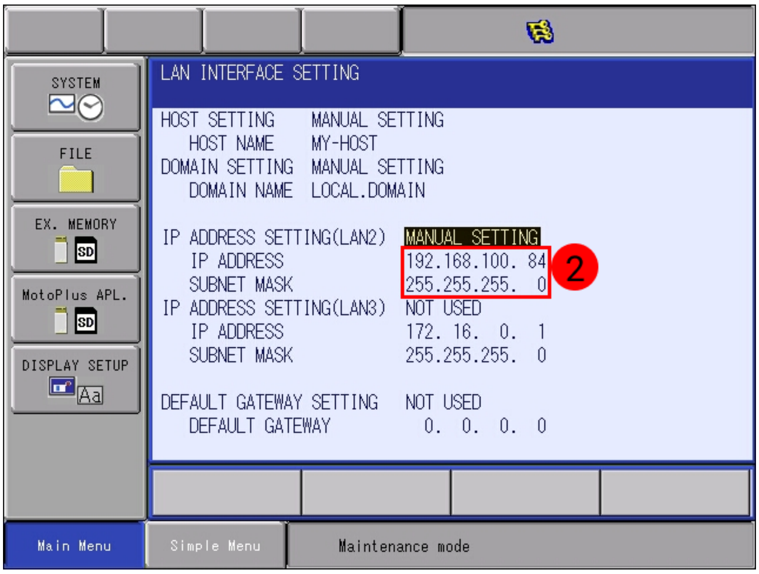
   ▼ *Operation instructions*

   a. In the **MANAGEMENT MODE** of the **maintenance** mode, select SYSTEM › SETUP › OPTION FUNCTION on the **Main Menu**. Select **DETAIL** of the **LAN INTERFACE SETTING** option, and press the SELECT key on the teach pendant to enter the **LAN INTERFACE SETTING** interface.



```
OPTION FUNCTION

□LAN INTERFACE SETTING               DETAIL
□NETWORK FUNCTION SETTING            DETAIL
□EtherNet/IP(CPU Board)              DETAIL
□DAYLIGHT SAVING TIME                DETAIL
□LIMITS CUSTOMIZATION                DETAIL
□TOOL NO. SWITCHING                  NOT USED
□SI UNIT INDICATION                  USED
□DISPLAY IO NAME IN JOB              NOT USED
□EXTERNAL IO SETUP                   DETAIL
□VARIABLE ALLOCATION                 DETAIL
□MotoPlus FUNC.                      USED
□AUTOBACKUP(ACP01)                   NOT USED
□Functional safety                   DETAIL
□SERVO POWER INDIVIDUAL CONTROL      DETAIL
```

b. Check the IP address of the robot (that is IP address of LAN2).

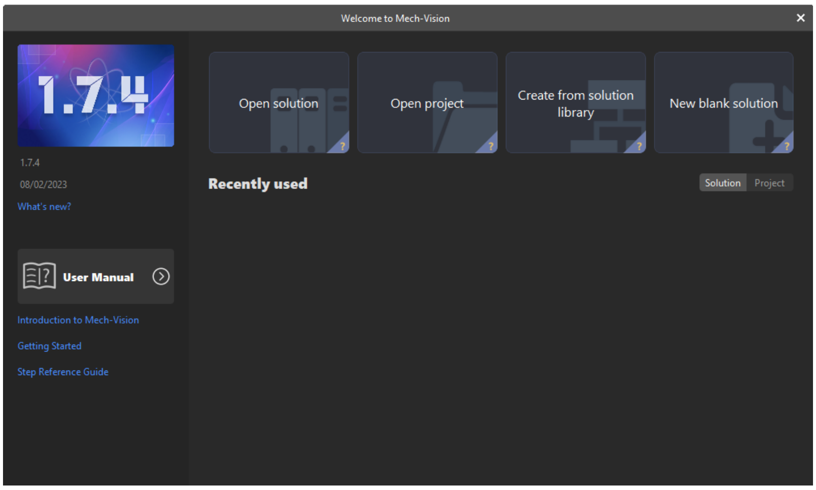


```
LAN INTERFACE SETTING

HOST SETTING     MANUAL SETTING
   HOST NAME     MY-HOST
DOMAIN SETTING   MANUAL SETTING
   DOMAIN NAME   LOCAL.DOMAIN

IP ADDRESS SETTING(LAN2)  MANUAL SETTING
   IP ADDRESS             192.168.100. 84
   SUBNET MASK            255.255.255.  0
IP ADDRESS SETTING(LAN3)  NOT USED
   IP ADDRESS             172. 16.  0.  1
   SUBNET MASK            255.255.255.  0

DEFAULT GATEWAY SETTING   NOT USED
   DEFAULT GATEWAY          0.  0.  0.  0
```

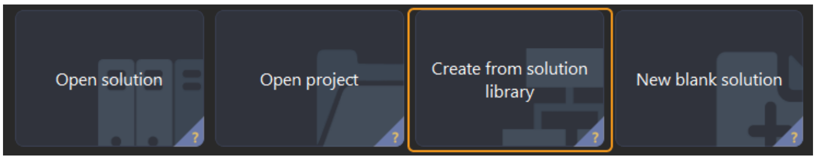
c. Make sure that the IP address of the robot and that of the IPC are in the same subnet. If they are not in the same subnet, please refer to the section Set the IP Addresses on the IPC to modify the IP address of the IPC.

Create a Mech-Vision Project and Save It

1. Open Mech-Vision. If the Welcome interface as shown below is displayed, it indicates that Mech-Vision is started successfully.

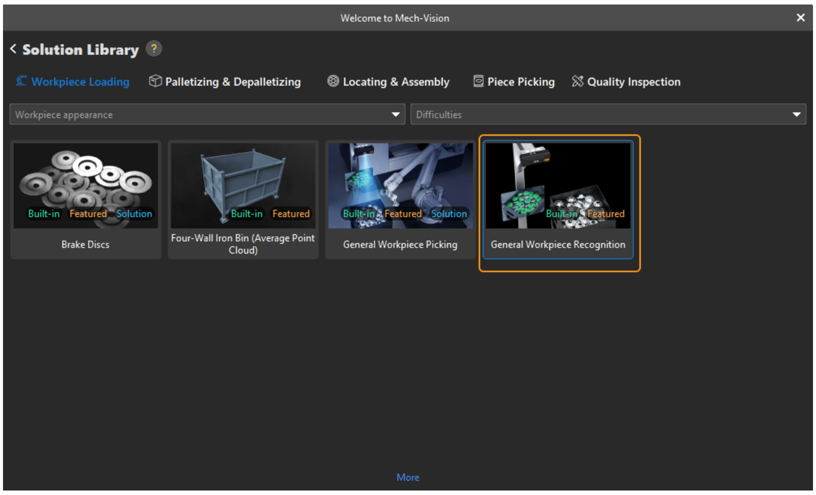


2. In the Welcome interface of Mech-Vision, click [**Create from solution library**] to open the Solution Library.



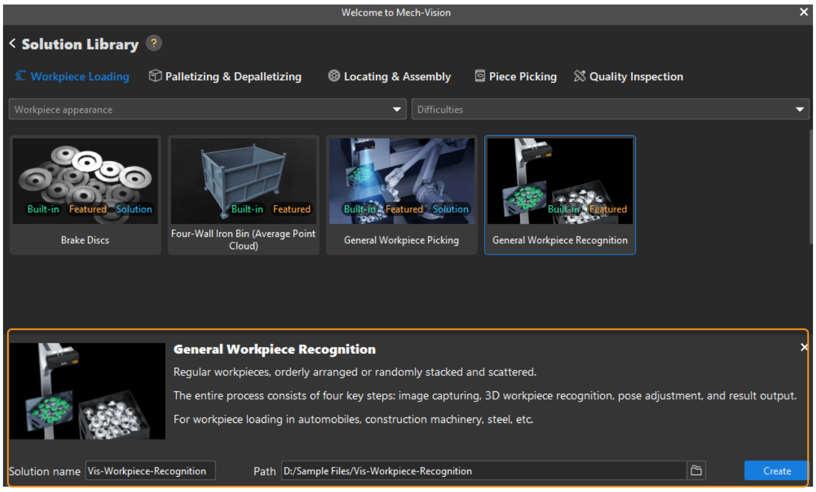
The Solution Library is a resource library that provides typical solutions or projects (with sample data) from various application scenarios.

3. Select the **General Workpiece Recognition** project from the Solution Library.



If you cannot find this project in the Solution Library, click [**More**] at the bottom of the Solution Library interface.

4. After this project is selected, the information introducing this project will be displayed on the lower part of the Solution Library interface. Set the solution name and path, and then click [**Create**].



General Workpiece Recognition

Regular workpieces, orderly arranged or randomly stacked and scattered.

The entire process consists of four key steps: image capturing, 3D workpiece recognition, pose adjustment, and result output.

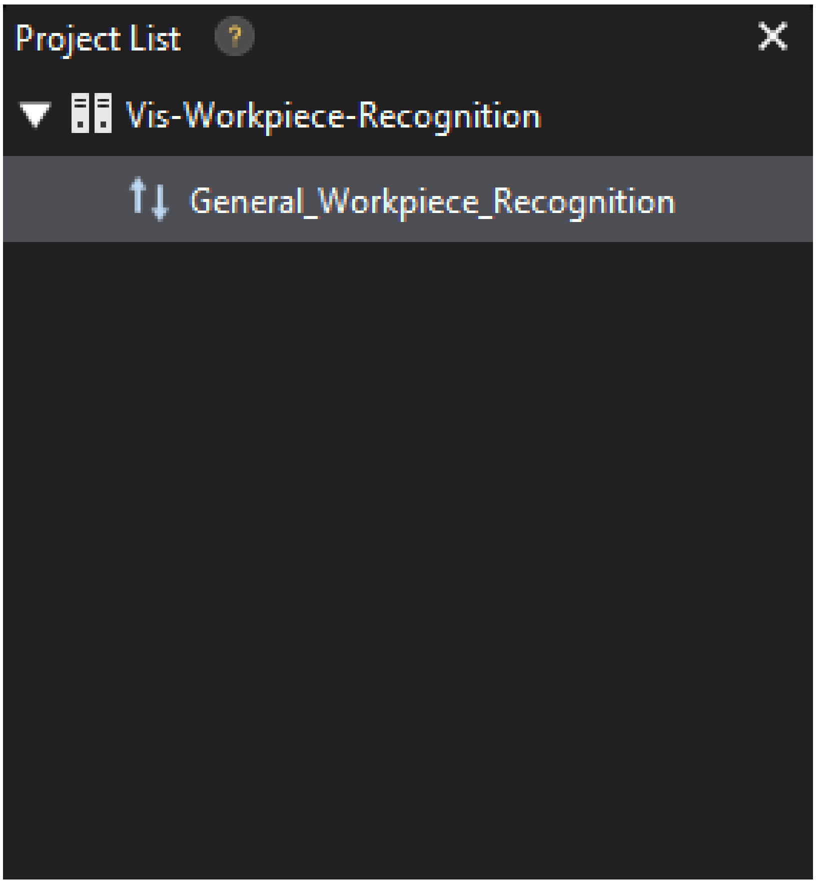
For workpiece loading in automobiles, construction machinery, steel, etc.

Solution name Vis-Workpiece-Recognition

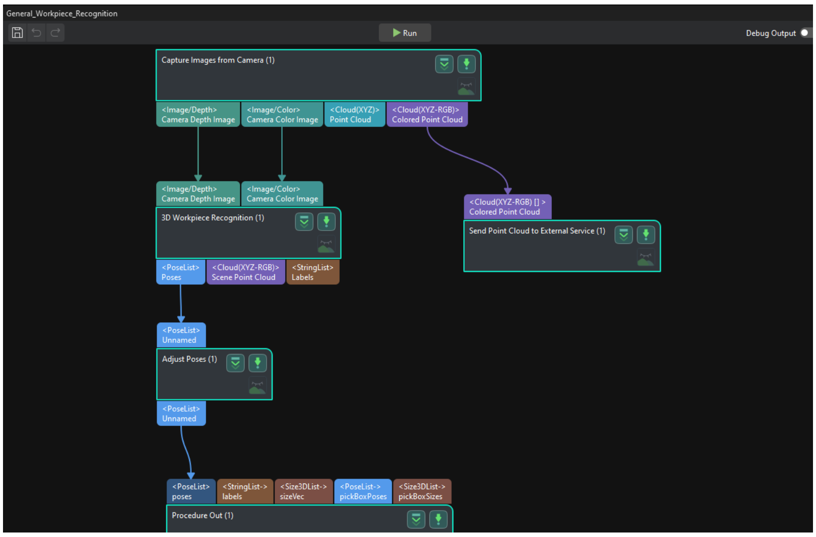
Path D:/Sample Files/Vis-Workpiece-Recognition

After the project is created, the created solution and project will be displayed in the project list in the upper-left corner of the Mech-Vision main interface.

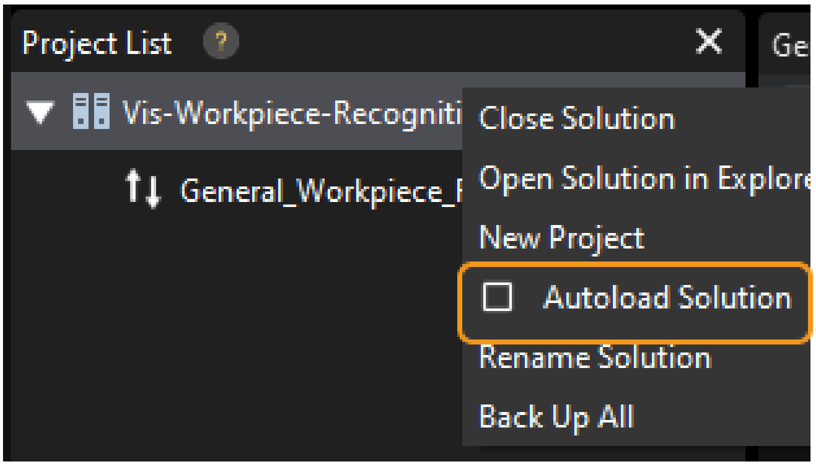
- A **solution** is a set of configurations and data related to robots and robot communication, vision processing, path planning, etc. that are required for the machine vision application.
- A **project** is a workflow of vision processing in the solution. Normally, a solution only contains one project, but it may contain more than one project in complex application scenarios. In this tutorial, the solution requires only one project.



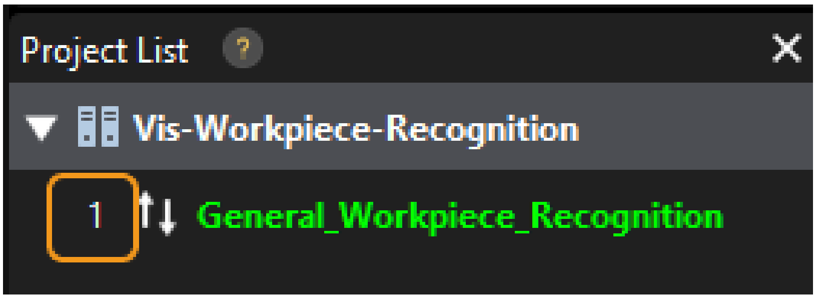
In the Graphical Programming Workspace of the main interface, the workflow of the “General Workpiece Recognition” project will be displayed.


General_Workpiece_Recognition
Run
Debug Output
Capture Images from Camera (1)
<Image/Depth> Camera Depth Image
<Image/Color> Camera Color Image
<Cloud(XYZ)> Point Cloud
<Cloud(XYZ-RGB)> Colored Point Cloud
<Image/Depth> Camera Depth Image
<Image/Color> Camera Color Image
3D Workpiece Recognition (1)
<PoseList> Poses
<Cloud(XYZ-RGB)> Scene Point Cloud
<StringList> Labels
<Cloud(XYZ-RGB) [] > Colored Point Cloud
Send Point Cloud to External Service (1)
<PoseList> Unnamed
Adjust Poses (1)
<PoseList> Unnamed
<PoseList> poses
<StringList-> labels
<Size3DList-> sizeVec
<PoseList-> pickBoxPoses
<Size3DList-> pickBoxSizes
Procedure Out (1)

5. In the project list, right-click the solution, and select **[Autoload Solution]**.

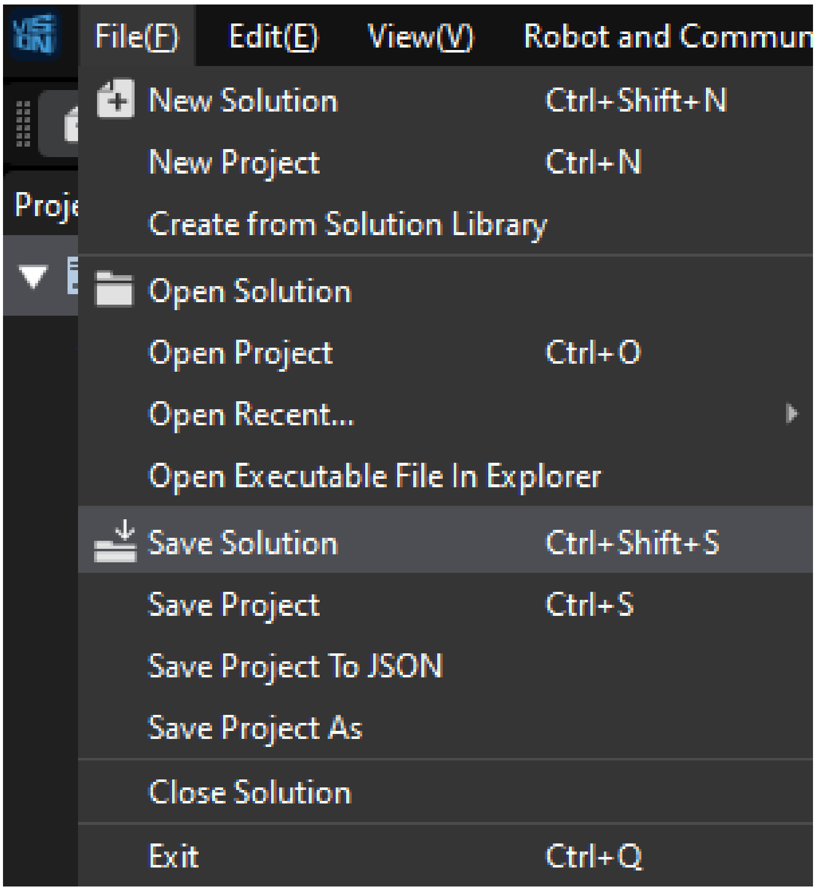


After this solution is set to autoload, the project name will be displayed in green, and the project ID will be displayed in the left of the project name.



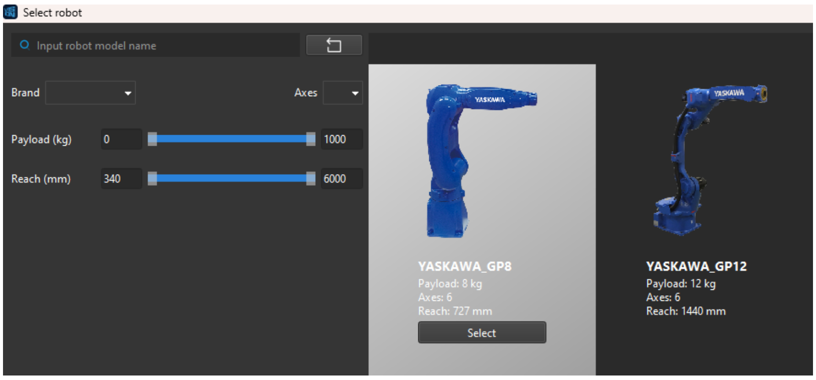
The project ID will be used by the robot pick-and-place program to trigger the Mech-Vision project to run.

6. On the menu bar, select File › Save Solution.

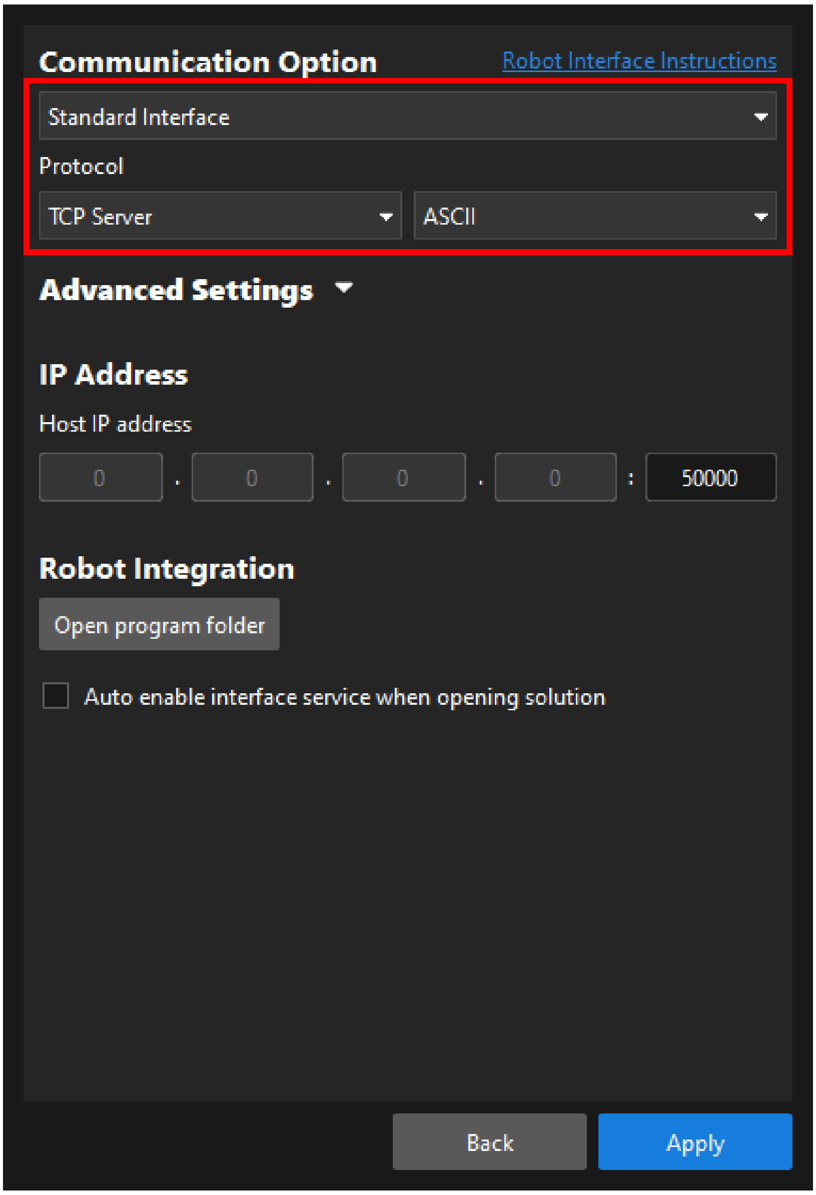


Set “Robot and Interface Configuration”

1. Click **[Robot and Interface Configuration]** in the toolbar of Mech-Vision.
2. Click the **Select robot** drop-down menu, and select **Listed robot**. Then, click **[Select robot model]**, select the YASKAWA robot model “YASKAWA-GP8” and click **[Next]** to continue.



3. Select **Standard Interface** in the **Communication Option** area, and **TCP Server** and **ASCII** for Protocol, and then click **[Apply]**.



Communication Option

Robot Interface Instructions

Standard Interface

Protocol

TCP Server

ASCII

Advanced Settings

IP Address

Host IP address

0 . 0 . 0 . 0 : 50000

Robot Integration

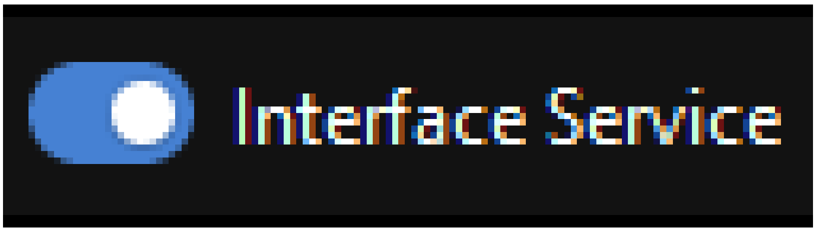
Open program folder

□ Auto enable interface service when opening solution

Back

Apply

4. Make sure the Interface Service is started: On the toolbar of Mech-Vision, the Interface Service switch on the far right is flipped and turned to blue.



Interface Service

Prepare the Program Files

1. On the IPC, open `Mech-Center/Robot_Interface/YASKAWA` in the installation directory of the Mech-Vision and Mech-Viz software.

2. Copy the **JBI** folder, **sample** folder, and the **mm_module_yrc1000.out** file to the root directory of the USB flash drive.

   - **mm_module_yrc1000.out**: the robot backend program file (MotoPlus application file).

  - **JBI** folder: stores robot frontend program files (Job files).
  - **sample**: stores simple pick-and-place example program files.

> If you are using the controller DX200, you need to copy the backend program file **mm_module_dx200.out** here.

3. Plug the USB flash drive into the USB port on the rear panel of the teach pendant.

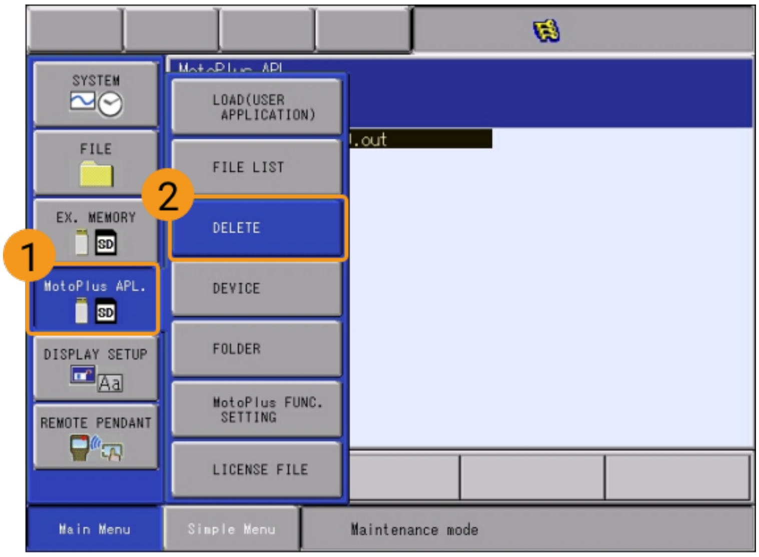
Make Sure That No Program Is Running in MotoPlus

Before you load robot program files to the robot, make sure that no other MotoPlus programs are running.

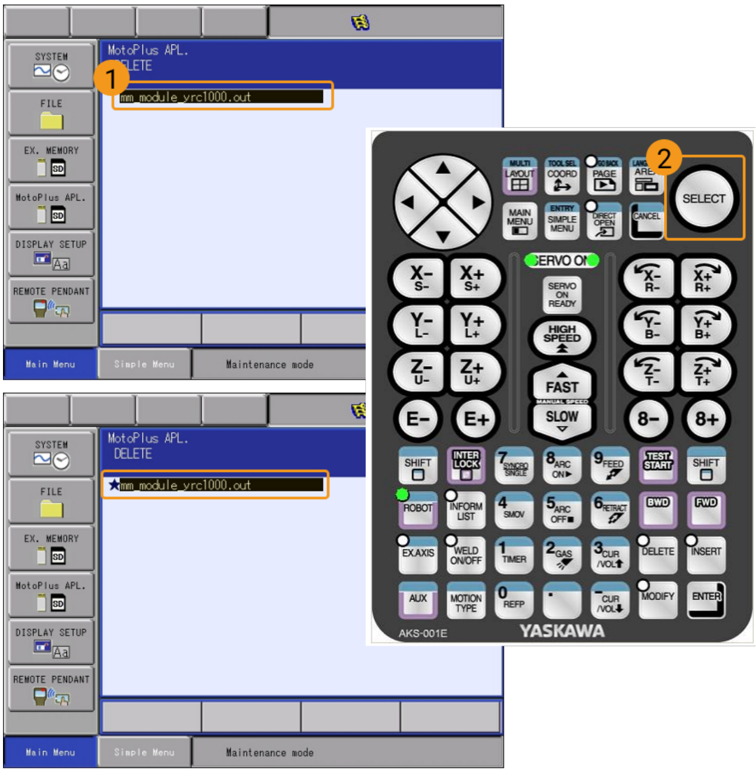
- If no programs are running, skip this section.
- If there are any running program, delete this application program.

▼ *Operation instructions*

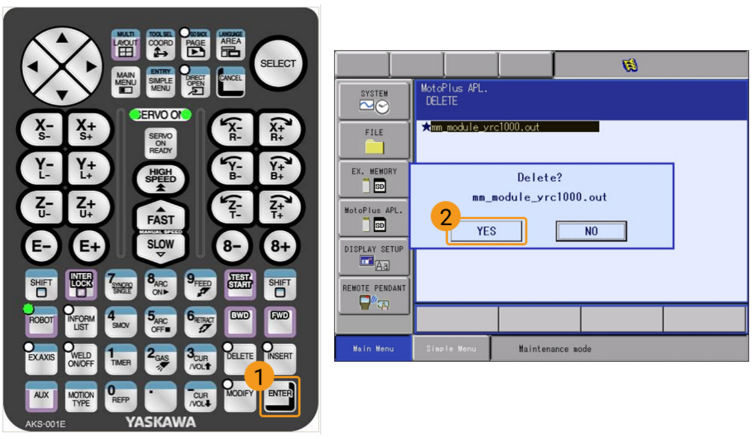
1. In the **MANAGEMENT MODE** of the **maintenance** mode, select MotoPlus APL. › DELETE on the **Main Menu**.



2. On the **MotoPlus APL. DELETE** interface, select the program file to delete, and then press the SELECT key on the teach pendant to select this file.



3. Press the ENTER key on the teach pendant, and press the [YES] button on the pop-up dialog box to delete the program.



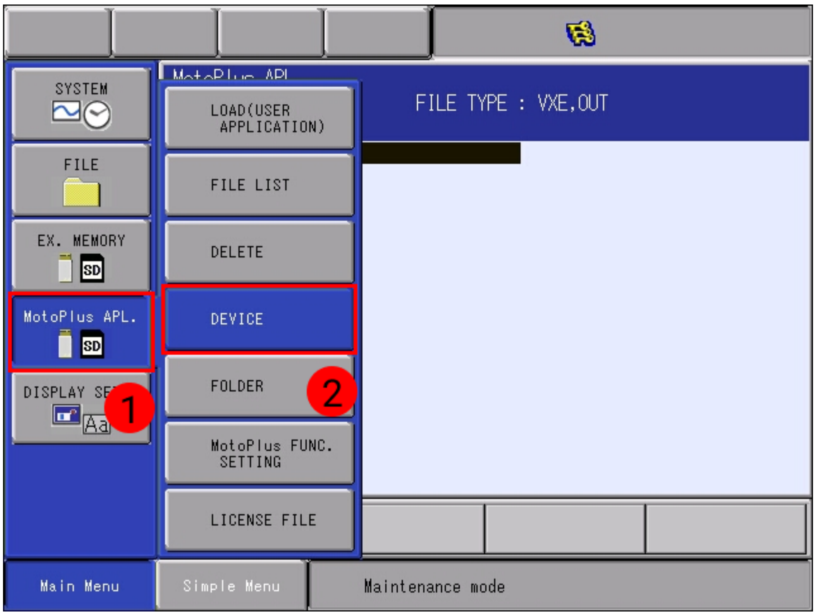
MotoPlus APL.
DELETE

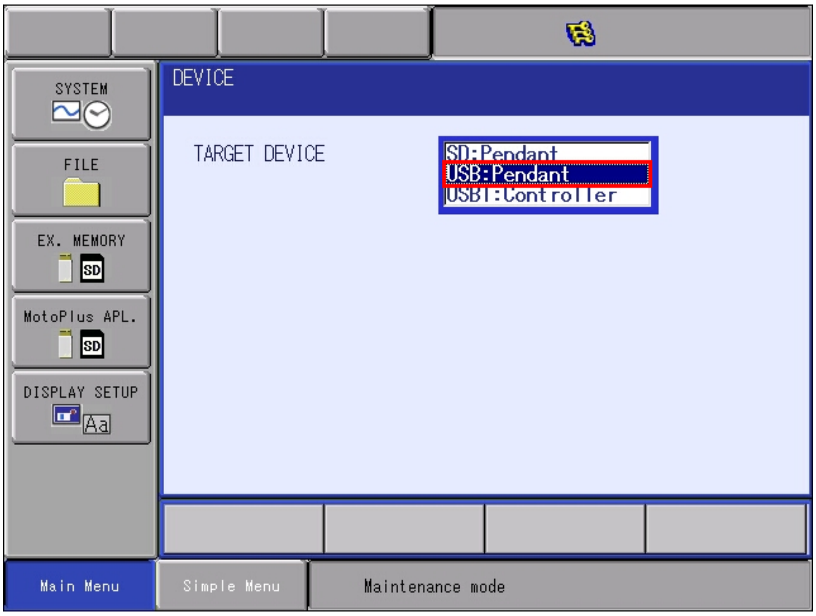
Delete?
mm_module_yrc1000.out

Load the Program Files to the Robot

Load the Backend Program File

1. Under the **maintenance** mode, select MotoPlus APL. › DEVICE › USB: Pendant on the **Main Menu**.

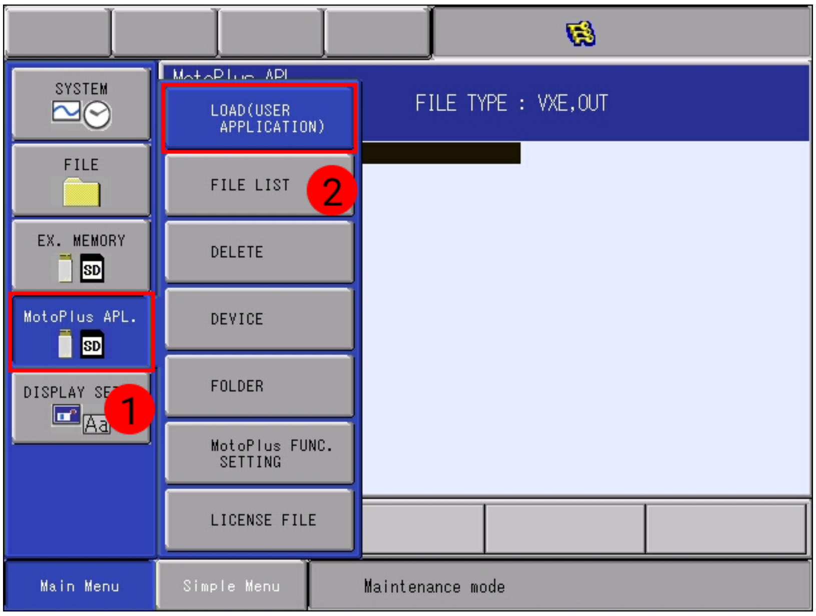




TARGET DEVICE

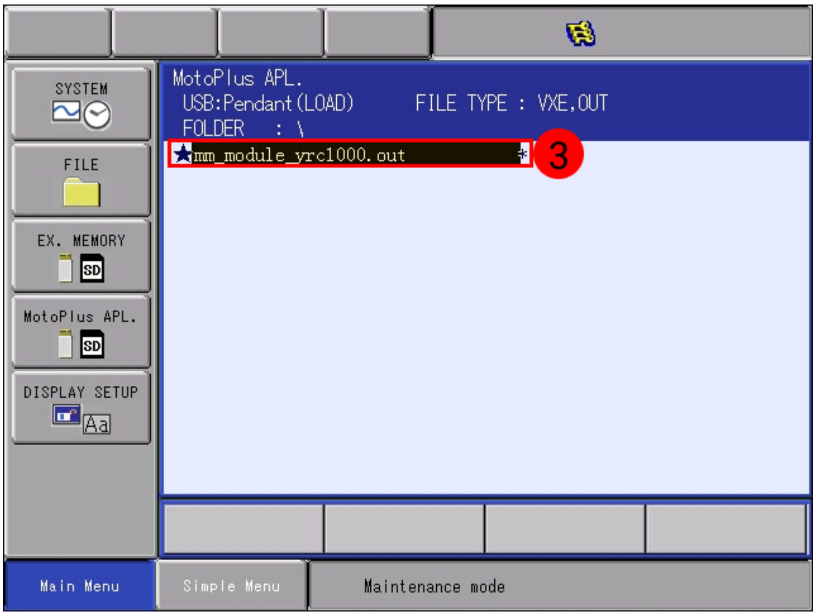
SD:Pendant
USB:Pendant
USB1:Controller

2. Select MotoPlus APL. › LOAD(USER APPLICATION). Select **mm_module_yrc1000.out**, press the ENTER key on the teach pendant, and click **[YES]** in the pop-up window to start loading the program.

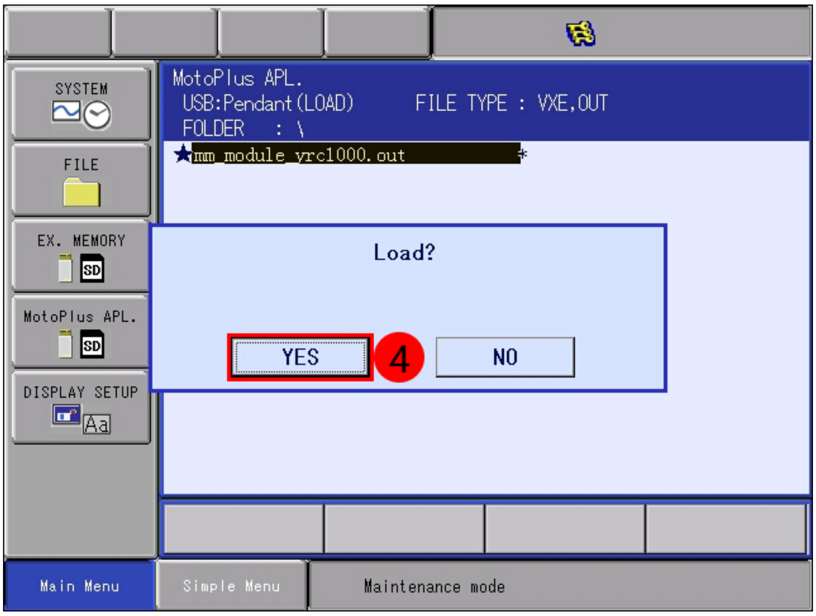


MotoPlus APL.

FILE TYPE : VXE,OUT



MotoPlus APL.
USB:Pendant(LOAD) FILE TYPE : VXE,OUT
FOLDER : \
★mm_module_yrc1000.out

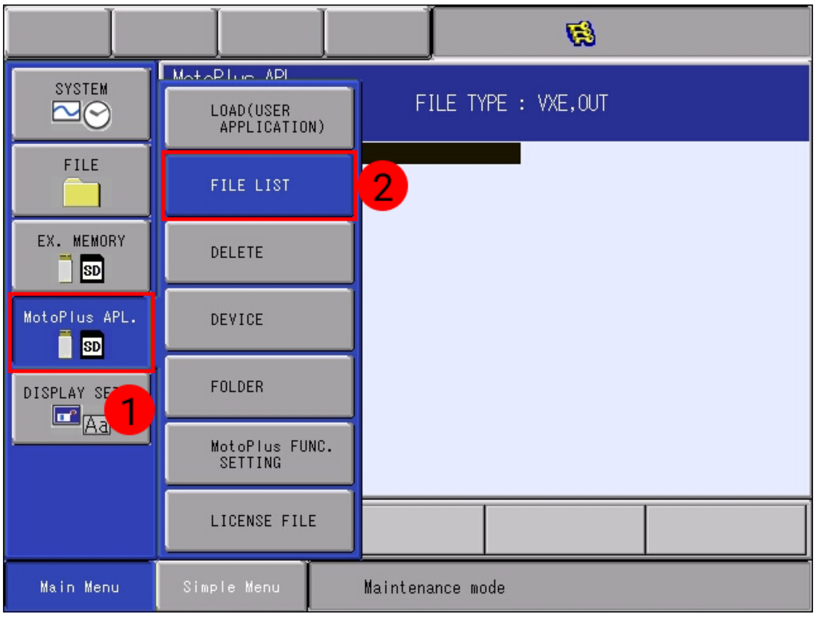


MotoPlus APL.
USB:Pendant(LOAD)   FILE TYPE : VXE,OUT
FOLDER   : \
★mm_module_yrc1000.out

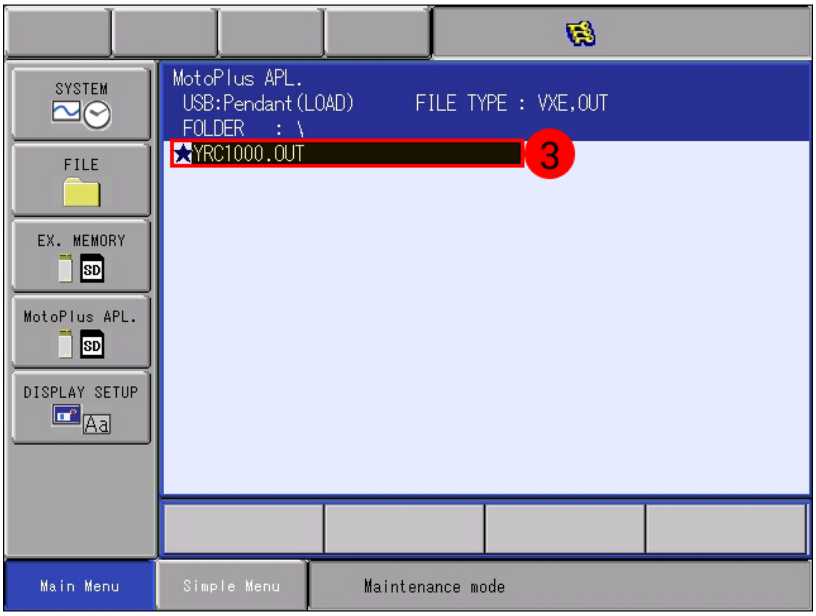
Load?

If you are using the controller DX200, you need to select **mm_module_dx200.out** here.

3. After the loading completes, select MotoPlus APL. › FILE LIST. If you can see the backend program file (YRC1000.OUT), the loading of the backend program file is successful.



FILE TYPE : VXE,OUT



```
MotoPlus APL.
 USB:Pendant(LOAD)      FILE TYPE : VXE,OUT
 FOLDER    : \
★YRC1000.OUT
```

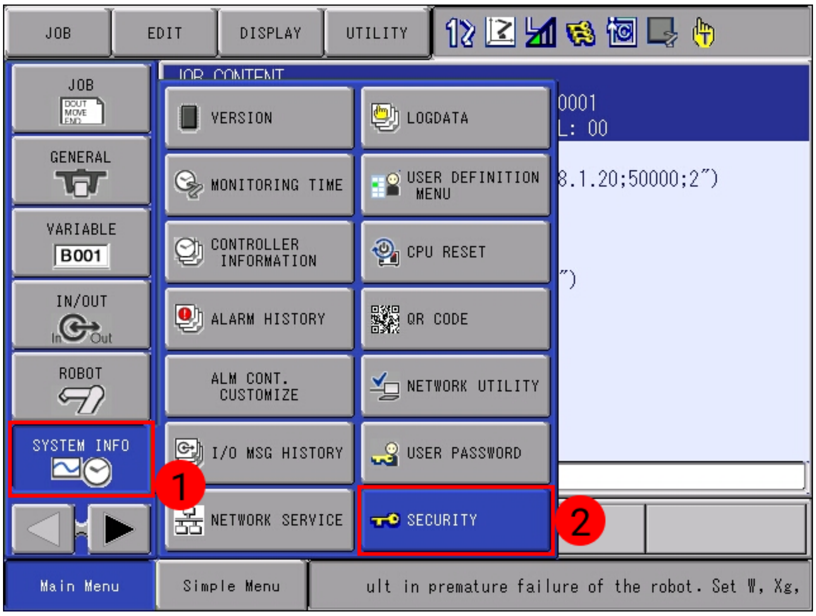
If you are using the controller DX200, you will see **DX200.OUT** here after the loading succeeds.

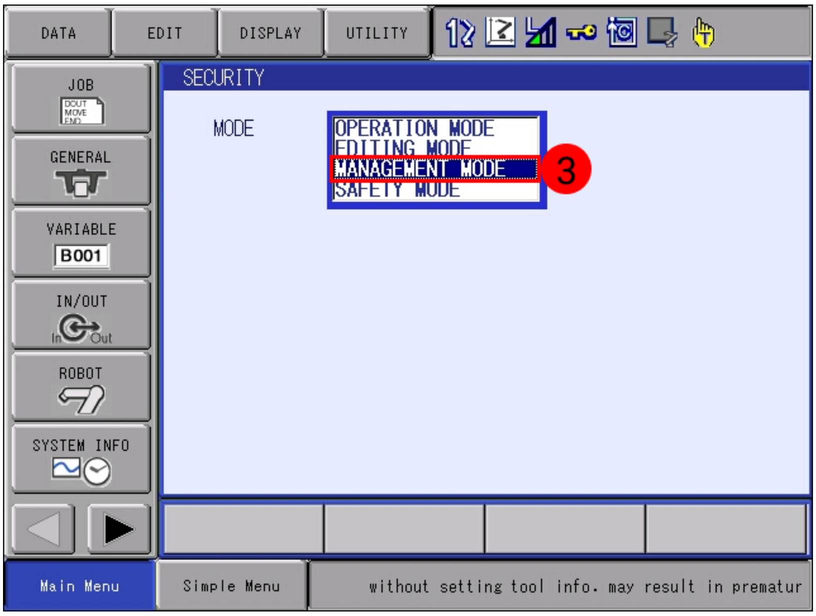
4. After you have loaded the backend program file, reboot the robot to enter the **online** mode.

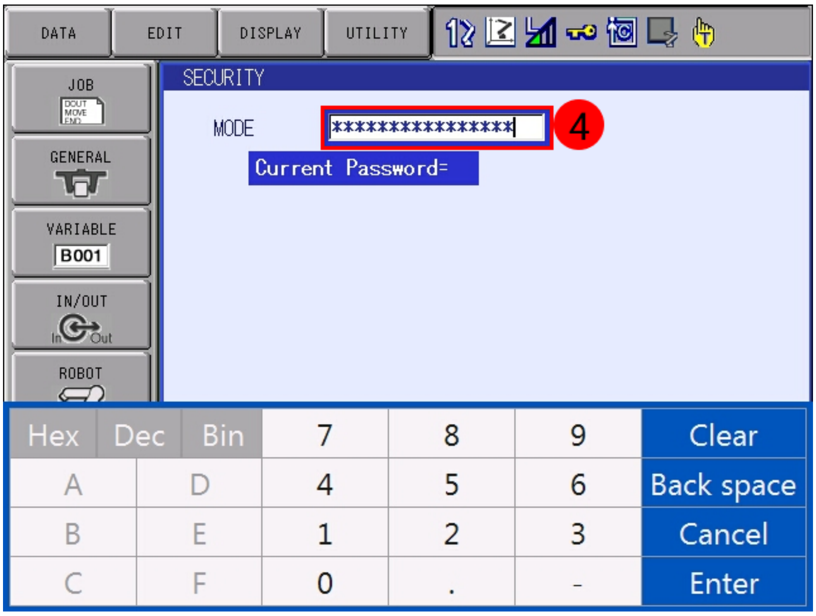
You need to load the frontend program files and the example program files under the online mode. Therefore, you need to perform this operation after loading the backend program file.

Load the Frontend Program Files

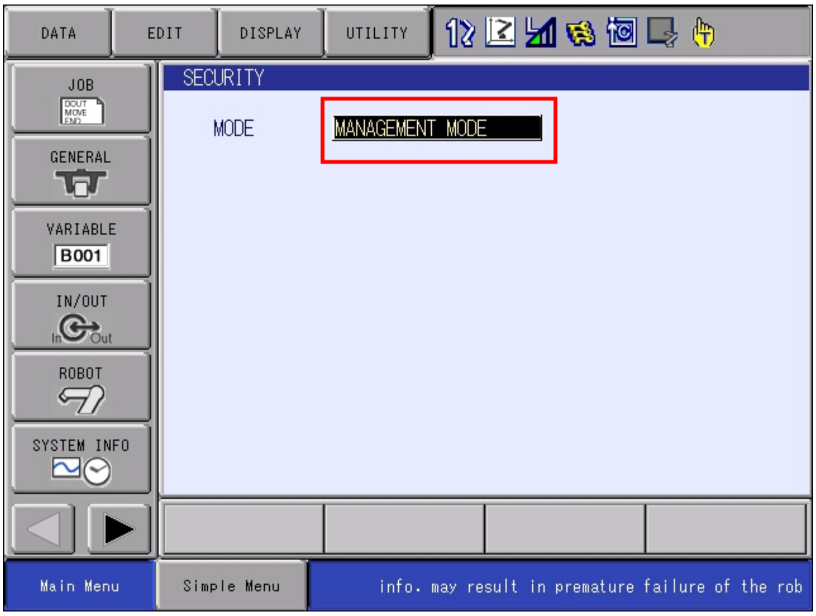
1. Under the **online** mode, select SYSTEM INFO › SECURITY on the **Main Menu**. Select MANAGEMENT MODE in the drop-down menu, and input the default password (sixteen 9s).







Press Enter in the lower-right corner of the teach pendant screen to enter the management mode.

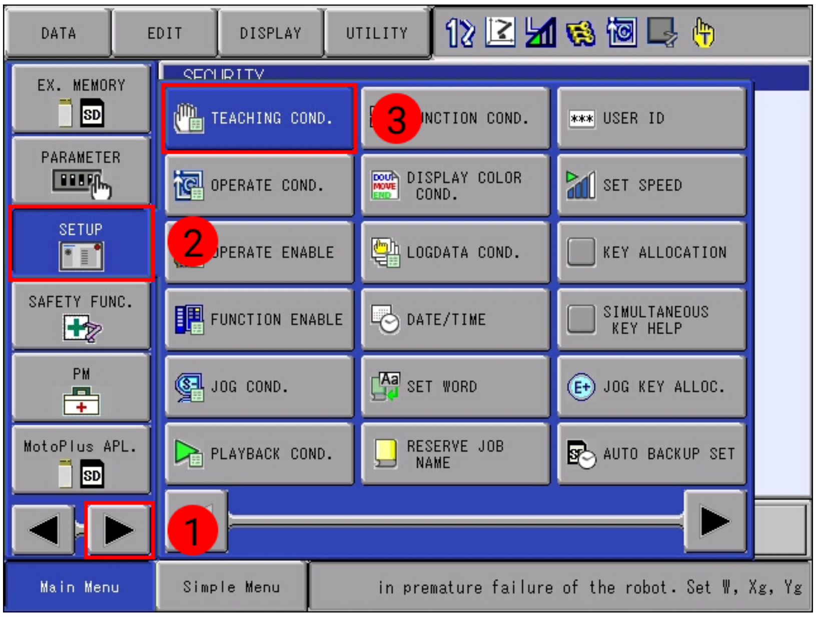


SECURITY

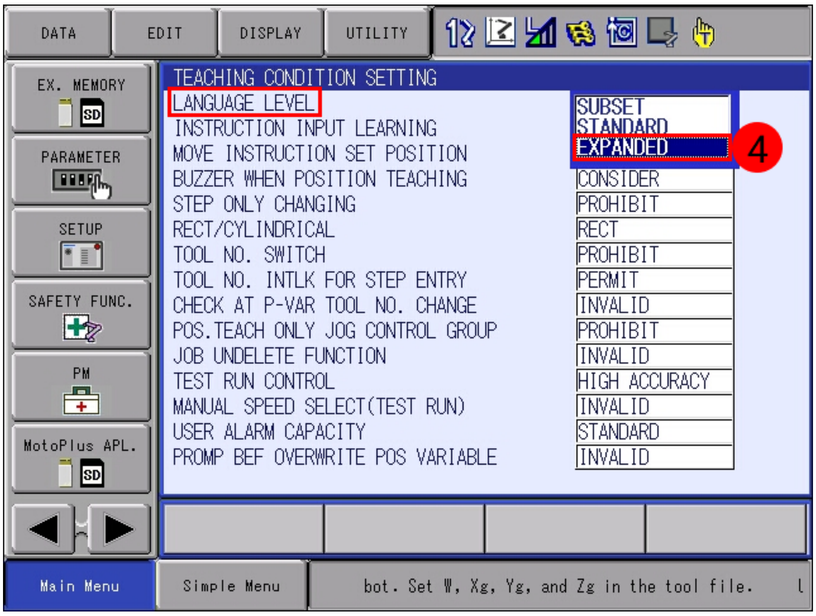
MODE MANAGEMENT MODE

info. may result in premature failure of the rob

2. Press the **right-arrow** ▶ button in the lower-left corner of the teach pendant screen. Then select SETUP › TEACHING COND, and select **EXPANDED** in the drop-down menu of LANGUAGE LEVEL.



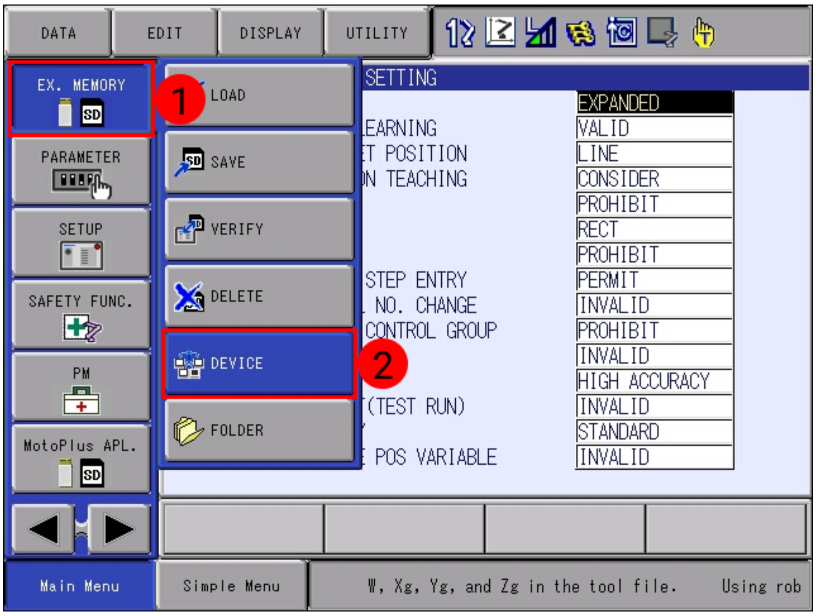
in premature failure of the robot. Set W, Xg, Yg

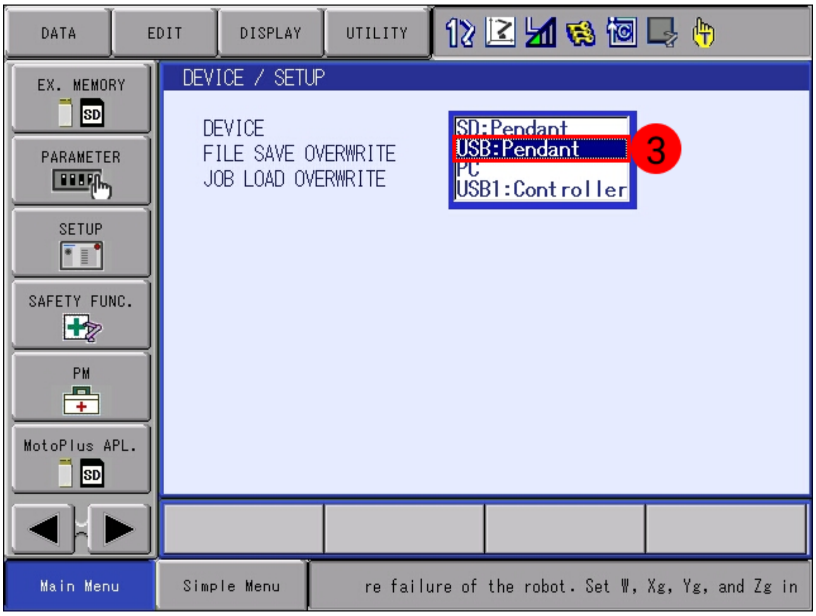


TEACHING CONDITION SETTING

LANGUAGE LEVEL	EXPANDED
INSTRUCTION INPUT LEARNING	
MOVE INSTRUCTION SET POSITION	
BUZZER WHEN POSITION TEACHING	CONSIDER
STEP ONLY CHANGING	PROHIBIT
RECT/CYLINDRICAL	RECT
TOOL NO. SWITCH	PROHIBIT
TOOL NO. INTLK FOR STEP ENTRY	PERMIT
CHECK AT P-VAR TOOL NO. CHANGE	INVALID
POS.TEACH ONLY JOG CONTROL GROUP	PROHIBIT
JOB UNDELETE FUNCTION	INVALID
TEST RUN CONTROL	HIGH ACCURACY
MANUAL SPEED SELECT(TEST RUN)	INVALID
USER ALARM CAPACITY	STANDARD
PROMP BEF OVERWRITE POS VARIABLE	INVALID

3. Select EX. MEMORY › DEVICE, and then select **USB: Pendant for** in the drop-down menu of DEVICE.

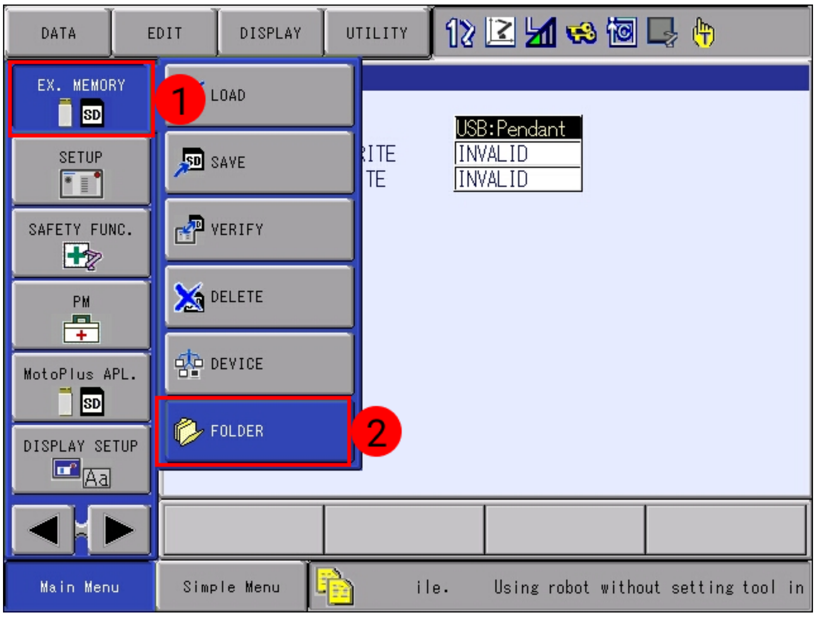


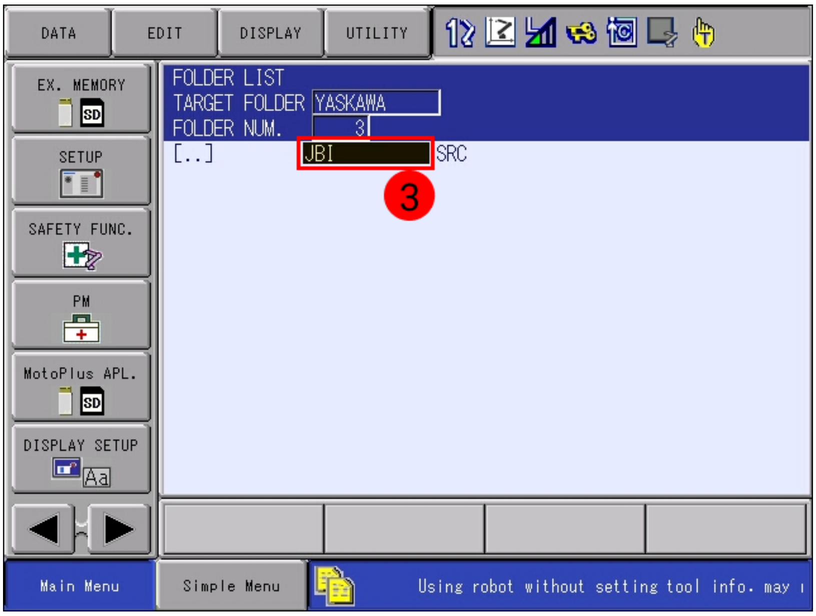


DEVICE / SETUP

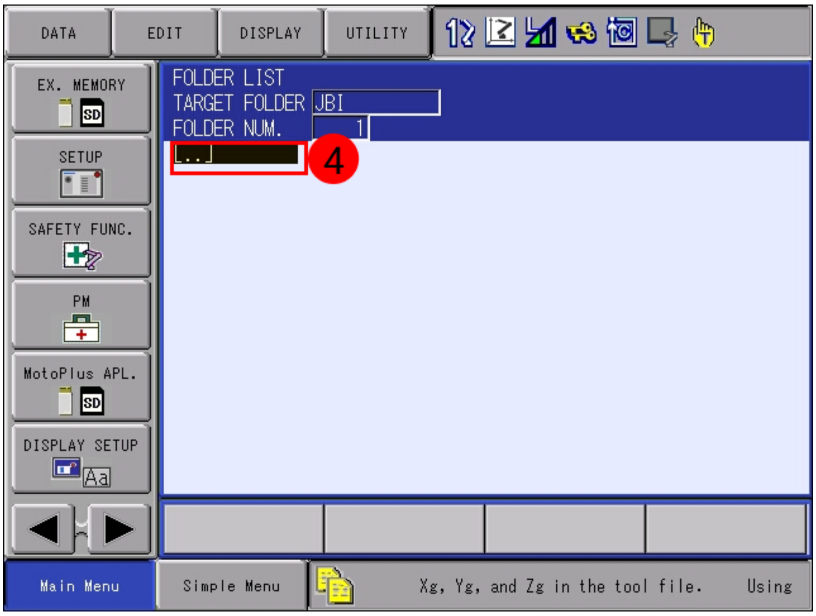
DEVICE	SD:Pendant USB:Pendant PC USB1:Controller
FILE SAVE OVERWRITE	
JOB LOAD OVERWRITE	

4. Select EX. MEMORY › FOLDER. In the FOLDER LIST, select and enter the **JBI** folder.



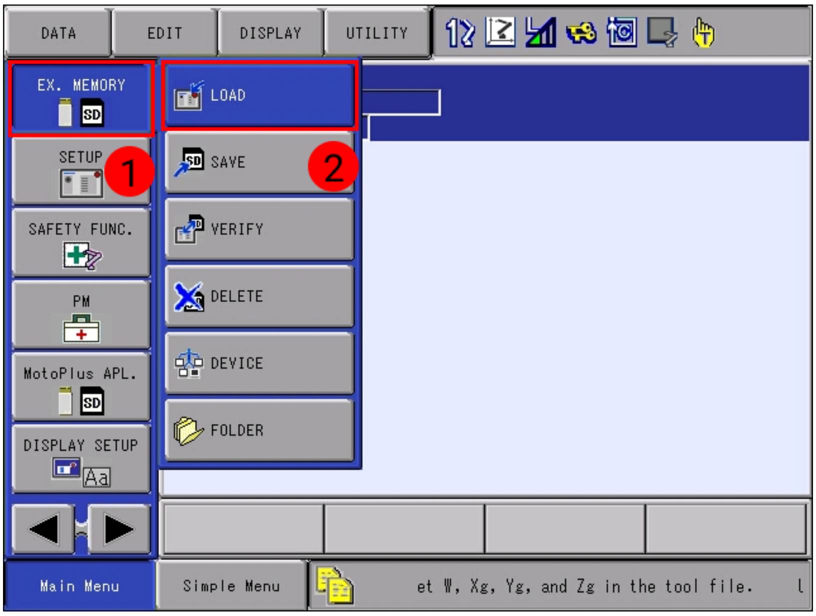

FOLDER LIST
TARGET FOLDER YASKAWA
FOLDER NUM. 3
[..] JBI SRC

Using robot without setting tool info. may

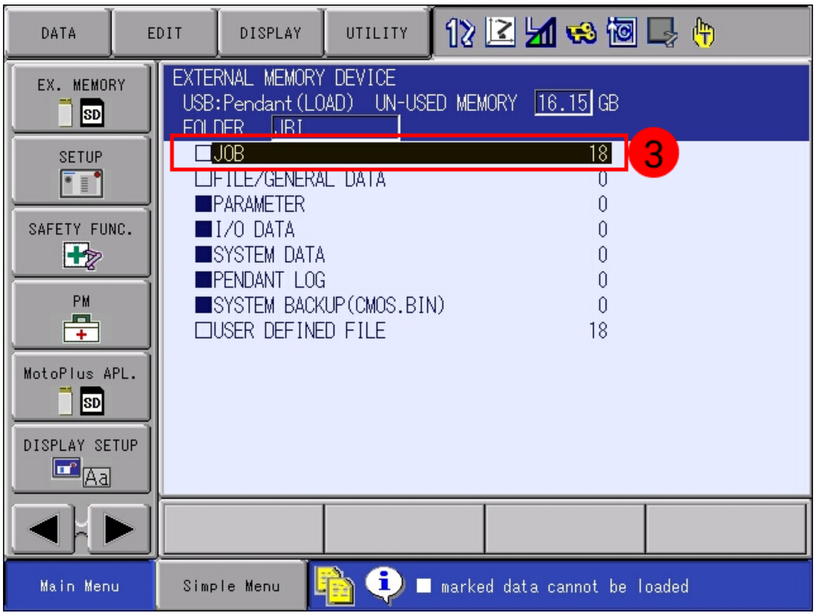

FOLDER LIST
TARGET FOLDER JBI
FOLDER NUM. 1
[..]

Xg, Yg, and Zg in the tool file. Using

5. Select EX. MEMORY › LOAD, and select **JOB**. A list of jobs to load will be displayed on the screen.

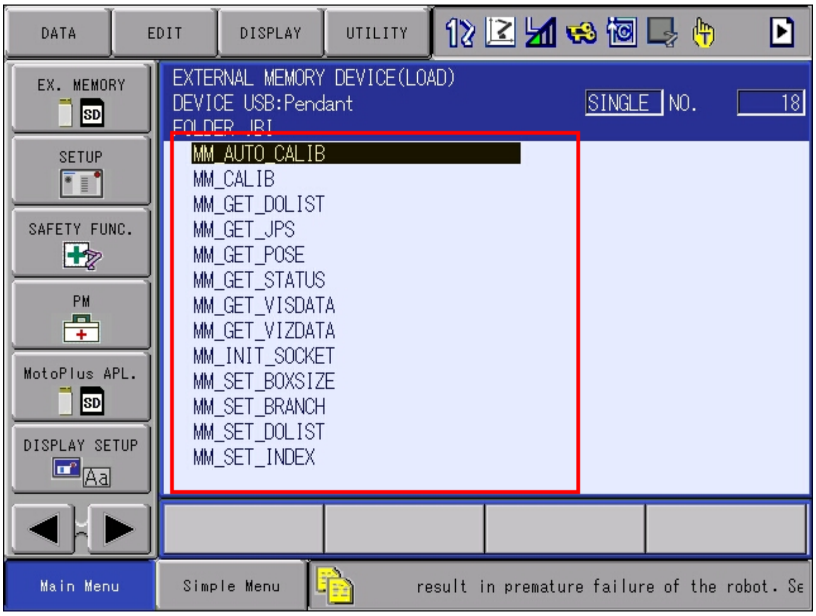


et W, Xg, Yg, and Zg in the tool file.



```
EXTERNAL MEMORY DEVICE
 USB:Pendant(LOAD)  UN-USED MEMORY 16.15 GB
 FOLDER   JBI
  □JOB                                    18
  □FILE/GENERAL DATA                       0
  ■PARAMETER                               0
  ■I/O DATA                                0
  ■SYSTEM DATA                             0
  ■PENDANT LOG                             0
  ■SYSTEM BACKUP(CMOS.BIN)                 0
  □USER DEFINED FILE                      18
```

■ marked data cannot be loaded

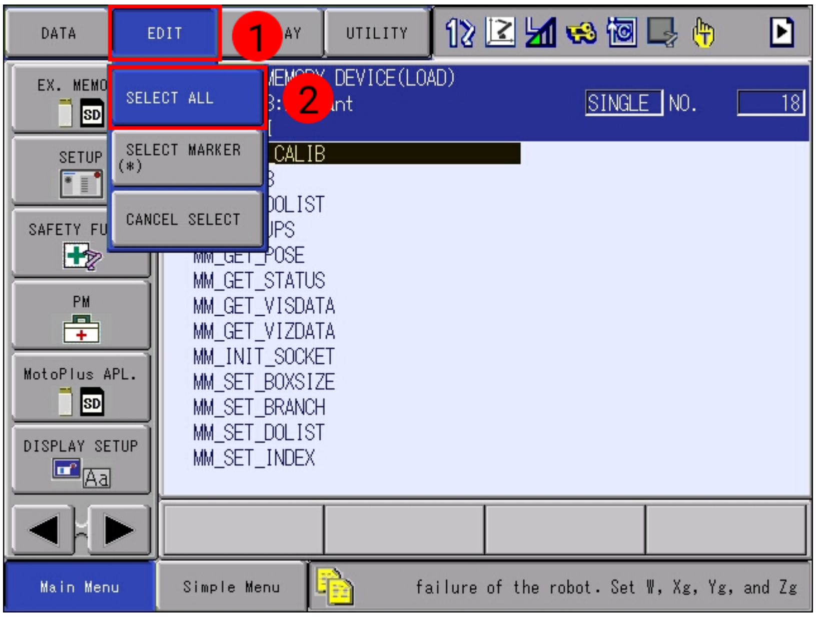


EXTERNAL MEMORY DEVICE(LOAD)
DEVICE USB:Pendant
FOLDER JBI

MM_AUTO_CALIB
MM_CALIB
MM_GET_DOLIST
MM_GET_JPS
MM_GET_POSE
MM_GET_STATUS
MM_GET_VISDATA
MM_GET_VIZDATA
MM_INIT_SOCKET
MM_SET_BOXSIZE
MM_SET_BRANCH
MM_SET_DOLIST
MM_SET_INDEX

result in premature failure of the robot. Se

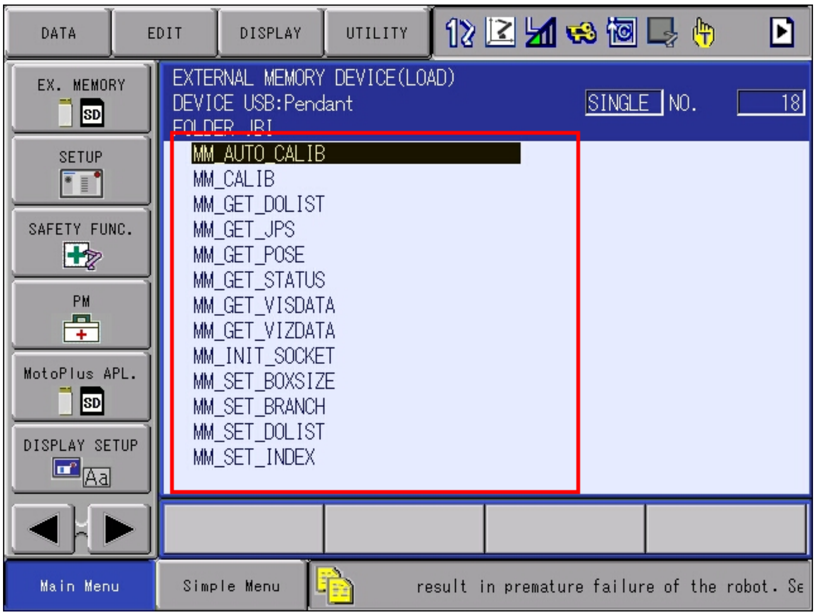
6. Select EDIT › SELECT ALL.



MEMORY DEVICE(LOAD)
B: nt

CALIB
B
DOLIST
JPS
MM_GET_POSE
MM_GET_STATUS
MM_GET_VISDATA
MM_GET_VIZDATA
MM_INIT_SOCKET
MM_SET_BOXSIZE
MM_SET_BRANCH
MM_SET_DOLIST
MM_SET_INDEX

failure of the robot. Set W, Xg, Yg, and Zg

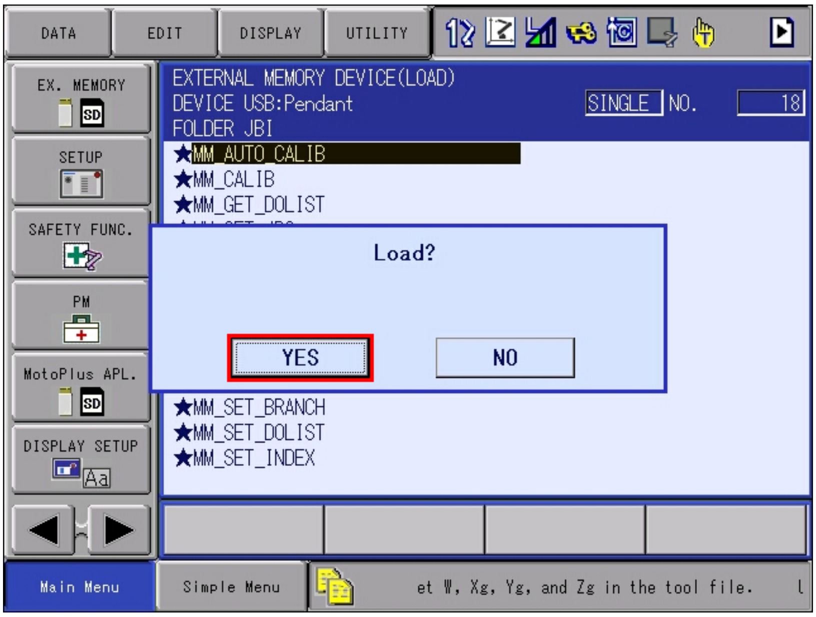


EXTERNAL MEMORY DEVICE(LOAD)
DEVICE USB:Pendant
FOLDER JBI

MM_AUTO_CALIB
MM_CALIB
MM_GET_DOLIST
MM_GET_JPS
MM_GET_POSE
MM_GET_STATUS
MM_GET_VISDATA
MM_GET_VIZDATA
MM_INIT_SOCKET
MM_SET_BOXSIZE
MM_SET_BRANCH
MM_SET_DOLIST
MM_SET_INDEX

result in premature failure of the robot. Se

7. Press the ENTER key on the teach pendant. Select **YES** in the pop-up dialog box to start loading the programs.



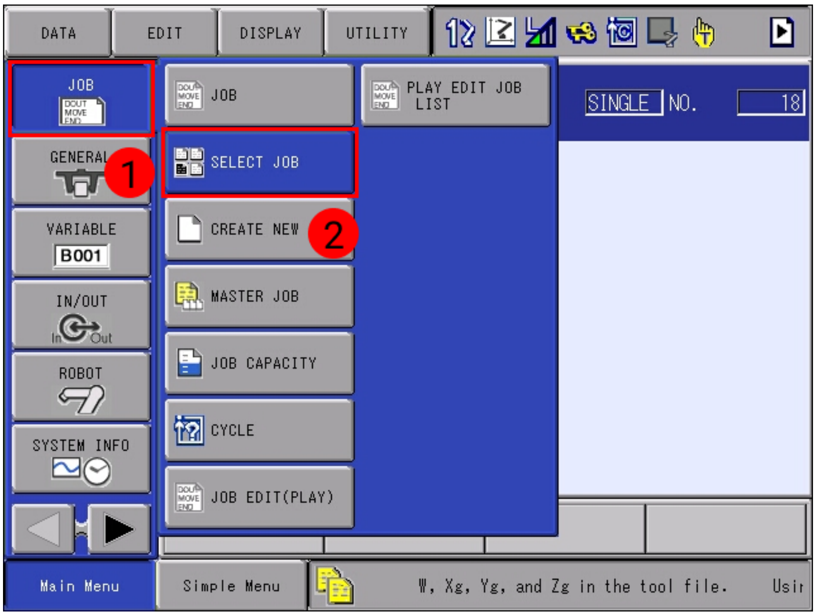
EXTERNAL MEMORY DEVICE(LOAD)
DEVICE USB:Pendant
FOLDER JBI

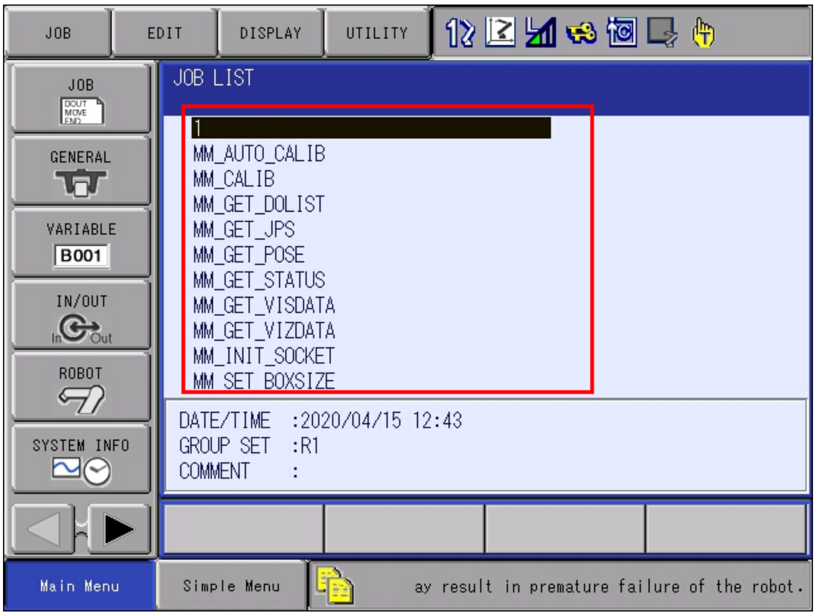
★MM_AUTO_CALIB
★MM_CALIB
★MM_GET_DOLIST
★MM_SET_BRANCH
★MM_SET_DOLIST
★MM_SET_INDEX

Load?

et W, Xg, Yg, and Zg in the tool file. l

8. After the loading completes, select JOB › SELECT JOB. If you can see all the jobs in the JOB LIST, the loading of frontend program files is successful.





JOB LIST

```
1
MM_AUTO_CALIB
MM_CALIB
MM_GET_DOLIST
MM_GET_JPS
MM_GET_POSE
MM_GET_STATUS
MM_GET_VISDATA
MM_GET_VIZDATA
MM_INIT_SOCKET
MM_SET_BOXSIZE
```

```
DATE/TIME  :2020/04/15 12:43
GROUP SET  :R1
COMMENT    :
```

Load the Example Program File

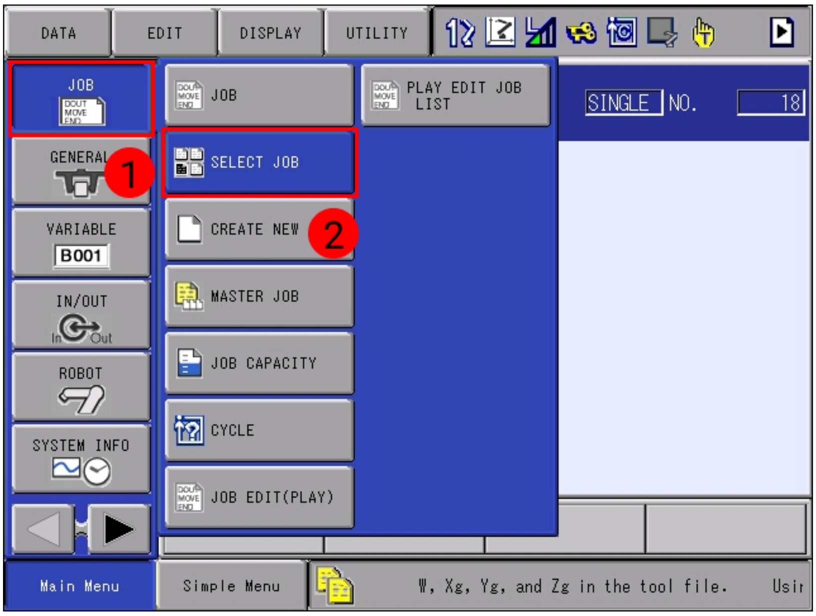
Repeat the loading steps in the section “Load the Frontend Program Files” to load the example program file.

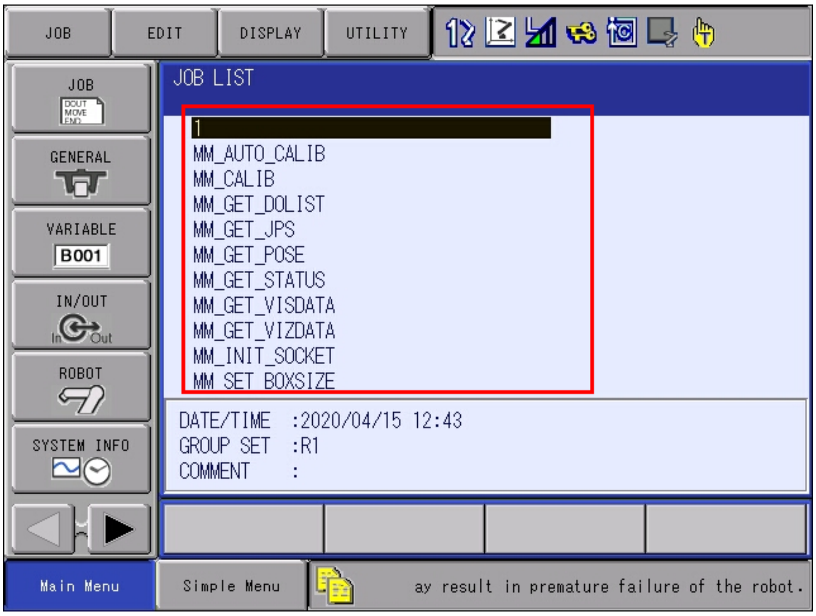
Please note that you need to select the **SAMPLE** folder when selecting a folder to load.

Test Standard Interface Communication

When you test the Standard Interface communication between the robot and the vision system, first make sure that the interface service is started, and then run a test program on the teach pendant to test the Standard Interface communication. You can directly use the loaded calibration program (MM_AUTO_CALIB) as the test program.

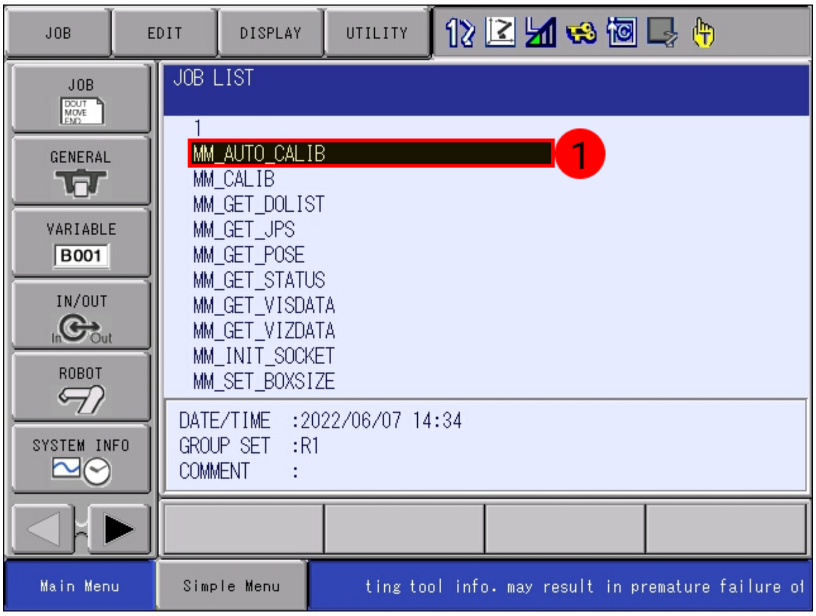
1. On the teach pendant, select JOB › SELECT JOB on the **Main Menu** to enter the JOB LIST interface.





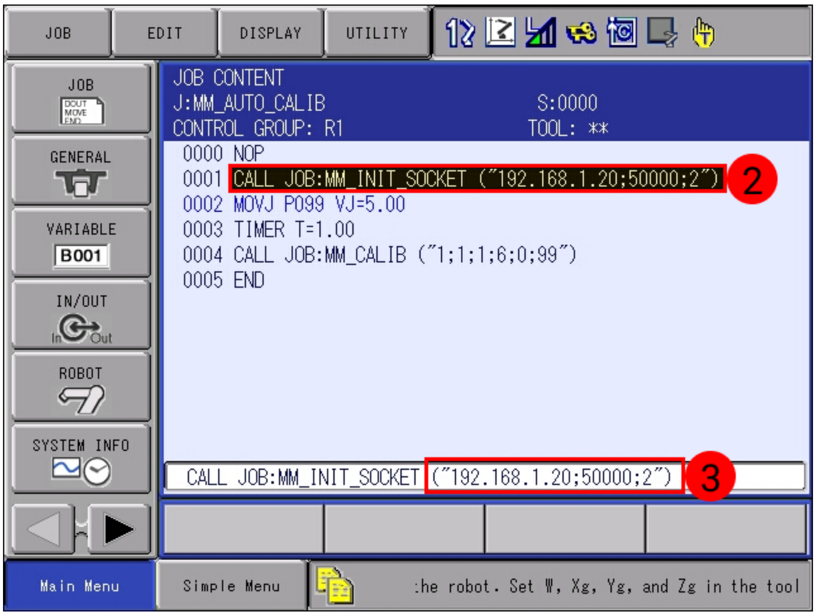
```
JOB LIST
1
MM_AUTO_CALIB
MM_CALIB
MM_GET_DOLIST
MM_GET_JPS
MM_GET_POSE
MM_GET_STATUS
MM_GET_VISDATA
MM_GET_VIZDATA
MM_INIT_SOCKET
MM_SET_BOXSIZE

DATE/TIME  :2020/04/15 12:43
GROUP SET  :R1
COMMENT    :
```

2. Select **MM_AUTO_CALIB** in the JOB LIST, and press the SELECT key to open the job. Move the cursor to line 0001, press the SELECT key. In the text box at the bottom, move the cursor to the IP address and port number, and press the ENTER key. Change the IP address and port number to the actual ones of the IPC, and press Enter on the screen.



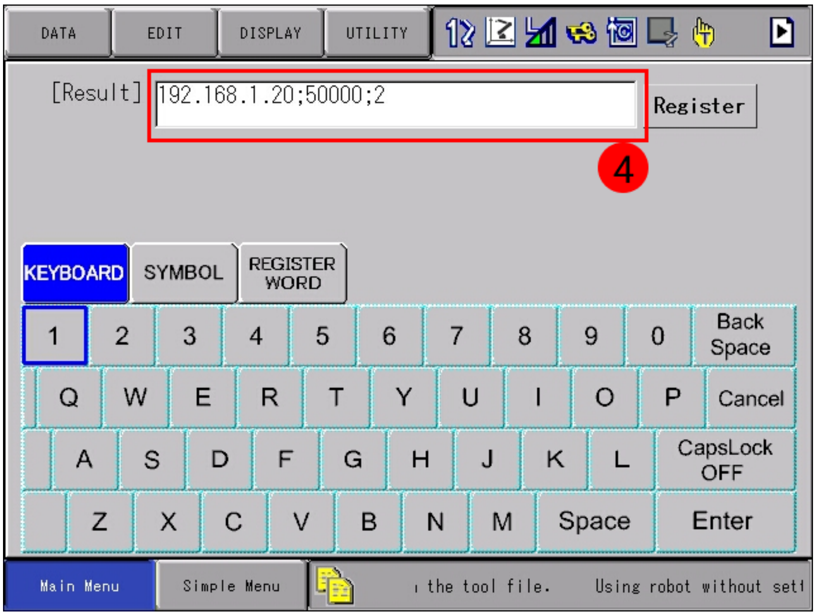
```
JOB LIST
 1
 MM_AUTO_CALIB
 MM_CALIB
 MM_GET_DOLIST
 MM_GET_JPS
 MM_GET_POSE
 MM_GET_STATUS
 MM_GET_VISDATA
 MM_GET_VIZDATA
 MM_INIT_SOCKET
 MM_SET_BOXSIZE
DATE/TIME  :2022/06/07 14:34
GROUP SET  :R1
COMMENT    :
```

ting tool info. may result in premature failure of



```
JOB CONTENT
J:MM_AUTO_CALIB                    S:0000
CONTROL GROUP: R1                TOOL: **
 0000 NOP
 0001 CALL JOB:MM_INIT_SOCKET ("192.168.1.20;50000;2")
 0002 MOVJ P099 VJ=5.00
 0003 TIMER T=1.00
 0004 CALL JOB:MM_CALIB ("1;1;1;6;0;99")
 0005 END

 CALL JOB:MM_INIT_SOCKET ("192.168.1.20;50000;2")
```

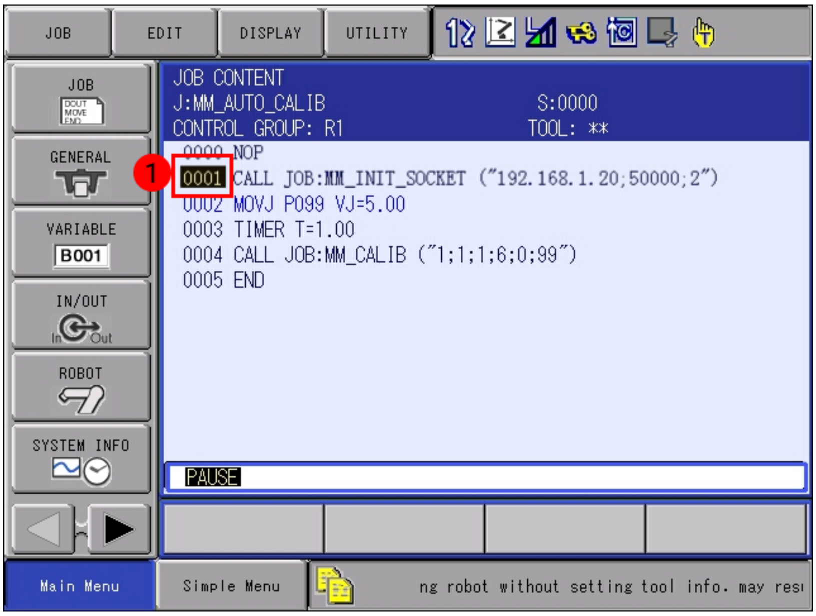
:he robot. Set W, Xg, Yg, and Zg in the tool


DATA
EDIT
DISPLAY
UTILITY
Register
4
KEYBOARD
SYMBOL
REGISTER WORD
1 2 3 4 5 6 7 8 9 0 Back Space
Q W E R T Y U I O P Cancel
A S D F G H J K L CapsLock OFF
Z X C V B N M Space Enter
Main Menu
Simple Menu

[Result] 192.168.1.20;50000;2

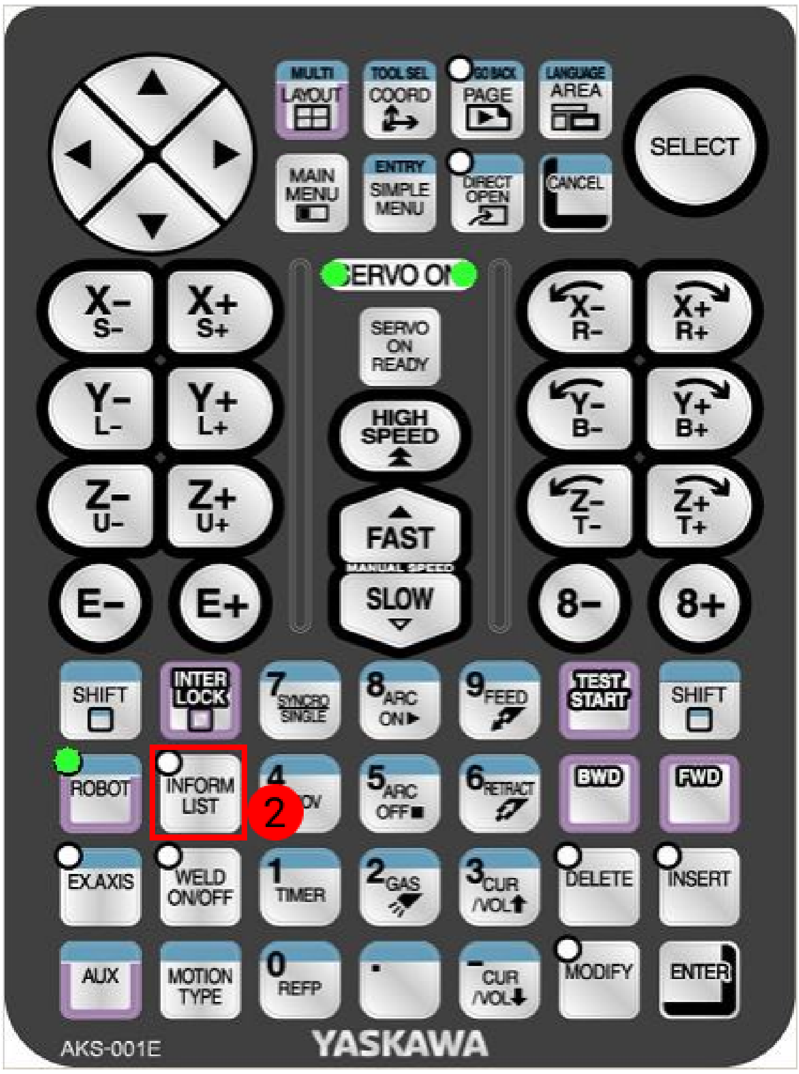
```
ı the tool file.    Using robot without setı
```

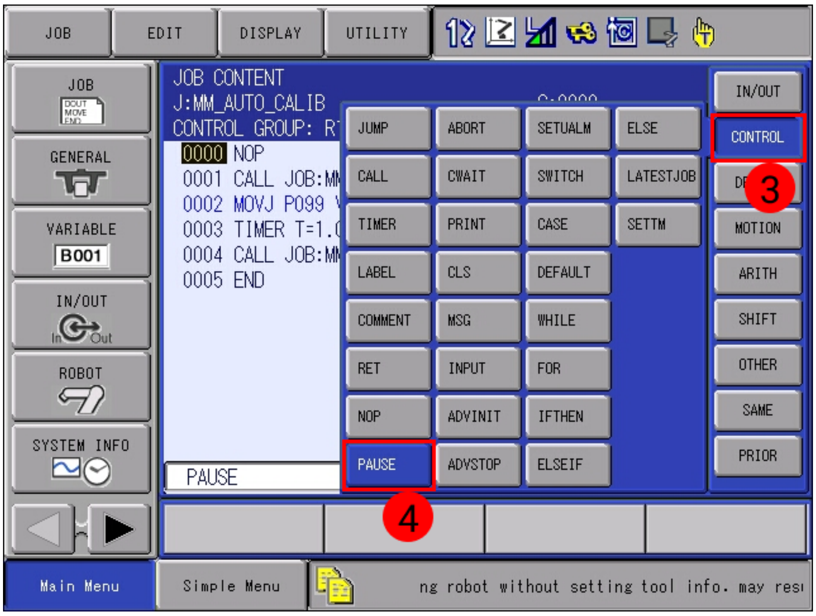
3. Move the cursor to line 0001, press the `INFORM LIST` key on the teach pendant. Select CONTROL › PAUSE in the pop-up menu of the screen, and then press `INSERT` and `ENTER` keys on the teach pendant. This step will insert a **PAUSE** command after line 0001.



```
JOB CONTENT
J:MM_AUTO_CALIB                    S:0000
CONTROL GROUP: R1                  TOOL: **
0000 NOP
0001 CALL JOB:MM_INIT_SOCKET ("192.168.1.20;50000;2")
0002 MOVJ P099 VJ=5.00
0003 TIMER T=1.00
0004 CALL JOB:MM_CALIB ("1;1;1;6;0;99")
0005 END

PAUSE
```

```
ng robot without setting tool info. may resı
```





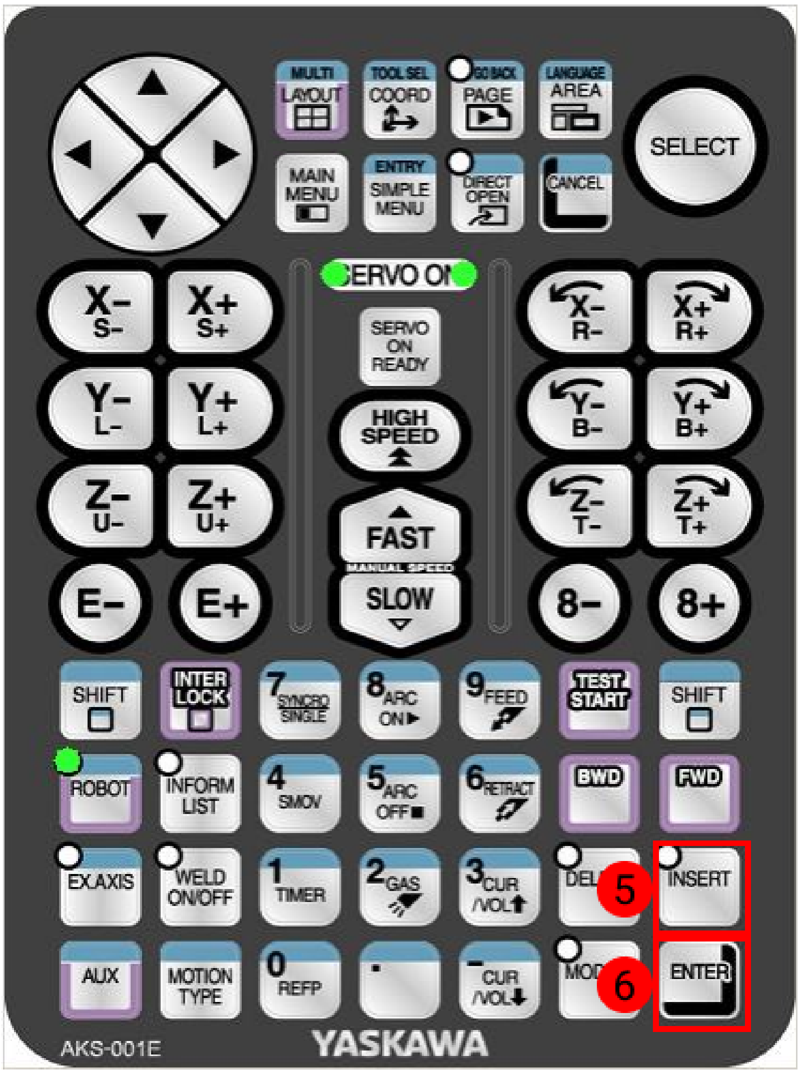
JOB | EDIT | DISPLAY | UTILITY

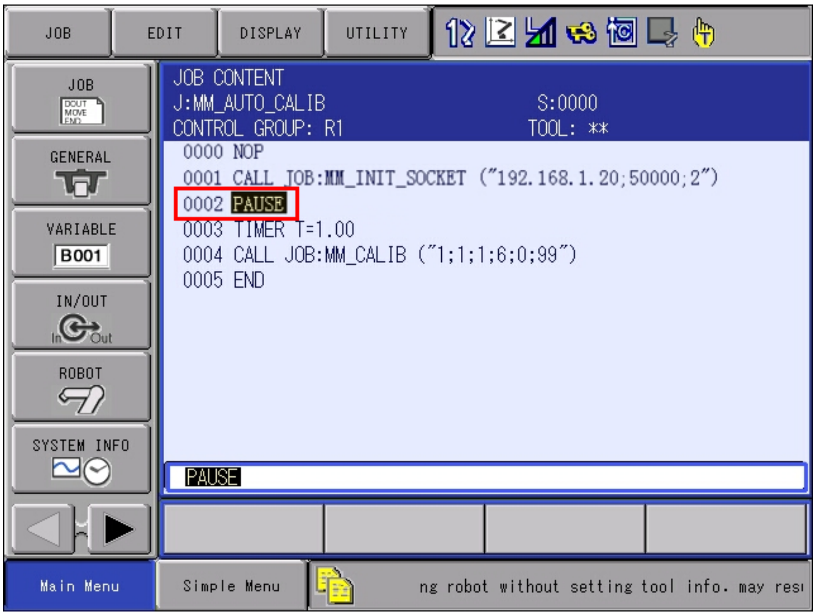
JOB CONTENT
J:MM_AUTO_CALIB
CONTROL GROUP: R1

```
0000 NOP
0001 CALL JOB:MM
0002 MOVJ P099 V
0003 TIMER T=1.0
0004 CALL JOB:MM
0005 END
```

PAUSE

ng robot without setting tool info. may resu





```
JOB CONTENT
J:MM_AUTO_CALIB                    S:0000
CONTROL GROUP: R1                  TOOL: **
 0000 NOP
 0001 CALL JOB:MM_INIT_SOCKET ("192.168.1.20;50000;2")
 0002 PAUSE
 0003 TIMER T=1.00
 0004 CALL JOB:MM_CALIB ("1;1;1;6;0;99")
 0005 END
```

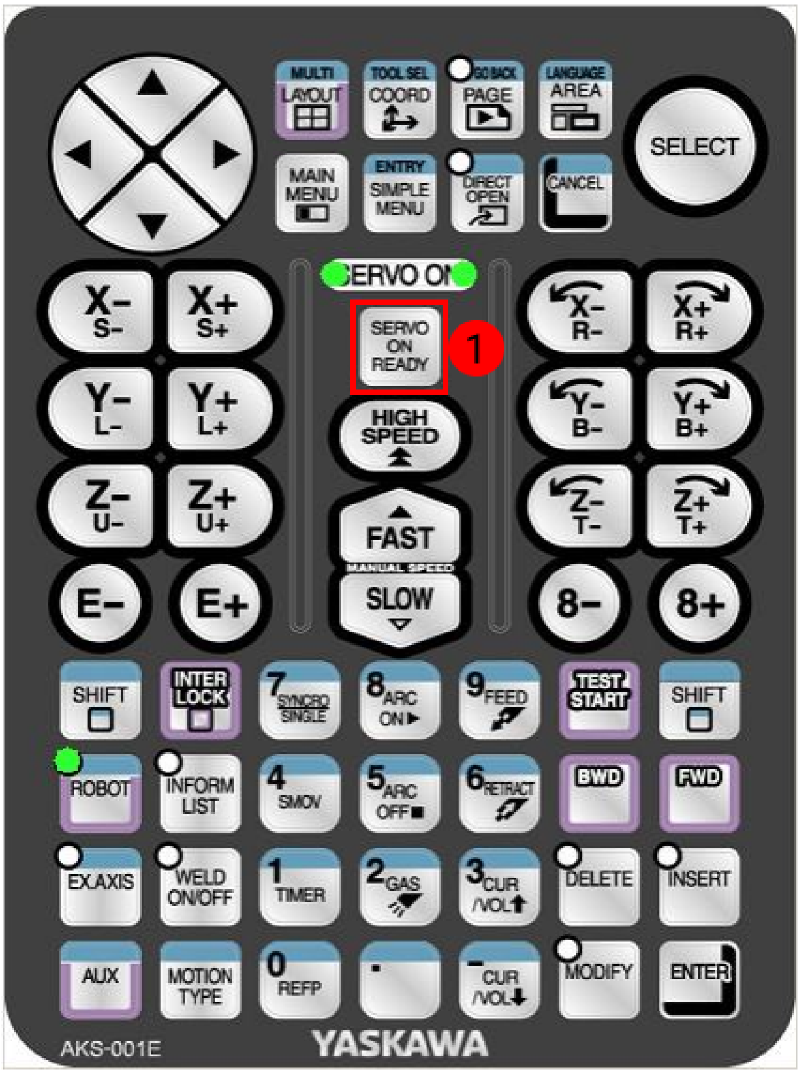
PAUSE

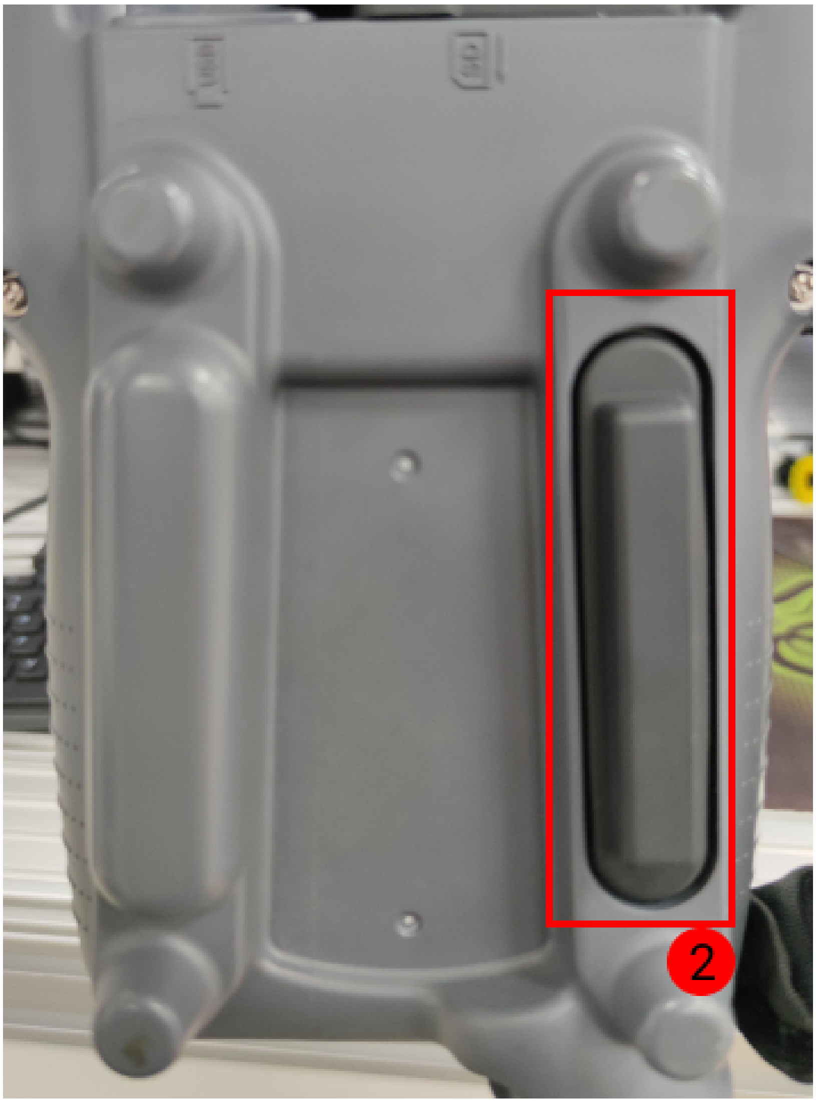
ng robot without setting tool info. may resu

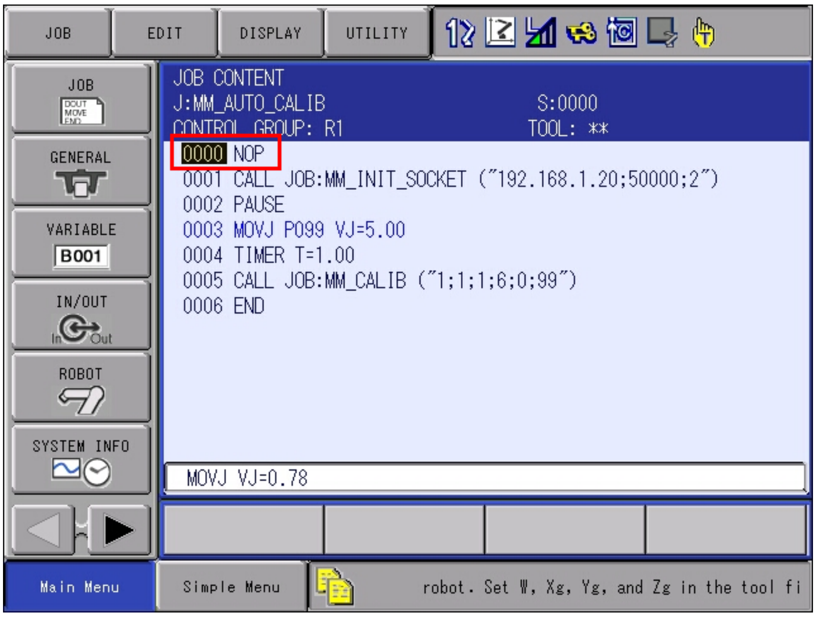
4. Twist the mode key on the teach pendant to TEACH to enter the **TEACH** mode.

5. On the teach pendant, press the SERVO ON READY key, and then hold down the enable

switch on the back while moving the cursor back to line 0000 on the screen.





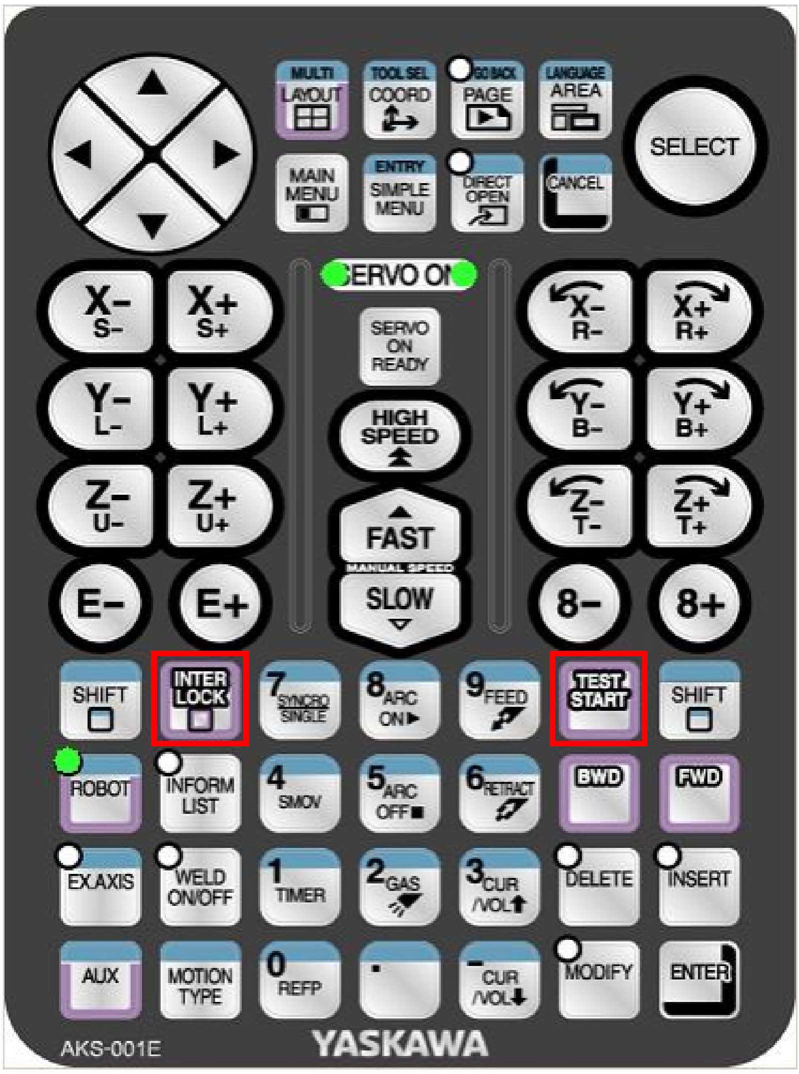


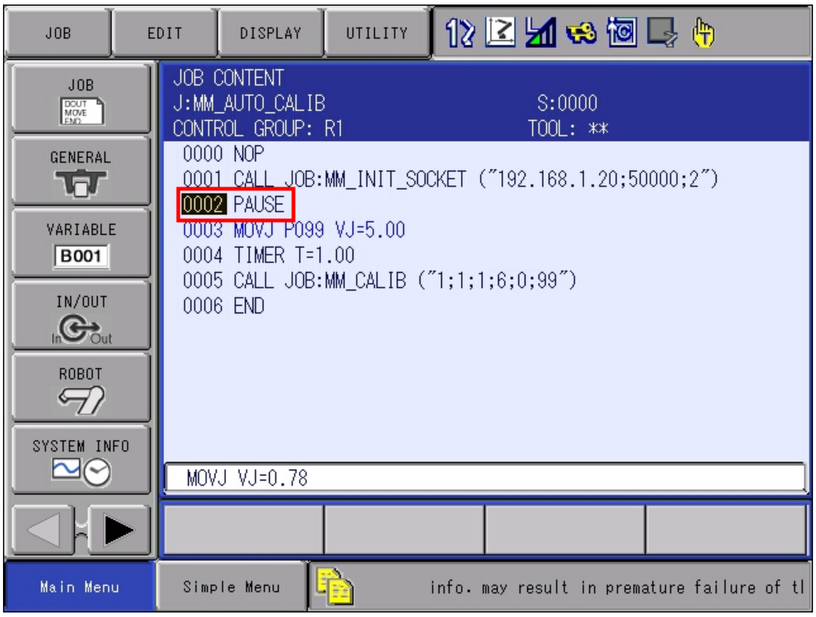
```
JOB CONTENT
J:MM_AUTO_CALIB                         S:0000
CONTROL GROUP: R1                       TOOL: **
 0000 NOP
 0001 CALL JOB:MM_INIT_SOCKET ("192.168.1.20;50000;2")
 0002 PAUSE
 0003 MOVJ P099 VJ=5.00
 0004 TIMER T=1.00
 0005 CALL JOB:MM_CALIB ("1;1;1;6;0;99")
 0006 END

 MOVJ VJ=0.78
```

robot. Set W, Xg, Yg, and Zg in the tool fi

This step ensures that the job runs from the first line.

6. Press and hold down the INTER LOCK and TEST STAR keys on the teach pendant at the same time. After the robot executes line 0001, the cursor will automatically stop at line 0002 where the PAUSE command resides.

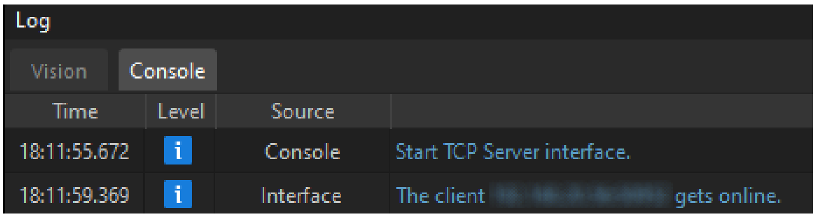




```
JOB CONTENT
J:MM_AUTO_CALIB                    S:0000
CONTROL GROUP: R1                  TOOL: **
 0000 NOP
 0001 CALL JOB:MM_INIT_SOCKET ("192.168.1.20;50000;2")
 0002 PAUSE
 0003 MOVJ P099 VJ=5.00
 0004 TIMER T=1.00
 0005 CALL JOB:MM_CALIB ("1;1;1;6;0;99")
 0006 END

 MOVJ VJ=0.78
```

info. may result in premature failure of tl

7. Check whether logs will be displayed on the Console tab of the Mech-Vision Log panel. If logs as below are displayed, the YASKAWA robot can establish the Standard Interface communication with the vision system successfully.



```
Time          Level  Source     
18:11:55.672  i      Console    Start TCP Server interface.
18:11:59.369  i      Interface  The client [redacted] gets online.
```

- Delete the **PAUSE** command after testing the communication to avoid mistaken pause during calibration. To delete the **PAUSE** command, move the cursor to line 0002 where the **PAUSE** command resides, and then press the DELETE and ENTER keys in sequence on the teach pendant.

3.3. Hand-Eye Calibration

In this tutorial, you will perform the automatic hand-eye calibration in the eye-to-hand setup.

The hand-eye calibration establishes **the transformation relationship between the camera and robot reference frames (that is camera extrinsic parameters)**. With this relationship, the object pose determined by the vision system can be transformed into that in the robot reference frame, which guides the robot in performing its tasks.

▶ https://www.youtube.com/watch?v=w2vN5qXXJ8M/PLVcMd7cW2rXVtrAejMyVQni2dUDv8bxje *(YouTube video)*

Video Tutorial: Hand-Eye Calibration (Standard Interface Communication)

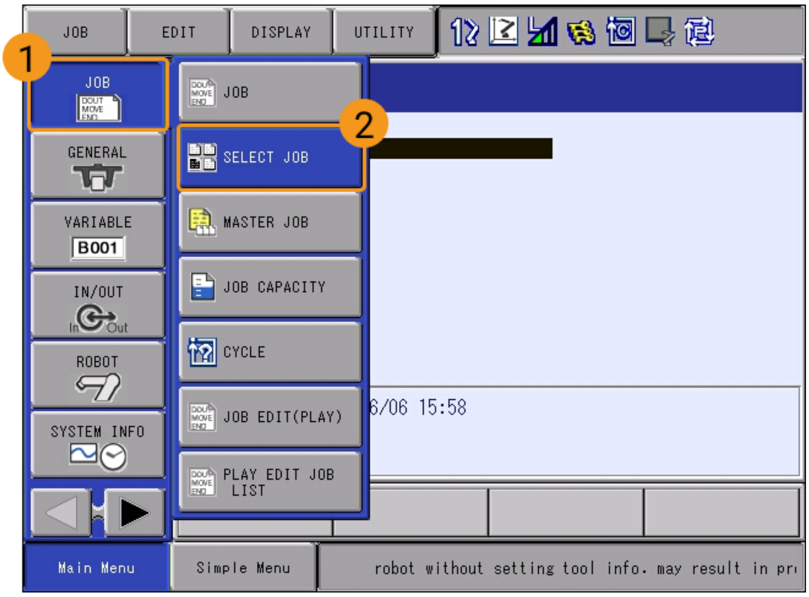
Preparation before Calibration

In this section, you will modify the robot calibration program, mount the calibration board, adjust the camera parameters, and complete the pre-calibration configuration.

Modify the Robot Calibration Program (MM_AUTO_CALIB)

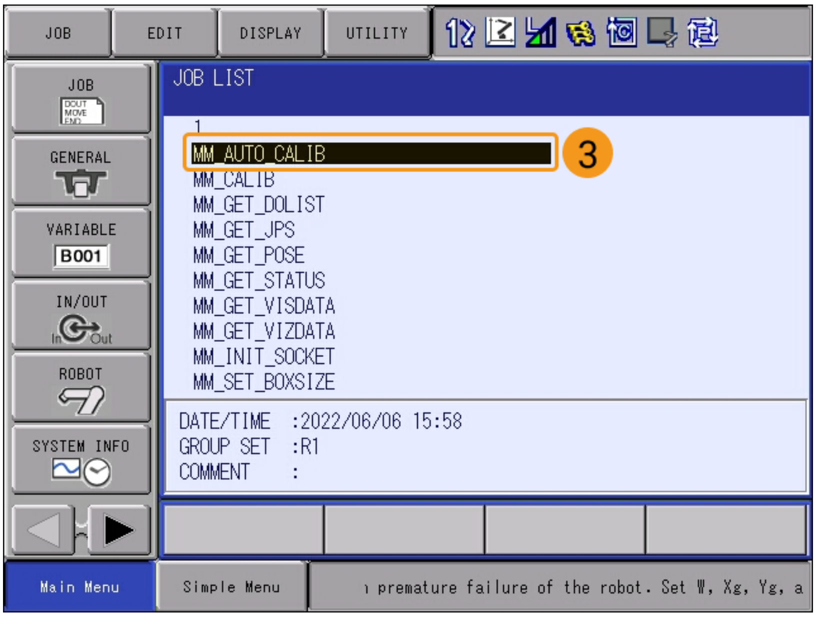
Select the Calibration Program

1. In the TEACH mode, select JOB › SELECT JOB. Move the cursor to **MM_AUTO_CALIB** in the JOB LIST and press the SELECT key on the teach pendant.



6/06 15:58

robot without setting tool info. may result in pr

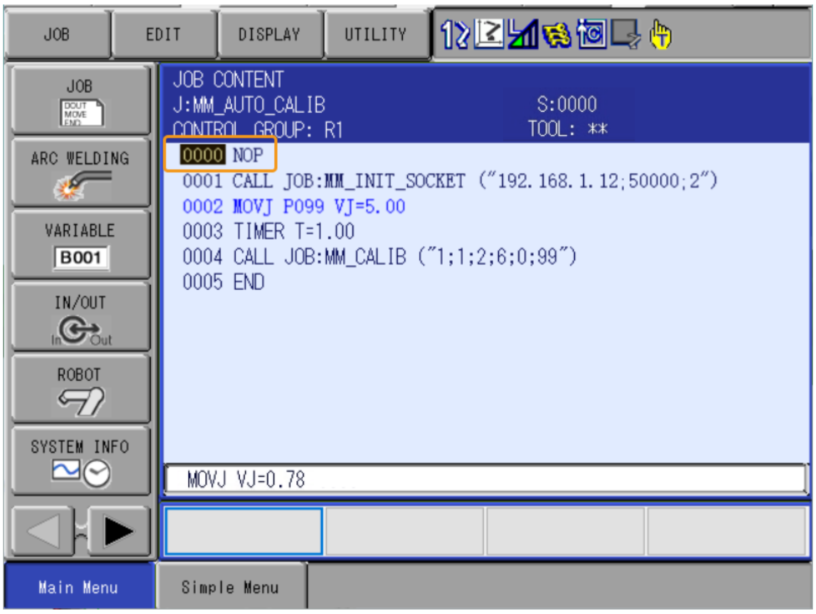


1
MM_AUTO_CALIB
MM_CALIB
MM_GET_DOLIST
MM_GET_JPS
MM_GET_POSE
MM_GET_STATUS
MM_GET_VISDATA
MM_GET_VIZDATA
MM_INIT_SOCKET
MM_SET_BOXSIZE

DATE/TIME :2022/06/06 15:58
GROUP SET :R1
COMMENT :

ı premature failure of the robot. Set W, Xg, Yg, a

The interface is as follows after the program file is selected.



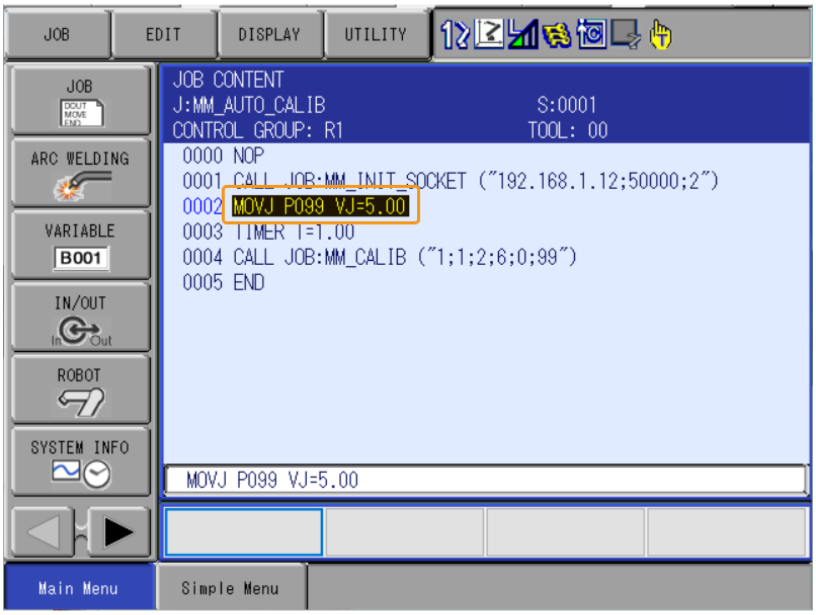
```
JOB CONTENT
J:MM_AUTO_CALIB                    S:0000
CONTROL GROUP: R1                  TOOL: **
0000 NOP
0001 CALL JOB:MM_INIT_SOCKET ("192.168.1.12;50000;2")
0002 MOVJ P099 VJ=5.00
0003 TIMER T=1.00
0004 CALL JOB:MM_CALIB ("1;1;2;6;0;99")
0005 END

MOVJ VJ=0.78
```

You have already modified the IP address and port of the IPC in the section "Test Standard Interface Communication". Therefore, you just only need to teach the calibration start point in this tutorial.

Teach the Calibration Start Point

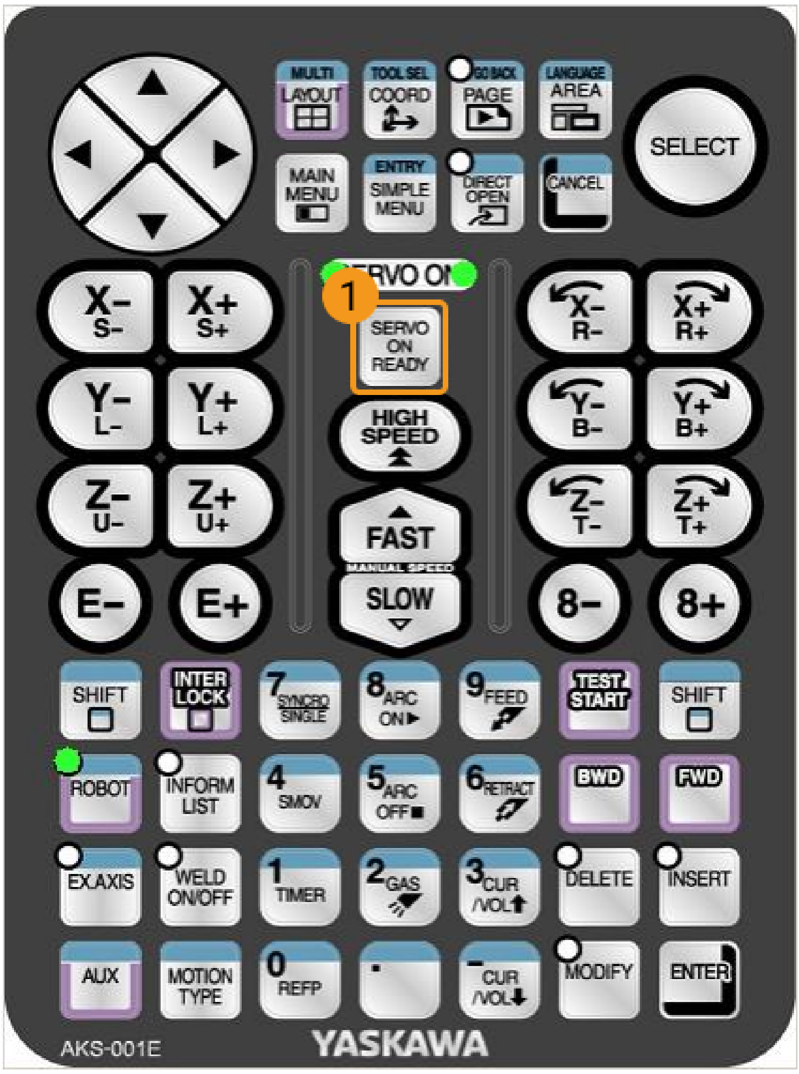
In the calibration process, the robot poses sent by the robot to Mech-Vision must be flange poses. Therefore, before this step, you need to switch the tool No. to that (usually 00) of the tool whose ToolOffset values are all 0. The Tool Center Point (TCP) of Tool00 is at the center of the flange.

1. Move the robot to the start point for calibration. Move the cursor to **MOVJ P099 VJ=5**, and press the `DIRECT OPEN` key on the teach pendant.

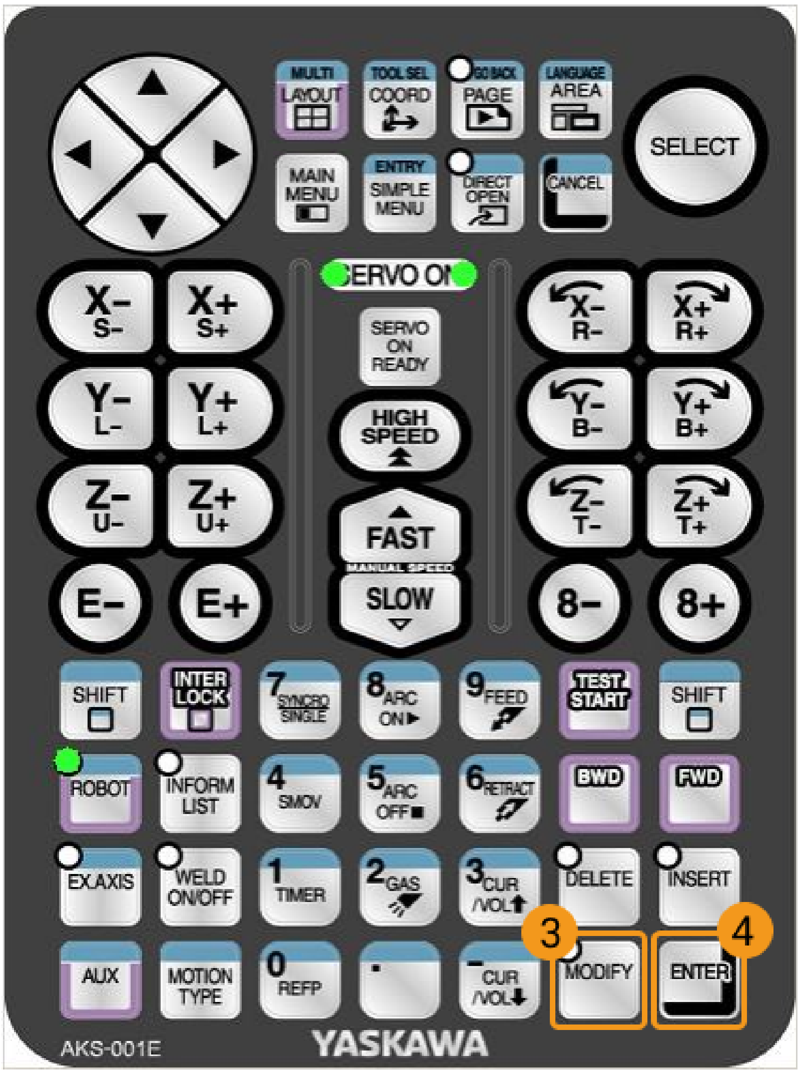


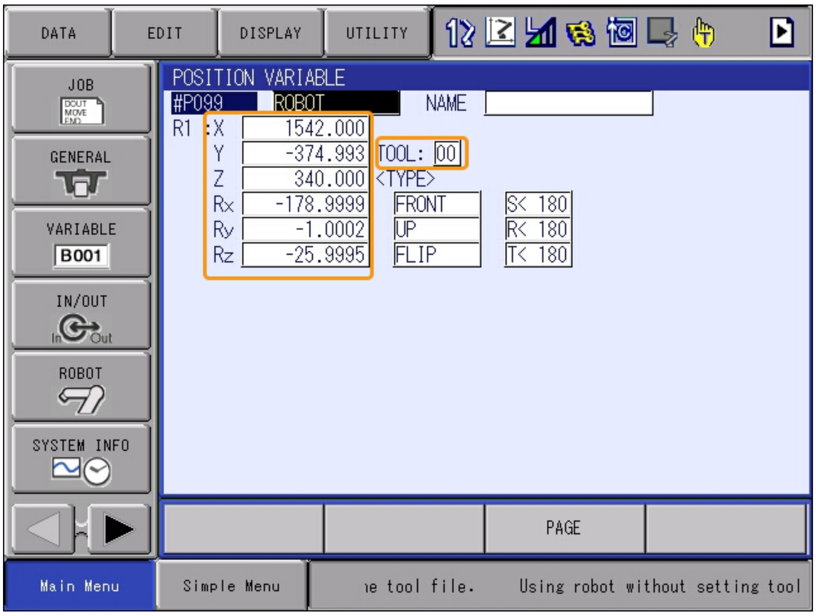
```
JOB CONTENT
J:MM_AUTO_CALIB                    S:0001
CONTROL GROUP: R1                  TOOL: 00
0000 NOP
0001 CALL JOB:MM_INIT_SOCKET ("192.168.1.12;50000;2")
0002 MOVJ P099 VJ=5.00
0003 TIMER T=1.00
0004 CALL JOB:MM_CALIB ("1;1;2;6;0;99")
0005 END

MOVJ P099 VJ=5.00
```

2. Change the value of **P099** to the current pose of the robot: Press the `SERVO ON READY` key on the teach pendant. Press the `MODIFY` key and then the `ENTER` key while holding the `enable switch` on the back. Make sure the value for TOOL is **00**.


1
SERVO ON READY
AKS-001E
YASKAWA

2





POSITION VARIABLE

#P099 ROBOT NAME

R1 :X	1542.000			
Y	-374.993	TOOL: 00		
Z	340.000	<TYPE>		
Rx	-178.9999		FRONT	S< 180
Ry	-1.0002		UP	R< 180
Rz	-25.9995		FLIP	T< 180

ıe tool file. Using robot without setting tool

3. Press the DIRECT OPEN key on the teach pendant again to return to JOB CONTENT.

So far, you have successfully modified the calibration program necessary for the automatic calibration process of YASKAWA robots.

Mount the Calibration Board

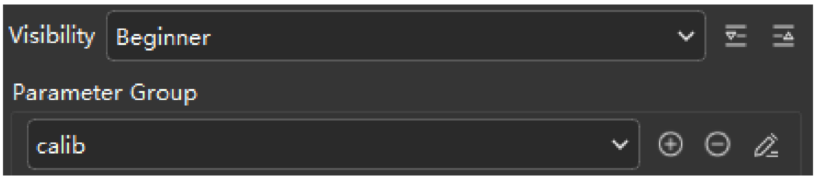
In the eye-to-hand setup, the calibration board needs to be mounted to the robot flange adapter.

Follow these steps:

1. Take the calibration board and flange adapter out of the system delivery package.
2. Use screws, gaskets, and nuts to fasten the flange adapter to the flange adapter.
3. Use screws, gaskets, and nuts to secure the calibration board to the flange adapter.
4. After mounting the calibration board, move the robot to the top of the lowest workpiece in the work area and into the camera's field of view (FOV).

Adjust Camera Parameters

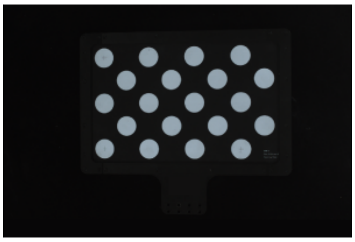
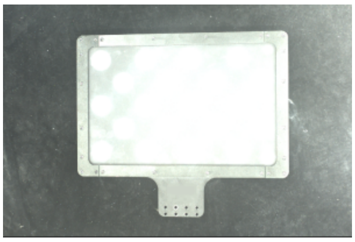
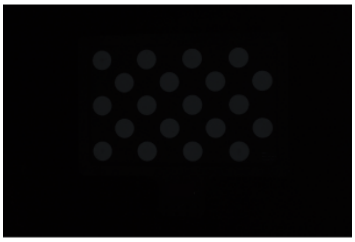
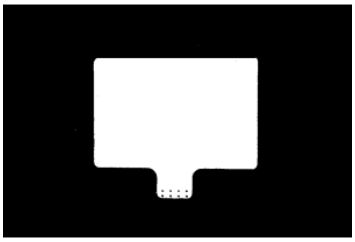
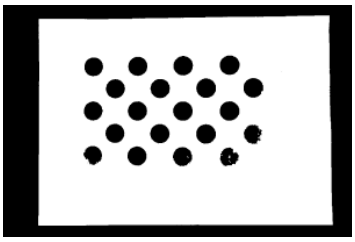
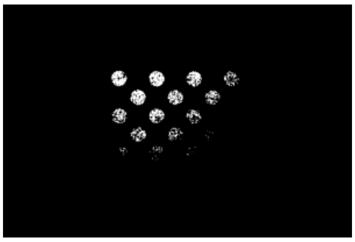
1. In Mech-Eye Viewer, connect to the camera, and then set **Parameter Group** to "calib".



Visibility: Beginner

Parameter Group: calib

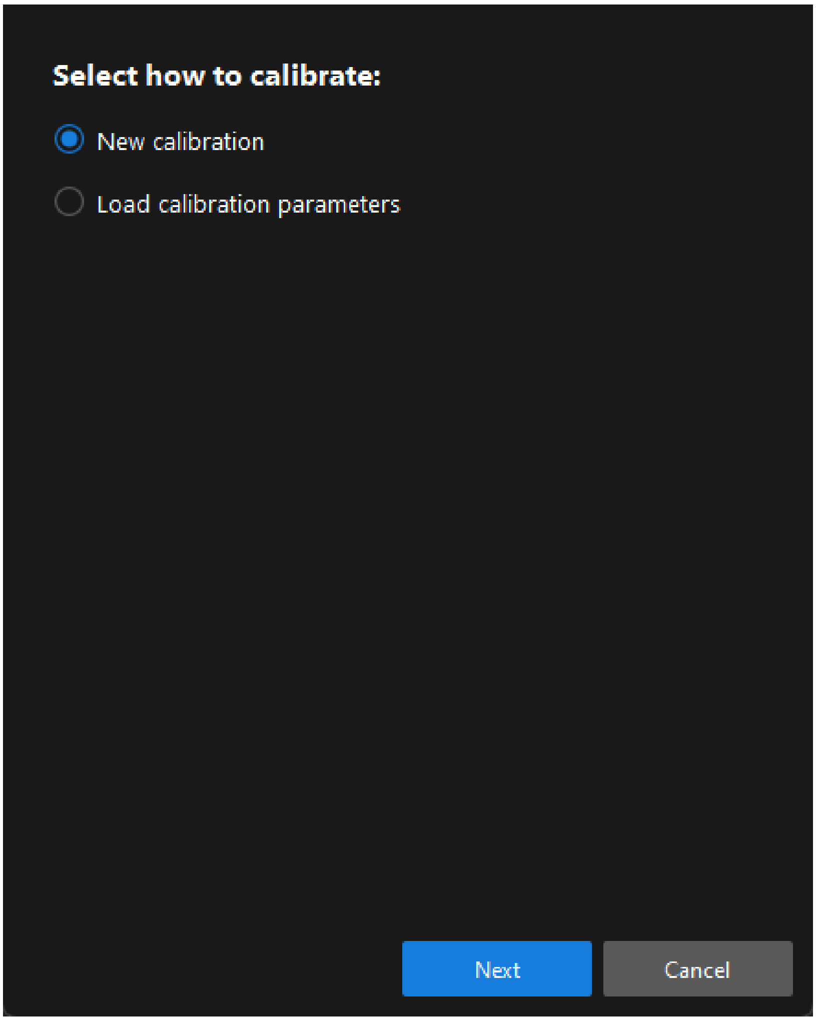
2. Adjust the 2D parameters to make sure that the 2D image of the calibration board is clear and neither overexposed nor underexposed.
3. Adjust 3D parameters to make sure that the obtained point clouds of the circles on the calibration board are complete and have clear contours. It is recommended to set the **Surface Smoothing** and **Outlier Removal** parameters of **Point Cloud Processing** to **Normal** to reduce point cloud fluctuation.

	Normal	Overexposed	Underexposed
2D image			
Point cloud			

Pre-calibration Configuration

1. In Mech-Vision, click the **[Camera Calibration (Standard)]** button on the toolbar. The **Configuration before Calibration** window will be prompted.

2. In the **Select how to calibrate** window, select the **New calibration** radio button, and then click the **[Next]** button.

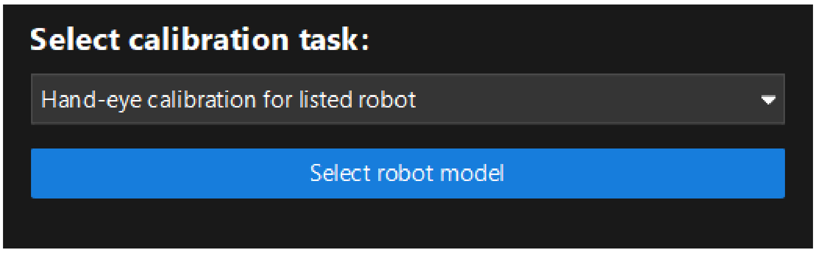


Select how to calibrate:

● New calibration

○ Load calibration parameters

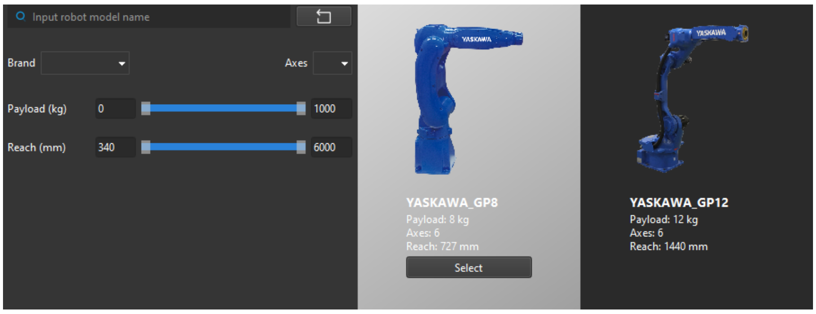
3. In the **Select calibration task** window, select **Hand-eye calibration for listed robot** from the drop-down list box, and click the **[Select robot model]** button.



Select calibration task:

Hand-eye calibration for listed robot

4. Click the **Brand** dropdown box, and select "YASKAWA". Then, at the right panel, select model "YASKAWA_GP8", click the **[Select]** button, and then click the **[Next]** button.



Input robot model name

Brand

Axes

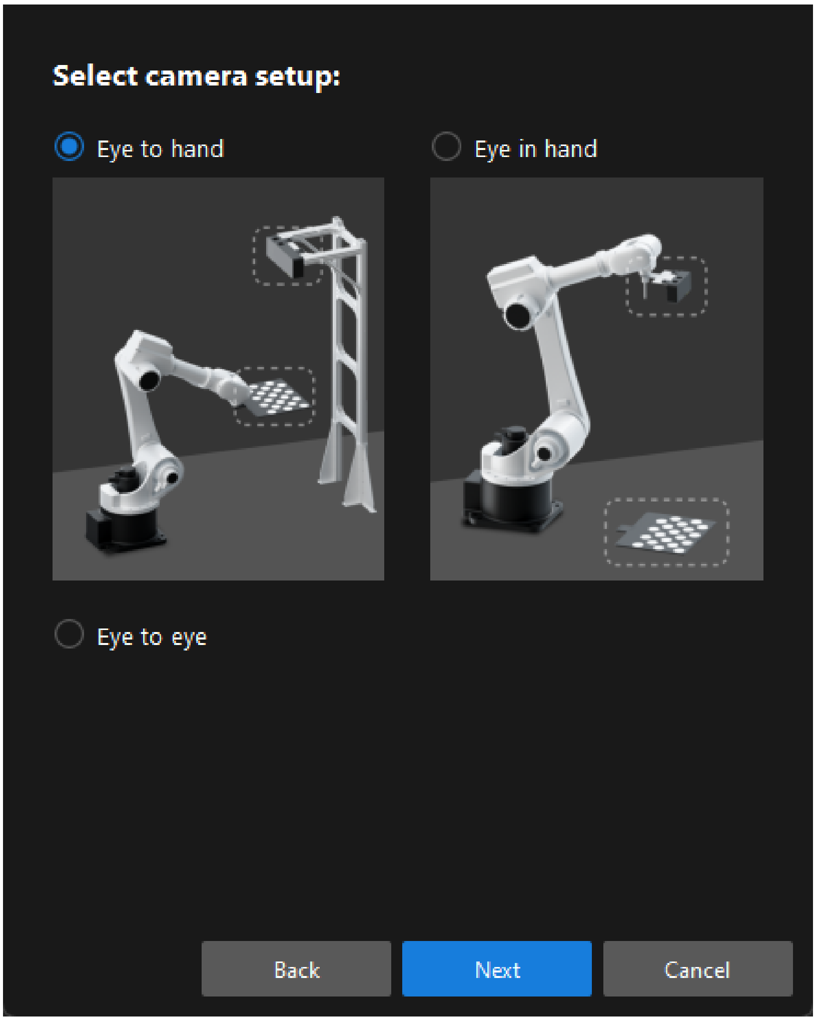
Payload (kg) 0 1000

Reach (mm) 340 6000

YASKAWA_GP8
Payload: 8 kg
Axes: 6
Reach: 727 mm

YASKAWA_GP12
Payload: 12 kg
Axes: 6
Reach: 1440 mm

5. In the **Select camera setup** window, select the **Eye to hand** radio button, and then click the **[Next]** button.



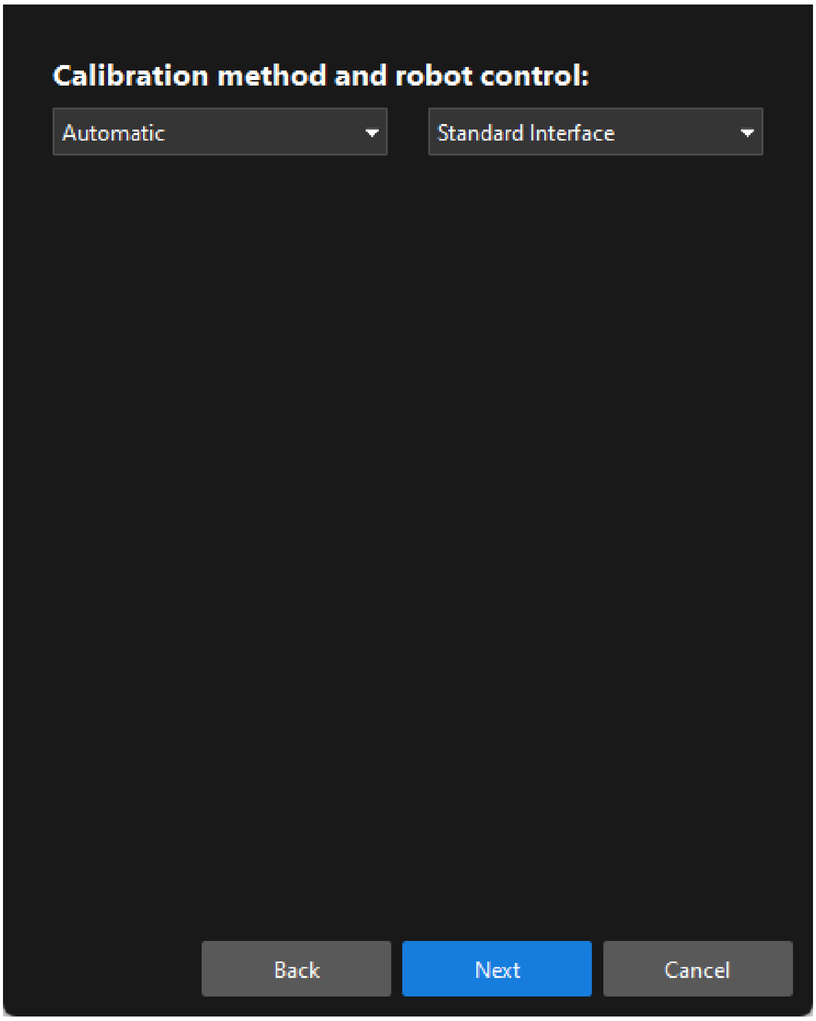
Select camera setup:

● Eye to hand

○ Eye in hand

○ Eye to eye

6. In the **Calibration method and robot control** window, select **Automatic** and **Standard Interface**, and then click the **[Next]** button.

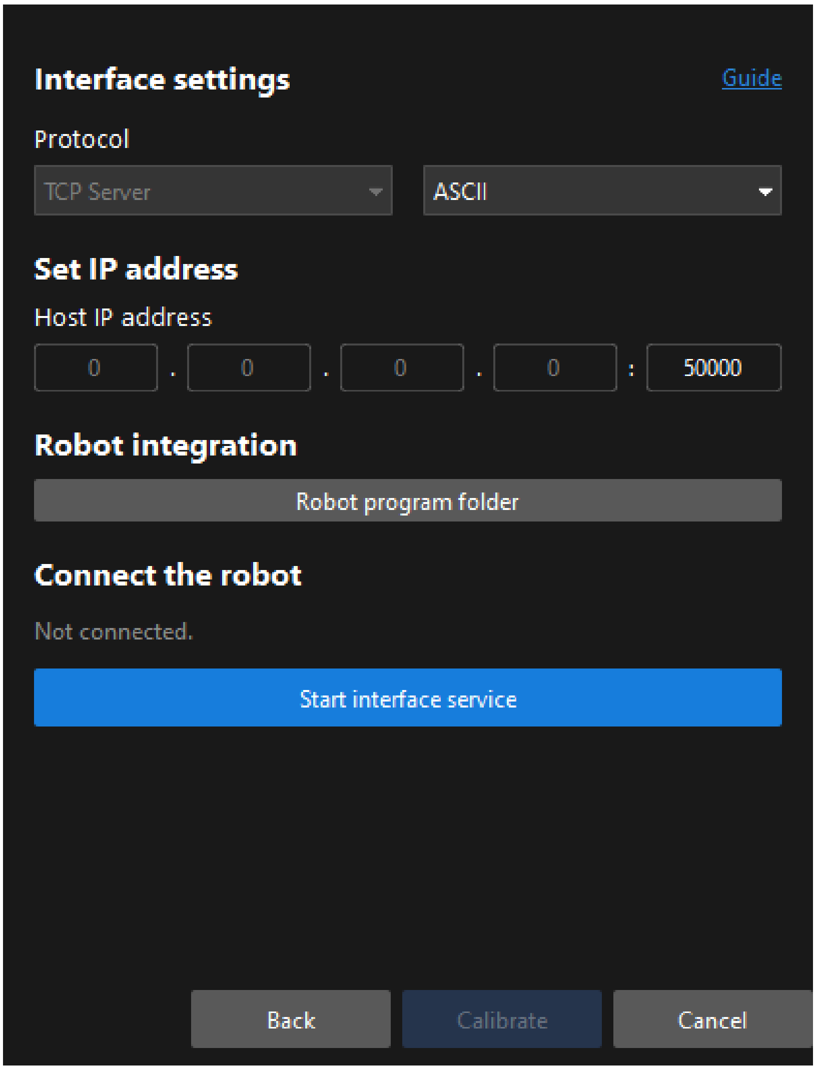


Calibration method and robot control:

Automatic

Standard Interface

7. In the **Interface settings** window, set the port of the **Host IP address** parameter to 50000.



Interface settings

Guide

Protocol

TCP Server

ASCII

Set IP address

Host IP address

0 . 0 . 0 . 0 : 50000

Robot integration

Connect the robot

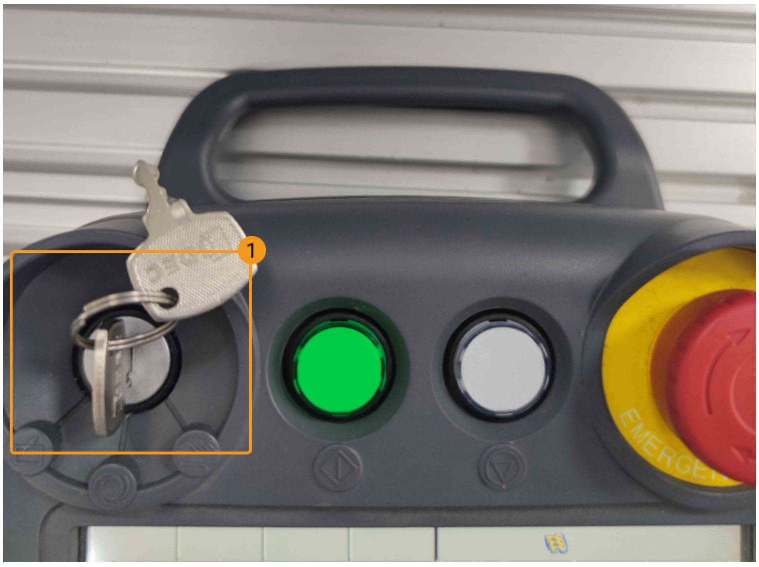
Not connected.

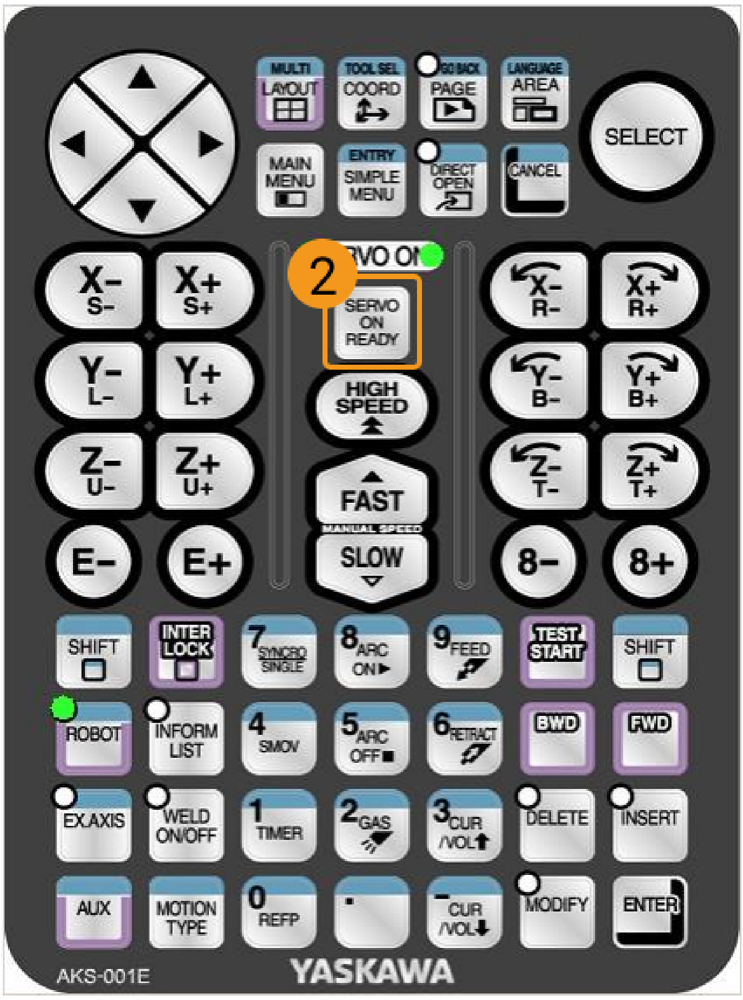
- In this tutorial, the vision system acts as the TCP server. Therefore, the **Host IP address** parameters should be set to the IP address and port of the IPC. Since the IP address of the IPC is a local IP address, you do not need to set it and the software uses “0.0.0.0” to indicate it.
- By default, the IPC uses port 50000 to provide the interface service. You can change the port number as required. If you have changed the port number, remember to use the correct port number when you try to establish communication with the vision system on the robot side.

8. Click the **[Connect the robot]** button in the **Connect the robot** area. The button will turn into **Waiting for the robot to connect...**
9. Run the calibration program on the robot teach pendant.

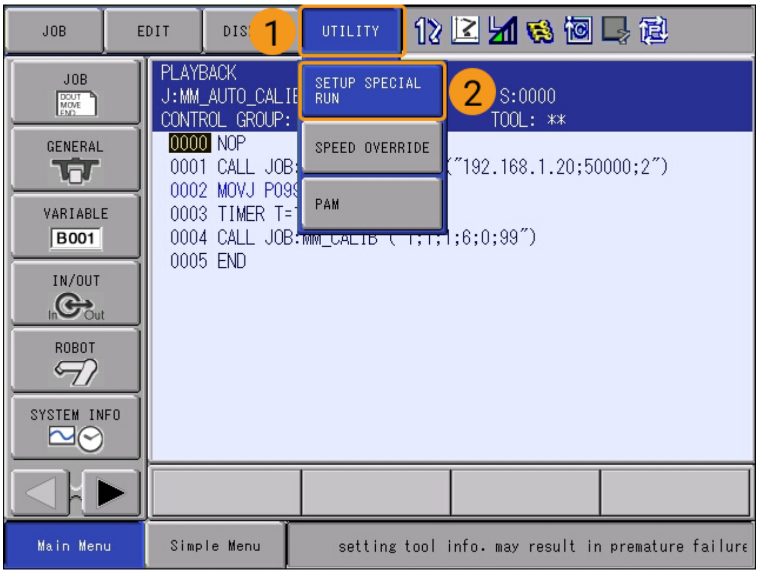
▼ *Operation instructions*

a. Move the cursor back to the first line (0000), turn the mode switch to `PLAY` mode to enter the **PLAY** mode, press the `SERVO ON READY` key on the teach pendant.



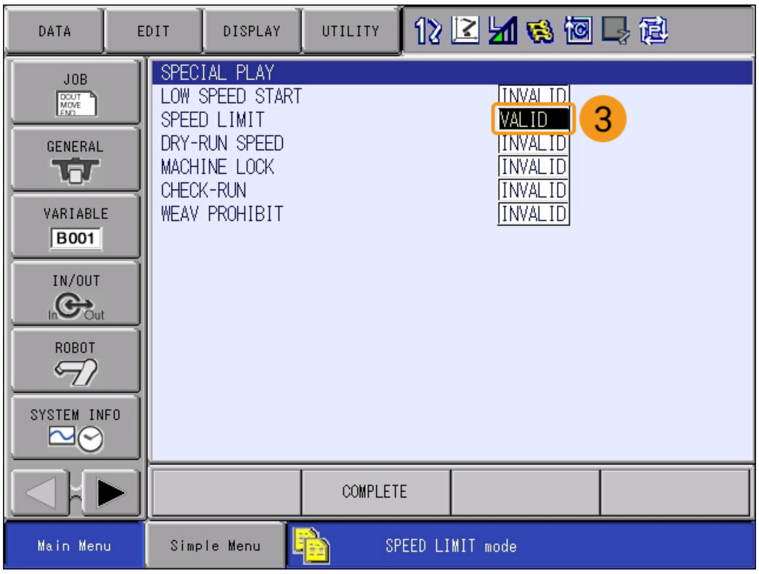


b. To move the robot at a low speed, select UTILITY › SETUP SPECIAL RUN on the teach pendant screen, and change **SPEED LIMIT** to **VALID**.



```
PLAYBACK
J:MM_AUTO_CALIE                      S:0000
CONTROL GROUP:                       TOOL: **
0000 NOP
0001 CALL JOB:                     ("192.168.1.20;50000;2")
0002 MOVJ P099
0003 TIMER T=
0004 CALL JOB:MM_CALIB ("1;1;1;6;0;99")
0005 END
```

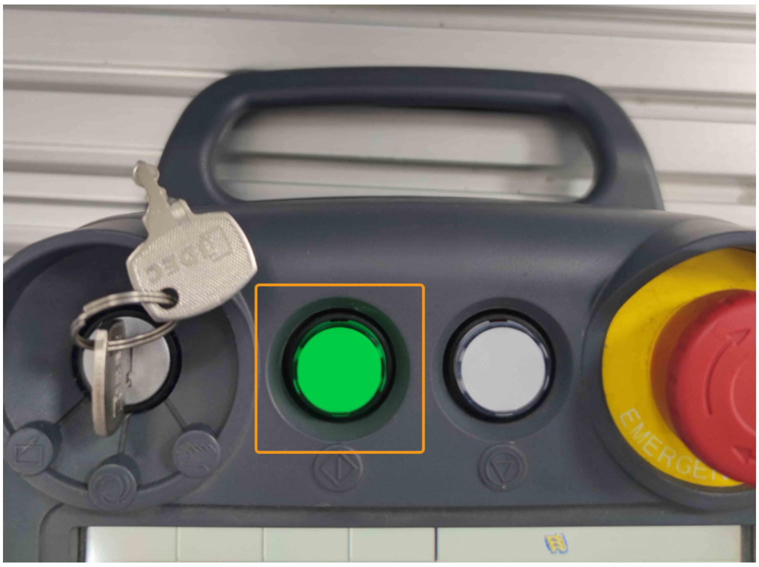
setting tool info. may result in premature failure



```
SPECIAL PLAY
LOW SPEED START          INVALID
SPEED LIMIT              VALID
DRY-RUN SPEED            INVALID
MACHINE LOCK             INVALID
CHECK-RUN                INVALID
WEAV PROHIBIT            INVALID
```

SPEED LIMIT mode

c. Press the green **[START]** button on the teach pendant. when the start LED is lit, the robot is running the program.



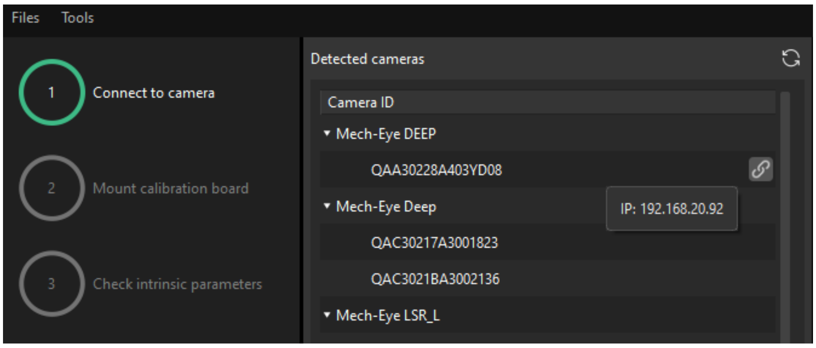
If the calibration program runs normally, “Connected” status message will be displayed in the **Connect the robot** area of Mech-Vision.

10. After the robot is connected, click the **[Calibrate]** button. The **Calibration (Eye to hand)** window will be prompted.

Calibration Procedure

Connect to a Camera

1. In the **Connect to Camera** step, find the camera to connect in the **Detected Cameras** list, and click the 🔗 button.

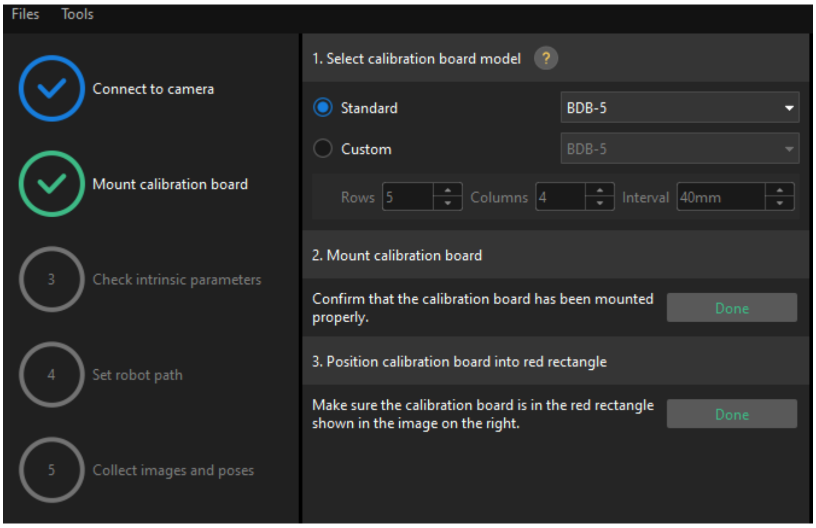


2. After the camera is connected, click the **[Capture once]** or **[Capture live]** button.
3. In the right **Image viewer** panel, ensure that the captured 2D image and depth map meet the calibration requirements and click the **[Next]** button on the bottom bar.

> If the captured image does not meet the calibration requirements, you need to open the Mech-Eye Viewer software to adjust the 2D and 3D exposure parameters and re-capture images.

Mount the Calibration Board

1. In the **Mount calibration board** step, select the **Standard** radio button and then select the calibration board model according to its model nameplate in the **1. Select calibration board type** area.
2. Make sure that the calibration board has been attached to the robot flange securely, and then click the **[Confirm]** button in the **2. Mount calibration board** area.
3. Confirm that the calibration board is in the center of the camera’s field of view (the red rectangle), and then click the **[Confirm]** button in the **3. Position calibration board into red rectangle** area.



1. Select calibration board model

● Standard BDB-5

○ Custom BDB-5

Rows 5 Columns 4 Interval 40mm

2. Mount calibration board

Confirm that the calibration board has been mounted properly.

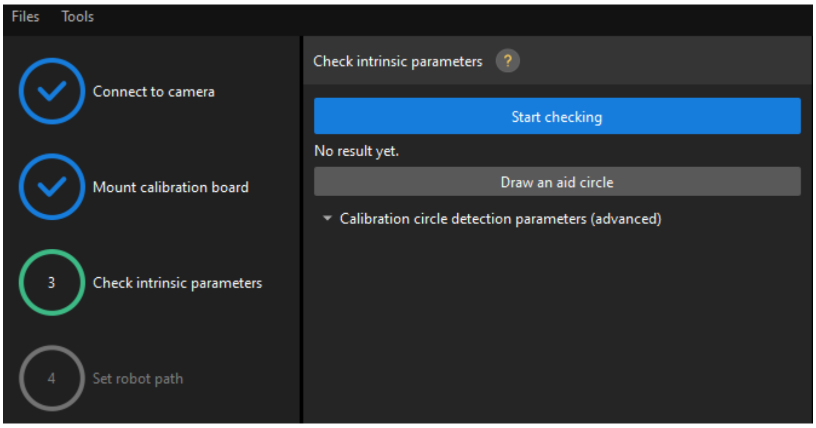
3. Position calibration board into red rectangle

Make sure the calibration board is in the red rectangle shown in the image on the right.

4. After all the operations related to the calibration board are completed, click the **[Next]** button on the bottom bar.

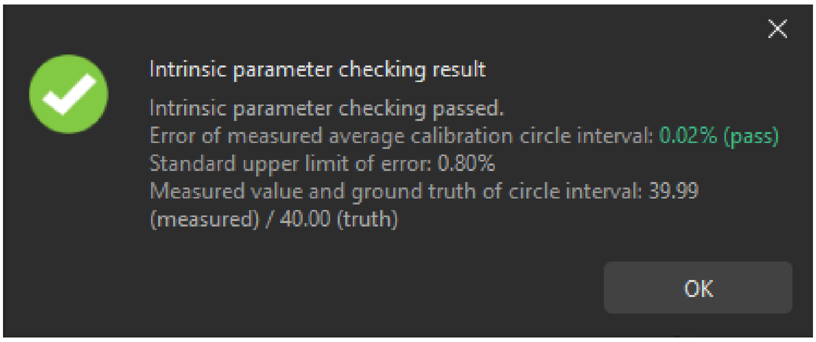
Check Intrinsic Parameters

1. In the **Check intrinsic parameters** step, click the **[Start checking]** button.


Files
Tools
Connect to camera
Mount calibration board
3 Check intrinsic parameters
4 Set robot path
Check intrinsic parameters
Start checking
Draw an aid circle
Calibration circle detection parameters (advanced)

No result yet.

2. Confirm that the camera intrinsic parameters check passes, and then click the **[Next]** button on the bottom bar.



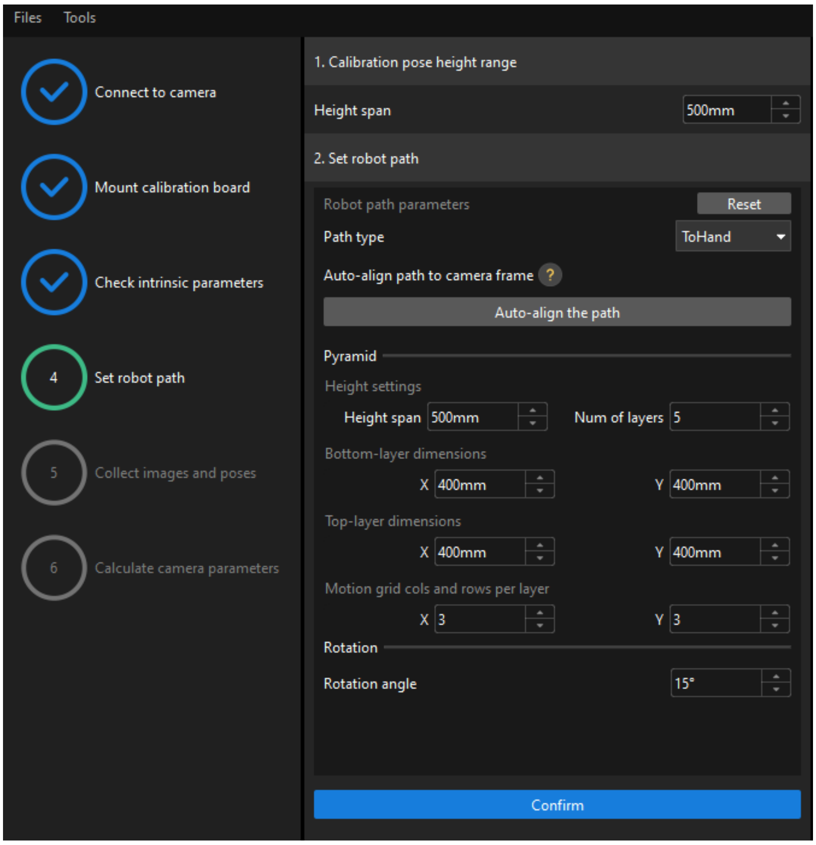
Intrinsic parameter checking result

Intrinsic parameter checking passed.
Error of measured average calibration circle interval: 0.02% (pass)
Standard upper limit of error: 0.80%
Measured value and ground truth of circle interval: 39.99 (measured) / 40.00 (truth)

If the camera intrinsic parameter check fails, adjust detection parameters by drawing aid circles or manually adjusting the calibration circle detection parameters.

Set Robot Path

1. In the **Set robot path** step, specify the **Height span** parameter. Please set this parameter according to the working distance range of the calibration board in the camera height direction.



1. Calibration pose height range

Height span 500mm

2. Set robot path

Robot path parameters

Path type ToHand

Auto-align path to camera frame

Pyramid

Height settings

Height span 500mm Num of layers 5

Bottom-layer dimensions

X 400mm Y 400mm

Top-layer dimensions

X 400mm Y 400mm

Motion grid cols and rows per layer

X 3 Y 3

Rotation

Rotation angle 15°

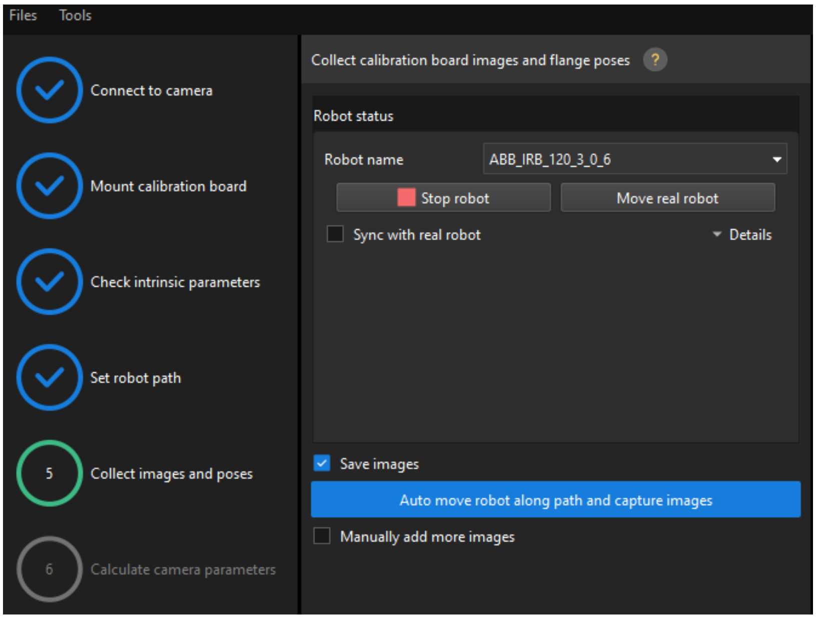
2. Set the **Path type** to **ToHand**, specify the pyramid parameters **Height span**, **Num of layers**, **Bottom-layer dimensions X/Y**, **Top-layer dimensions X/Y**, and **Motion grid cols and rows per layer**, specify the satellite parameter **Rotation angle**, and then click the **[Confirm]** button.

   The set robot path should cover the work area.

3. In the right **Scene Viewer** panel, confirm that the waypoints of the automatically generated motion path will not collide with obstacles in the environment, and then click the **[Next]** button.

Collect Images and Poses

1. In the **Collect images and poses** step, select the **Save images** checkbox.



Robot name: ABB_IRB_120_3_0_6

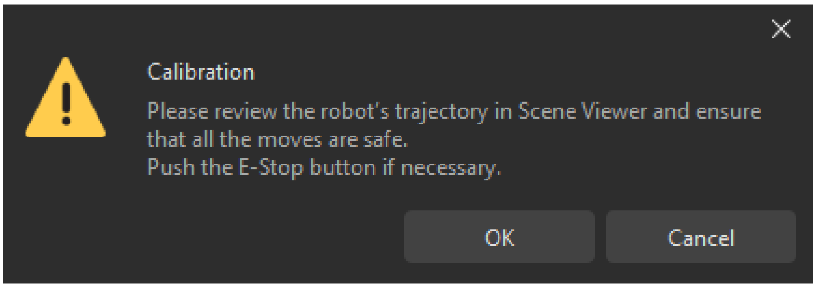
□ Sync with real robot

☑ Save images

Auto move robot along path and capture images

□ Manually add more images

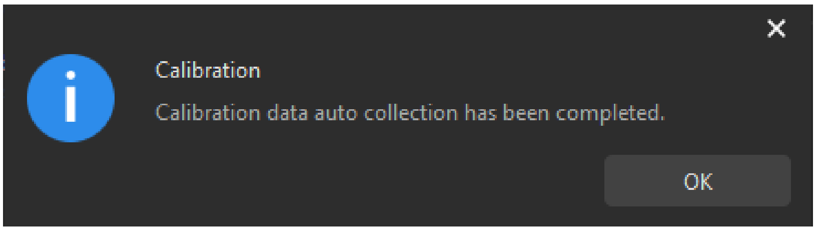
2. Click the **[Control robot to auto move along path and capture images]** button.
3. Read the safety window carefully and click the **[OK]** button.



Calibration

Please review the robot's trajectory in Scene Viewer and ensure that all the moves are safe.
Push the E-Stop button if necessary.

4. Wait until the robot finishes moving along the preset path and the camera finishes capturing images on all waypoints. Captured images can be seen in the right **Marker Image and Pose List** panel during this process.
5. After the automatic image capturing finishes, click the **[OK]** button, and then click the **[Next]** button on the bottom bar.

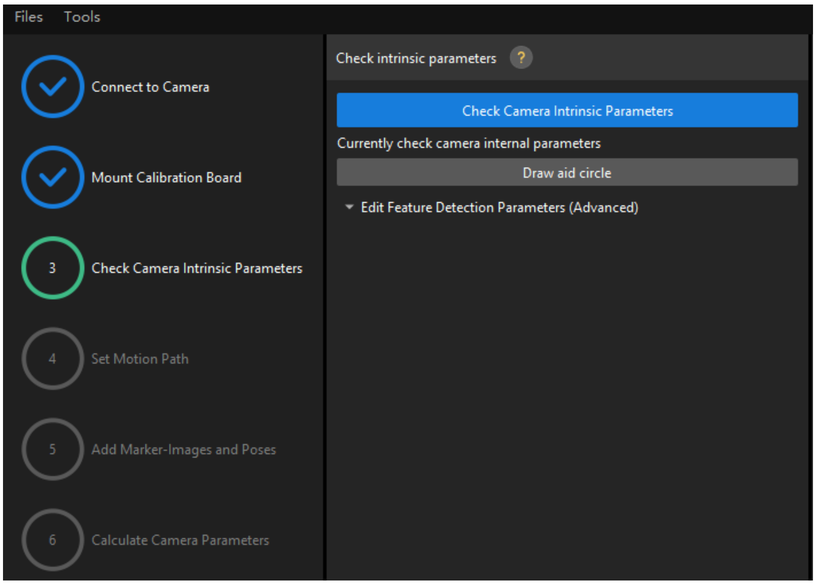


Calibration

Calibration data auto collection has been completed.

Calculate Camera Parameters

1. In the **Calculate camera parameters** window, click the **[Calculate camera extrinsic parameters]** button.


Files
Tools
Connect to Camera
Mount Calibration Board
3
Check Camera Intrinsic Parameters
4
Set Motion Path
5
Add Marker-Images and Poses
6
Calculate Camera Parameters
Check intrinsic parameters
?
Check Camera Intrinsic Parameters
Draw aid circle
Edit Feature Detection Parameters (Advanced)

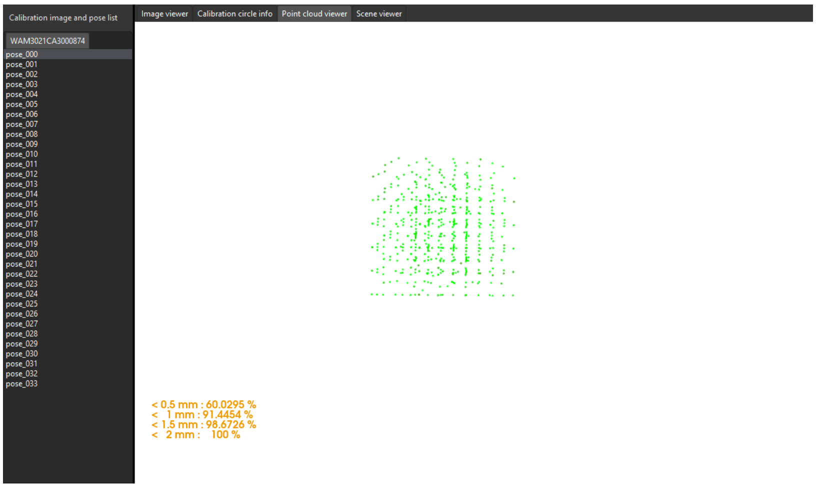
Currently check camera internal parameters

2. In the prompted window indicating calibration success, click the [OK] button.
3. View the calibration error of the point cloud in the right **Point cloud viewer** panel.

> The error point cloud shows the disparity between the calculated value and the actual value of the circles on the calibration board for each pose.

4. Confirm that the calibration accuracy meets the project requirements.

   Determine the error value with 100% to obtain the rough calibration accuracy. For example, the calibration accuracy in the following figure is within 2.5 mm.



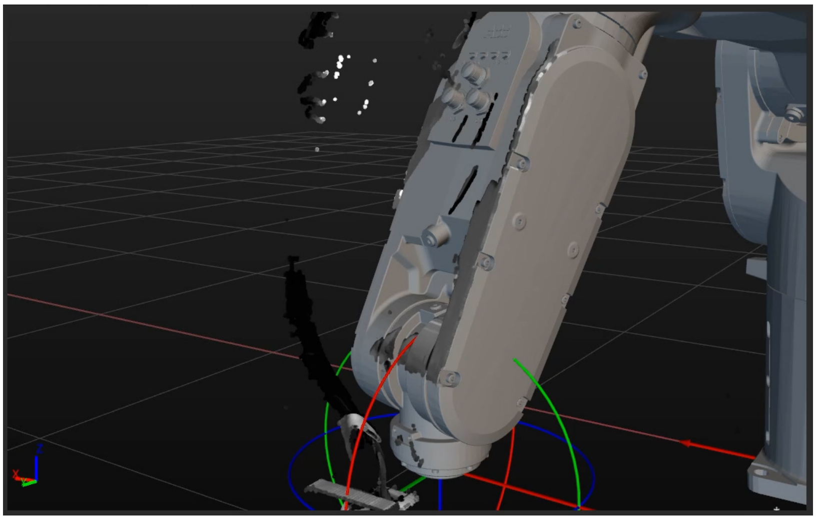
```
< 0.5 mm : 60.0295 %
<   1 mm : 91.4454 %
< 1.5 mm : 98.6726 %
<   2 mm :   100 %
```

   To improve the calibration accuracy, please refer to the section Calibration Result Check and Analysis.

Validate and Save the Calibration Result

1. After calibration, move the robot arms into the FOV of the camera.
2. In the **Calculate camera parameters** step, click the [**Re-calculate camera extrinsic parameters**] button. This operation triggers the camera to capture images.
3. Click **Scene viewer** to check the coincidence degree between the point cloud of the robot and the robot model.

If the point cloud of the robot approximately coincides with the robot model, the calibration is successful.



4. Click the [Save] button on the bottom bar. In the prompted **Save Calibration Files** dialog box, click the [OK] button. The camera calibration result will be automatically saved in the “calibration” directory of the project.

3.4. Workpiece Locating

Before using this tutorial, you should have created a Mech-Vision solution using the “General Workpiece Recognition” case project in the “Create a Mech-Vision Project and Save It” section.

In this tutorial, you will first learn the project workflow, and then deploy the project by adjusting the Step parameters to recognize the workpieces’ poses and output the vision result.

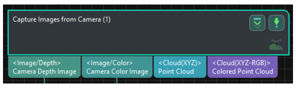
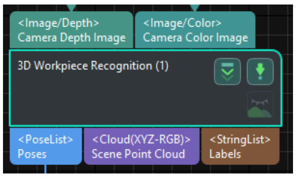
▶ https://www.youtube.com/watch?v=CFNR4zeyR0k/PLVcMd7cW2rXVtrAejMyVQni2dUDv8bxje *(YouTube video)*

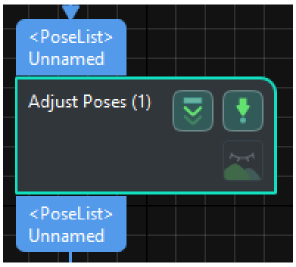
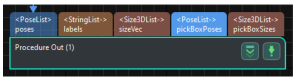
Video Tutorial: Workpiece Locating

ℹ In this project, every time the Mech-Vision is run, a vision point will be output.

Introduction to the Project Workflow

The following table describes each Step in the project workflow.

No.	Phase	Step	Image	Description
1	Capture images	Capture Images from Camera	 Capture Images from Camera (1) <Image/Depth> Camera Depth Image <Image/Color> Camera Color Image <Cloud(XYZ)> Point Cloud <Cloud(XYZ-RGB)> Colored Point Cloud	Connect to the camera and capture images
2	Recognize workpieces	3D Workpiece Recognition	 <Image/Depth> Camera Depth Image <Image/Color> Camera Color Image 3D Workpiece Recognition (1) <PoseList> Poses <Cloud(XYZ-RGB)> Scene Point Cloud <StringList> Labels	Use 3D matching algorithms to calculate the workpieces’ poses (as pick points)

No.	Phase	Step	Image	Description
3	Adjust poses	Adjust poses	 <PoseList> Unnamed Adjust Poses (1) <PoseList> Unnamed	Transform the pick points from the camera reference frame to the robot reference frame
4	Output the vision result	Procedure Out	 <PoseList> poses <StringList> labels <Size3DList> sizeVec <PoseList> pickBoxPoses <Size3DList> pickBoxSizes Procedure Out (1)	Output the workpieces' poses for the robot to pick

ℹ A pick point refers to a point on the workpiece on which the robot can pick the object.

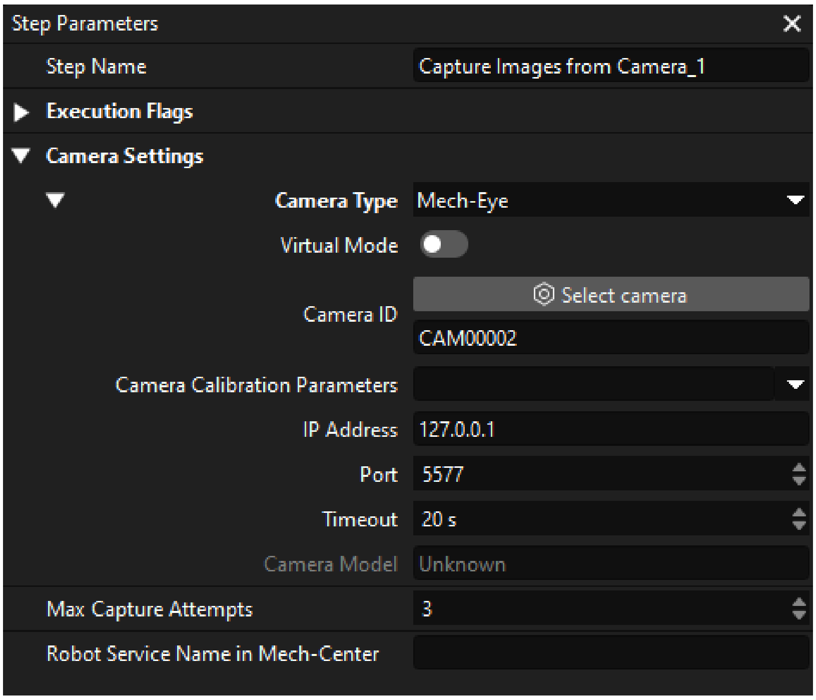
Adjust Step Parameters

In this section, you will deploy the project by adjusting the parameters of each Step.

Capture Images from Camera

You should adjust the parameters of the "Capture Images from Camera" Step to connect to the camera.

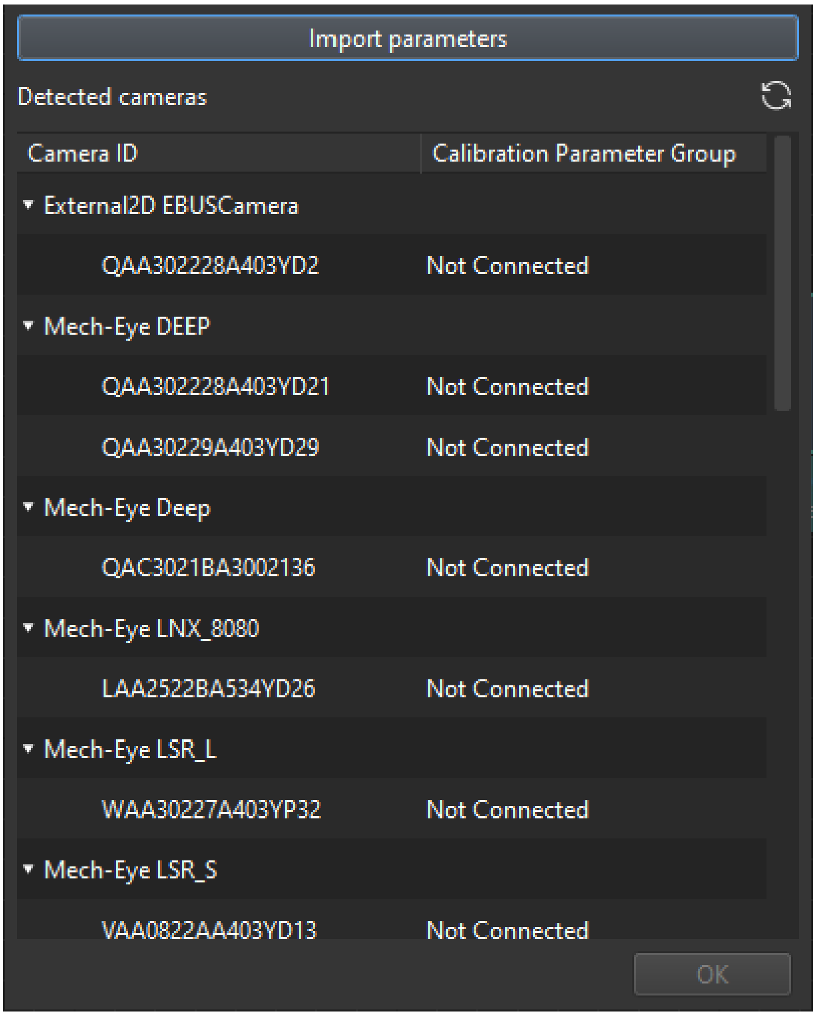
1. Select the "Capture Images from Camera" Step, and click **[Select camera]** on the Step parameters tab.



Step Parameters

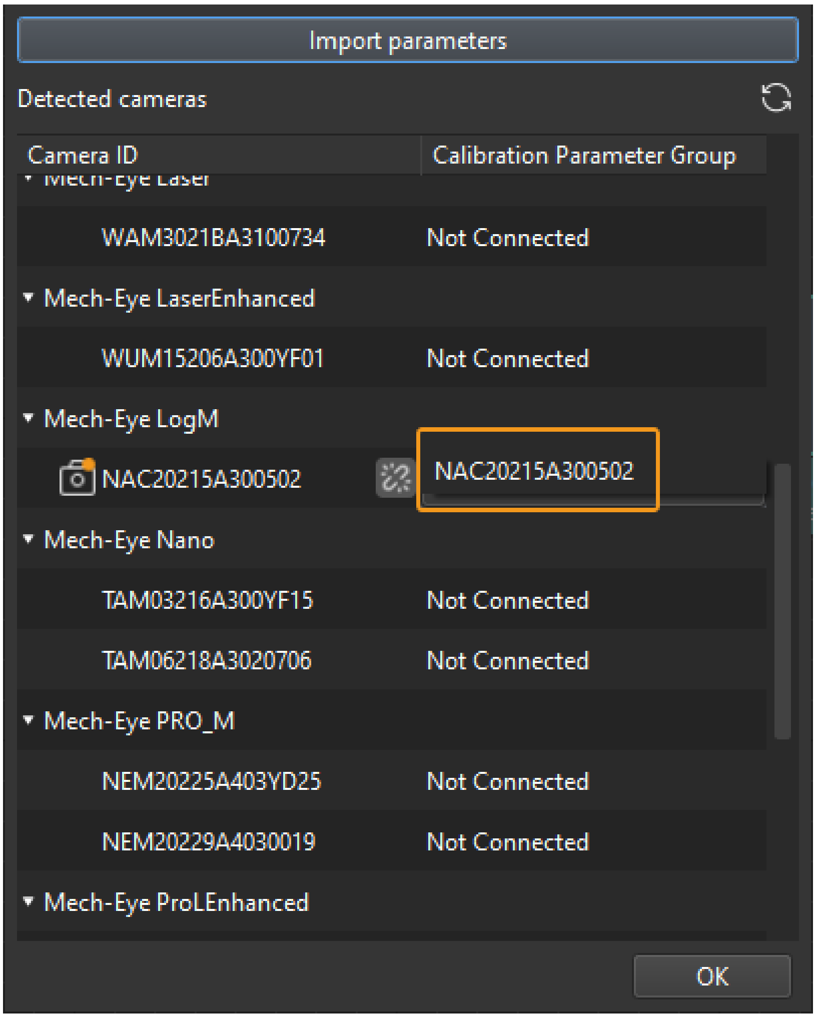
Step Name	Capture Images from Camera_1
▶ **Execution Flags**	
▼ **Camera Settings**	
▼ **Camera Type**	Mech-Eye
Virtual Mode	○
Camera ID	Select camera CAM00002
Camera Calibration Parameters	
IP Address	127.0.0.1
Port	5577
Timeout	20 s
Camera Model	Unknown
Max Capture Attempts	3
Robot Service Name in Mech-Center	

2. In the pop-up window, click 🔗 on the right of the camera serial No. to connect the camera. After the camera is connected, the icon 🔗 will turn into ⛓.



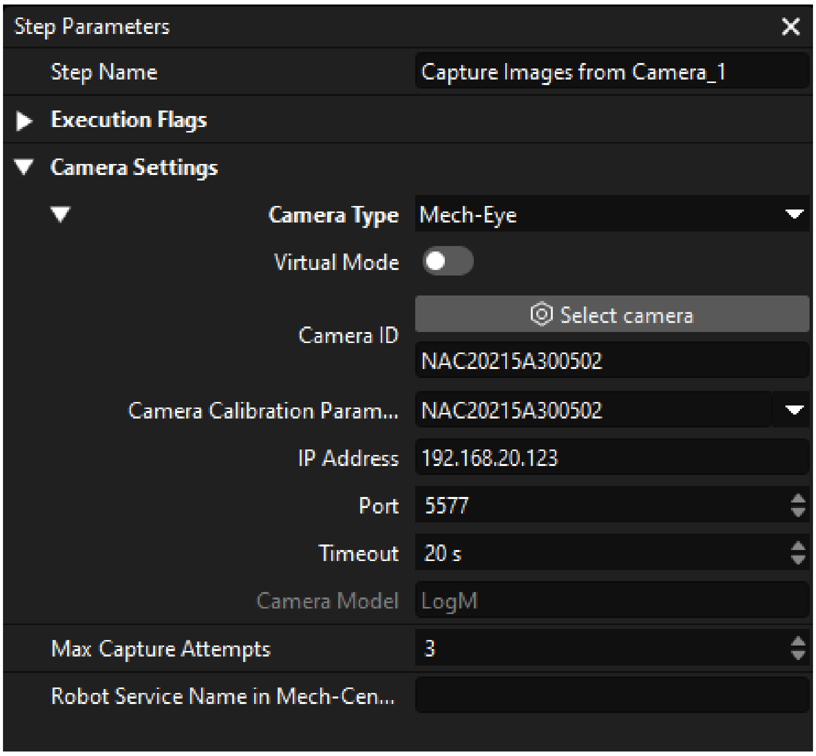
Camera ID	Calibration Parameter Group
▾ External2D EBUSCamera	
QAA302228A403YD2	Not Connected
▾ Mech-Eye DEEP	
QAA302228A403YD21	Not Connected
QAA30229A403YD29	Not Connected
▾ Mech-Eye Deep	
QAC3021BA3002136	Not Connected
▾ Mech-Eye LNX_8080	
LAA2522BA534YD26	Not Connected
▾ Mech-Eye LSR_L	
WAA30227A403YP32	Not Connected
▾ Mech-Eye LSR_S	
VAA0822AA403YD13	Not Connected

After the camera is connected, select the parameter group. Click **[Select from]** and select the displayed parameter group.



Camera ID	Calibration Parameter Group
▾ Mech-Eye Laser	
WAM3021BA3100734	Not Connected
▾ Mech-Eye LaserEnhanced	
WUM15206A300YF01	Not Connected
▾ Mech-Eye LogM	
NAC20215A300502	NAC20215A300502
▾ Mech-Eye Nano	
TAM03216A300YF15	Not Connected
TAM06218A3020706	Not Connected
▾ Mech-Eye PRO_M	
NEM20225A403YD25	Not Connected
NEM20229A4030019	Not Connected
▾ Mech-Eye ProLEnhanced	

3. After the camera is connected and the parameter group is selected, the calibration parameter group, IP address, and ports of the camera will be obtained automatically. Just keep the default settings of the other parameters.



Step Parameters

Step Name: Capture Images from Camera_1

▶ Execution Flags

▼ Camera Settings

▼ Camera Type: Mech-Eye

Virtual Mode: ○

Camera ID: Select camera; NAC20215A300502

Camera Calibration Param...: NAC20215A300502

IP Address: 192.168.20.123

Port: 5577

Timeout: 20 s

Camera Model: LogM

Max Capture Attempts: 3

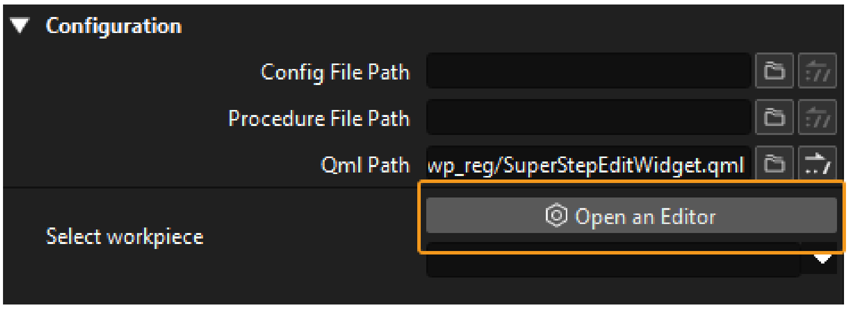
Robot Service Name in Mech-Cen...:

Now, you have connected the software to the camera.

3D Workpiece Recognition

The “3D Workpiece Recognition” Step has integrated a 3D Workpiece Recognition visualized configurator, which provides point cloud preprocessing, model-based matching, and pose (pick point) calculation.

Select the “3D Workpiece Recognition” Step, and click [Open an Editor] on the Step parameters tab.



▼ Configuration

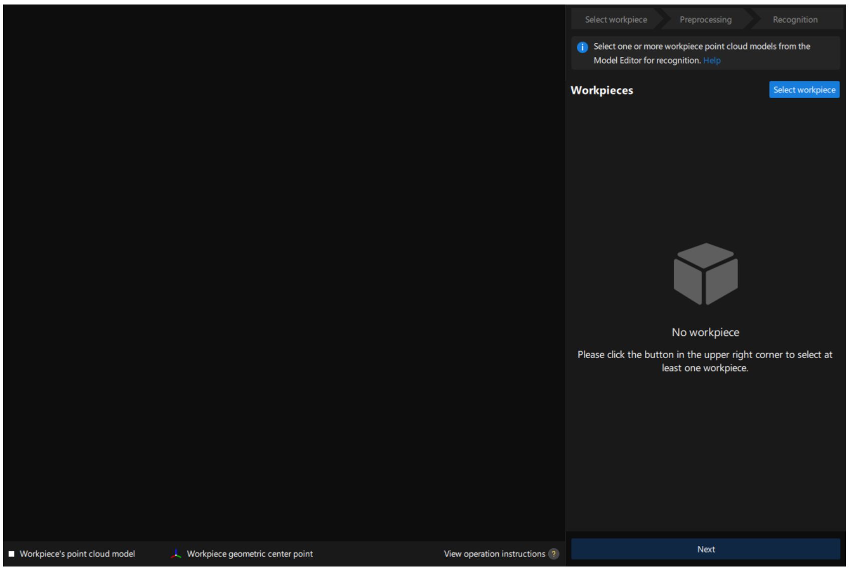
Config File Path:

Procedure File Path:

Qml Path: wp_reg/SuperStepEditWidget.qml

Select workpiece: Open an Editor

The 3D Workpiece Recognition visualized configurator is shown below.

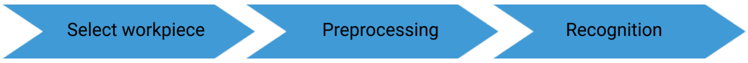

Select workpiece
Preprocessing
Recognition
Workpieces
Select workpiece
Workpiece's point cloud model
Workpiece geometric center point
View operation instructions ?
Next

Select one or more workpiece point cloud models from the Model Editor for recognition. Help

No workpiece

Please click the button in the upper right corner to select at least one workpiece.

Then follow the procedure to recognize workpieces.



Select workpiece

Preprocessing

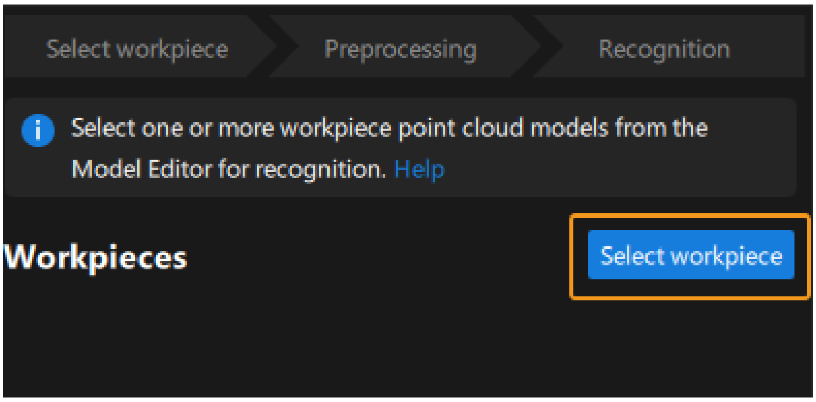
Recognition

Select Workpiece

After entering the 3D Workpiece Recognition visualized configurator, you need to make the point cloud model for the workpieces to recognize.

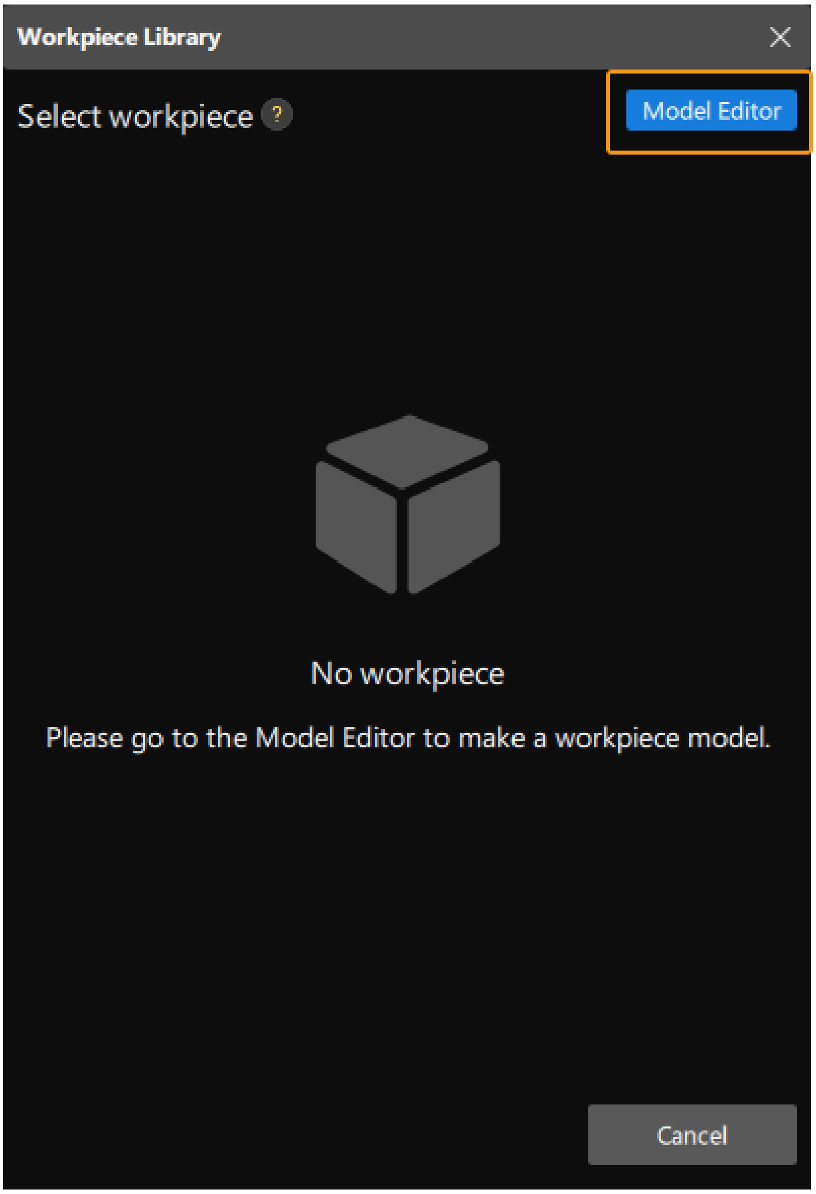
1. Open the Model Editor.

   At the upper right corner of the 3D Workpiece Recognition visualized configurator, click **[Select workpiece]**.



Select one or more workpiece point cloud models from the Model Editor for recognition. Help

   In the pop-up Workpiece Library window, click **[Model Editor]**.



Workpiece Library

Select workpiece ?

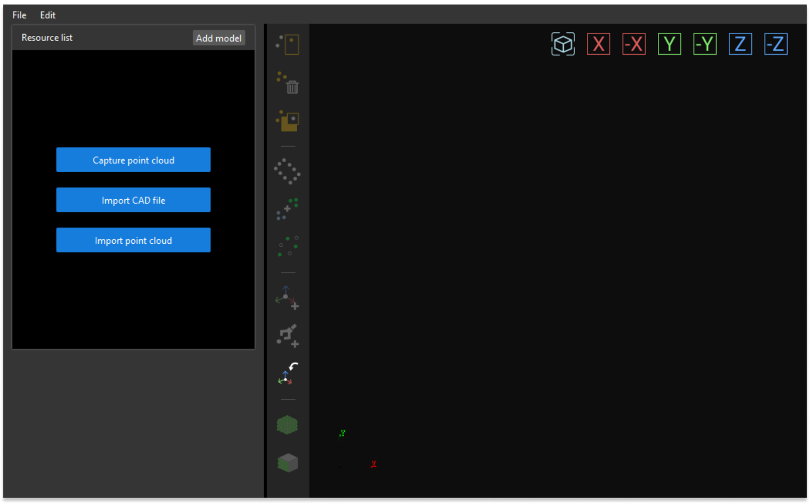
Model Editor

No workpiece

Please go to the Model Editor to make a workpiece model.

Cancel

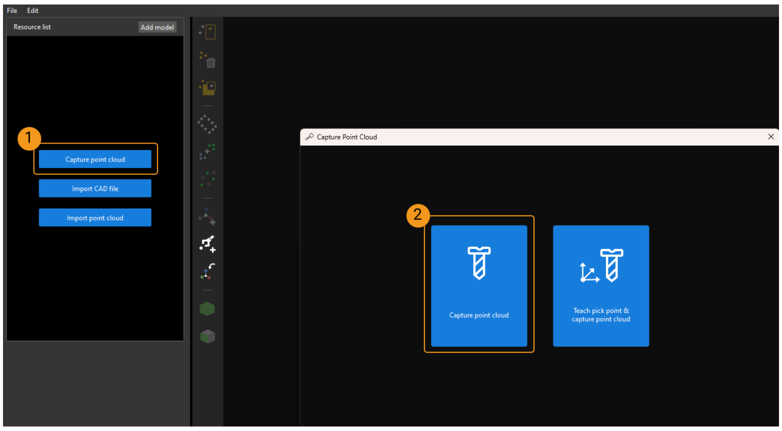
The following figure shows the Model Editor.



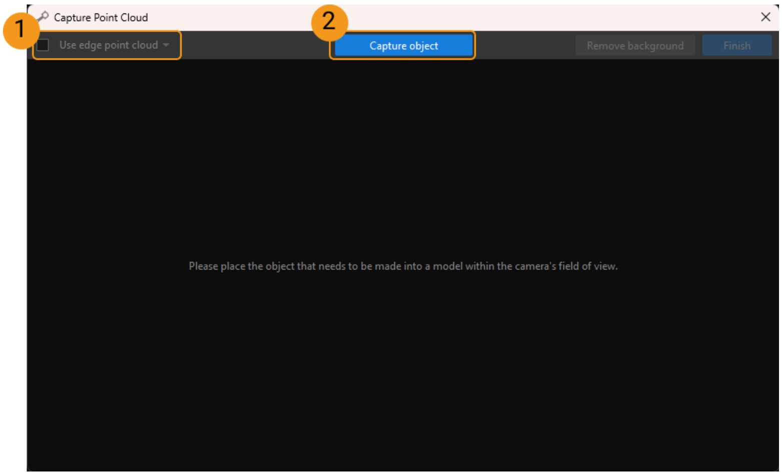
2. Generate the point cloud model by capturing point clouds.

a. Capture the depth map of the object.

Click the **[Capture point cloud]** button in the start screen, and select **[Capture point cloud]** in the pop-up window.

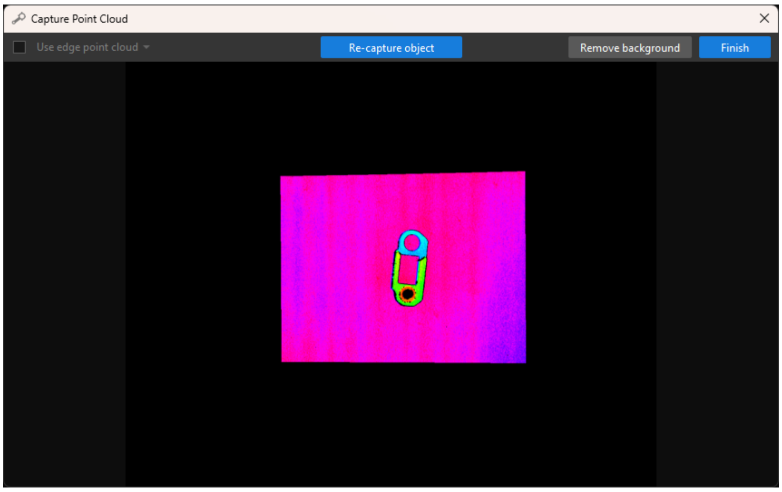


Due to the complex surface features of the track link, it is recommended to create a surface point cloud model for this workpiece. Therefore, clear the **Use edge point cloud** checkbox. Then click the **[Capture object]** button to capture the depth map of the target object.



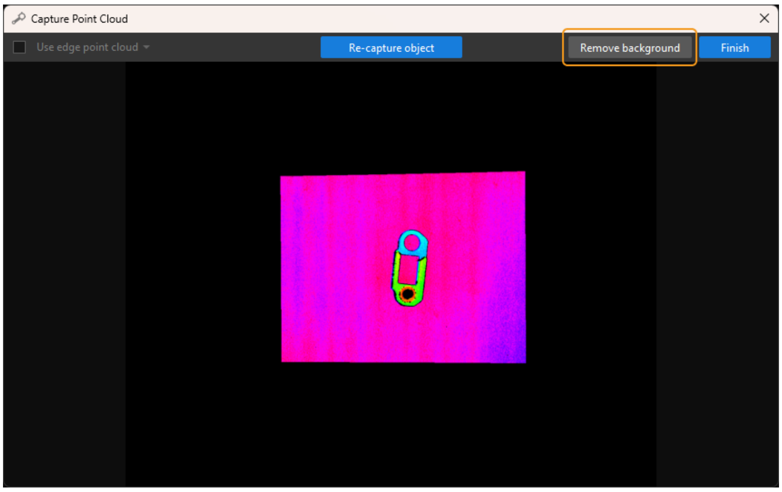
Please place the object that needs to be made into a model within the camera's field of view.

The captured depth map of the target object and background is shown as below.

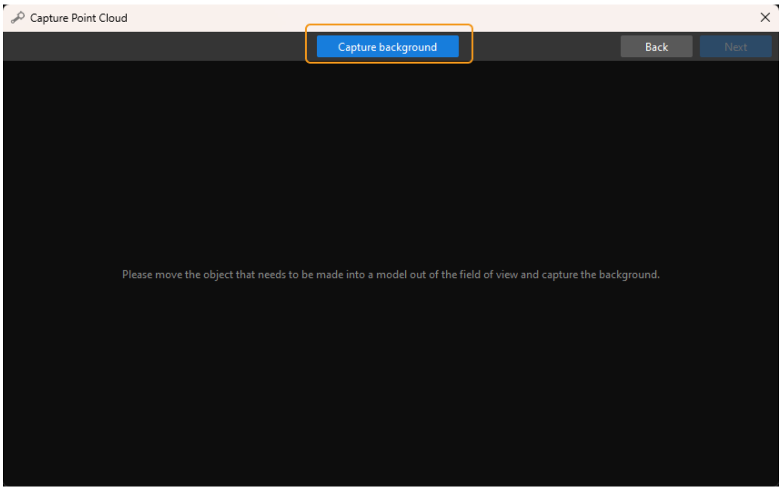


b. Capture background.

In the upper-right corner, click the [**Remove background**] button.

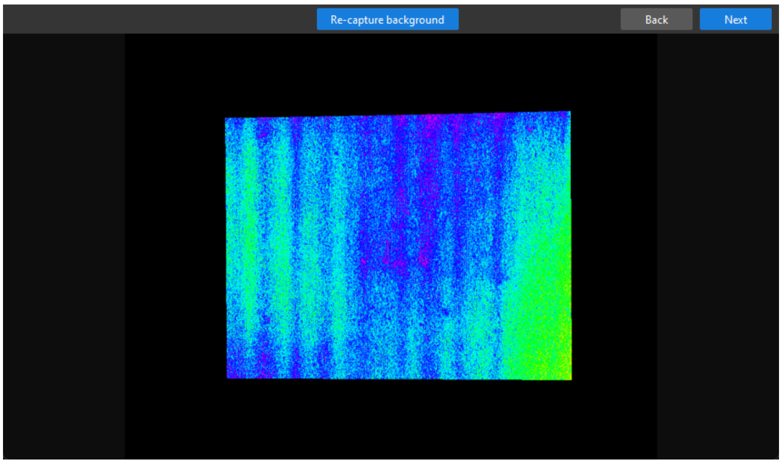


Remove the target object in the camera's field of view first, and click [**Capture background**] again to capture a depth map of the background.



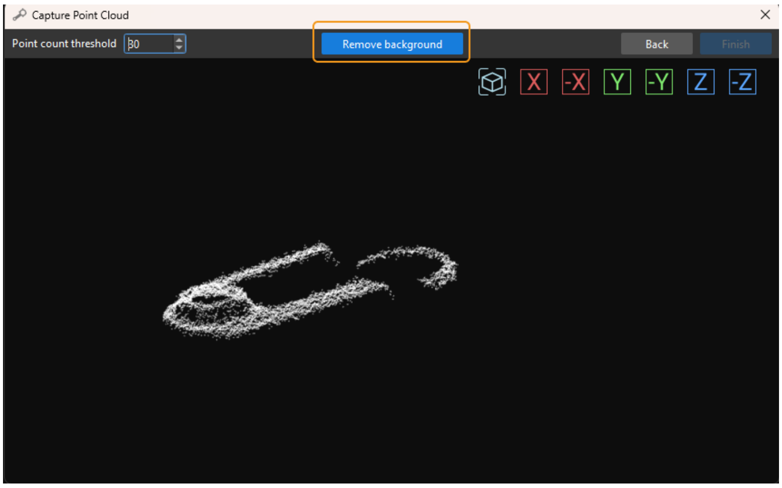
Please move the object that needs to be made into a model out of the field of view and capture the background.

The depth map of the background is shown as below. Then, click the [**Next**] button in the upper-right corner.

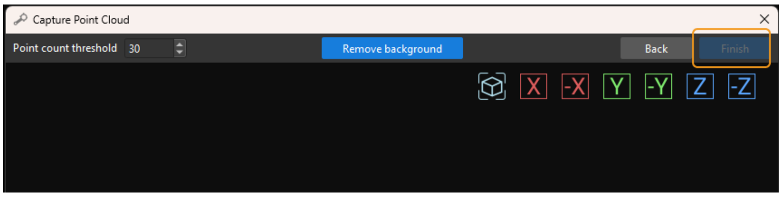


c. Subtract the background.

Click **[Remove background]** to generate a point cloud model of the target object, as shown below.



Click **[Finish]** to import the point cloud model of the target object to Matching Model and Pick Point Editor.



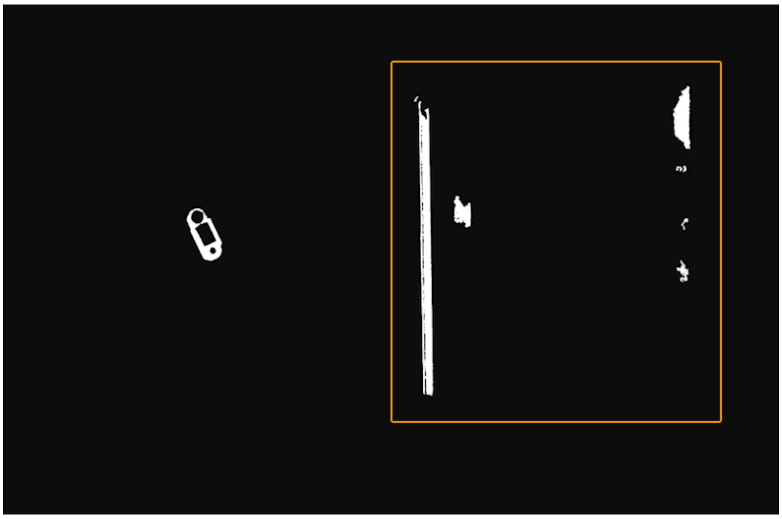
3. Edit the point cloud model.

A generated point cloud model may not meet the actual requirement. In such case, you need to edit the model, including removing outliers and downsampling the point cloud.

a. Remove unwanted points.

Click the button, select unwanted points to remove, and then click the button to remove selected points.

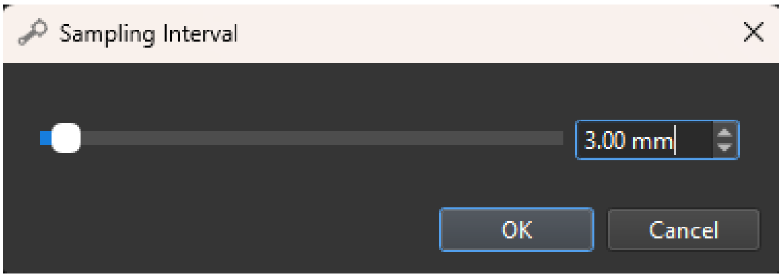
As the figure above, selected points are unwanted points and can be removed by following this step.



b. Downsample the point cloud.

Point cloud downsampling aims to reduce the number of points in the point cloud model, thus improving model matching efficiency.

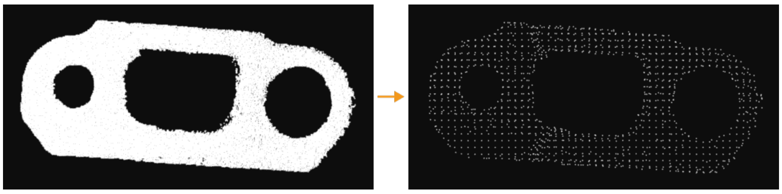
Click the ⁘ button and set the sampling interval in the pop-up window.



Sampling Interval

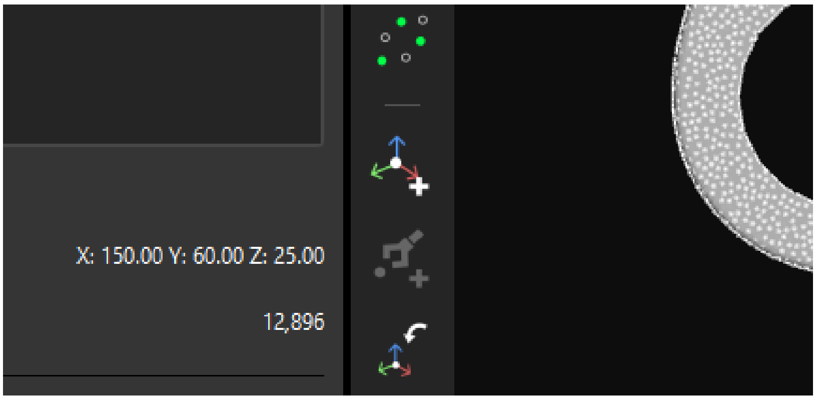
3.00 mm

In the figure below, the left image is a point cloud model before downsampling, and the right one is after downsampling with a sampling interval of 3 mm.

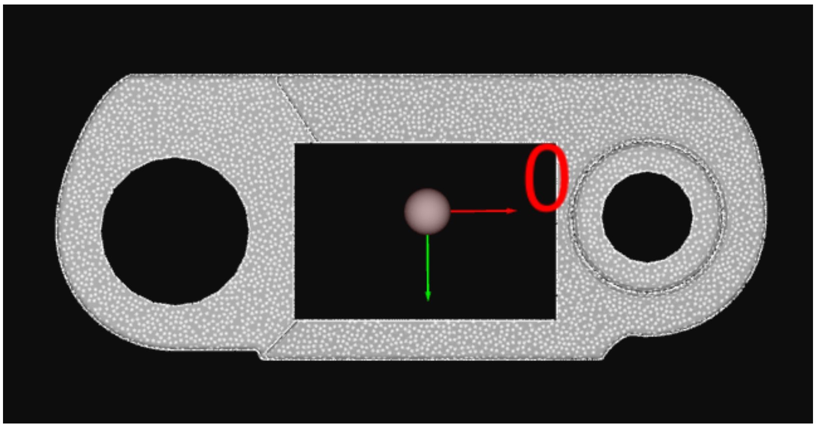


4. Add a pick point.

Click the ⊹ button on the toolbar to add a pose as a pick point to the point cloud model of the workpiece.

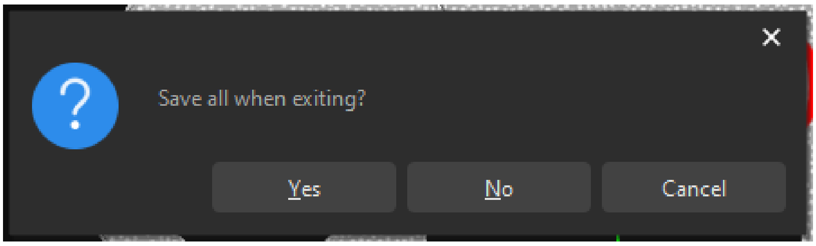


The following figure shows the added pick point.



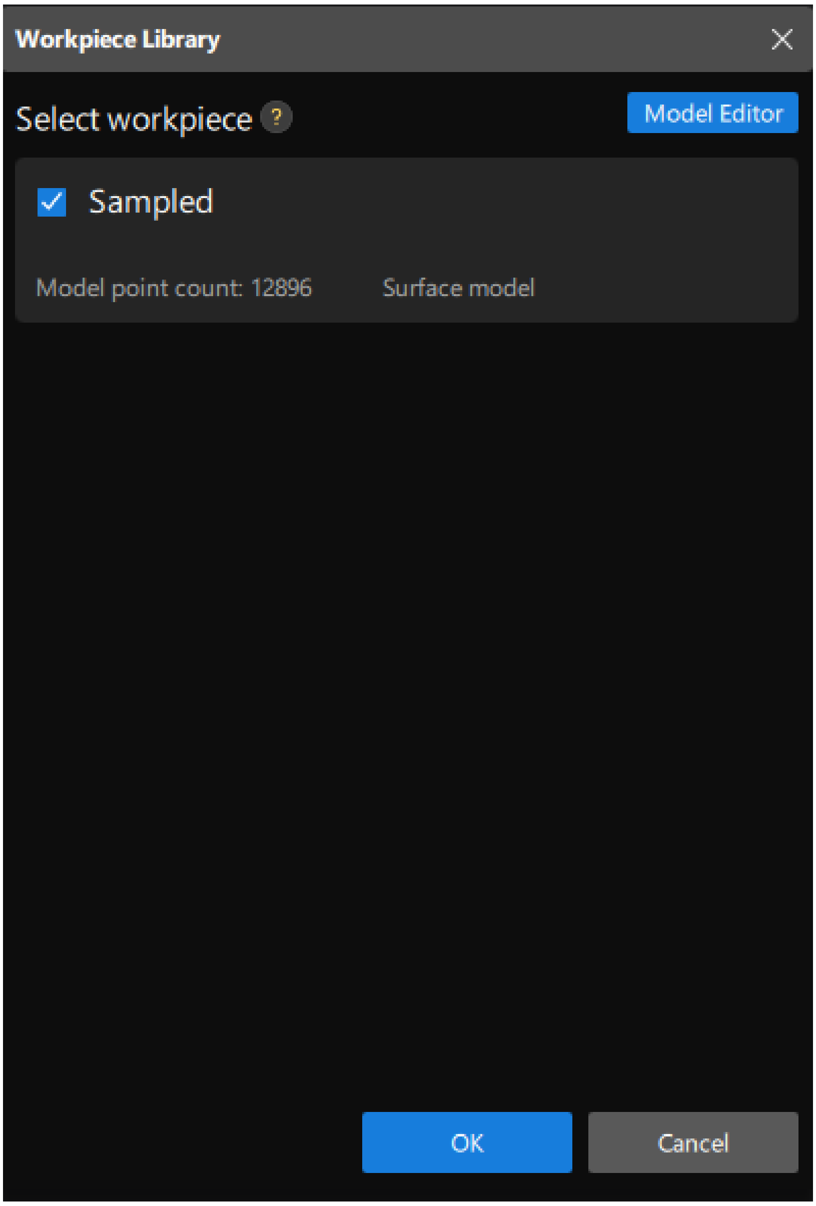
5. Save the model and the pick point.

   Close the Matching Model and Pick Point Editor, and click [Yes] in the pop-up window.

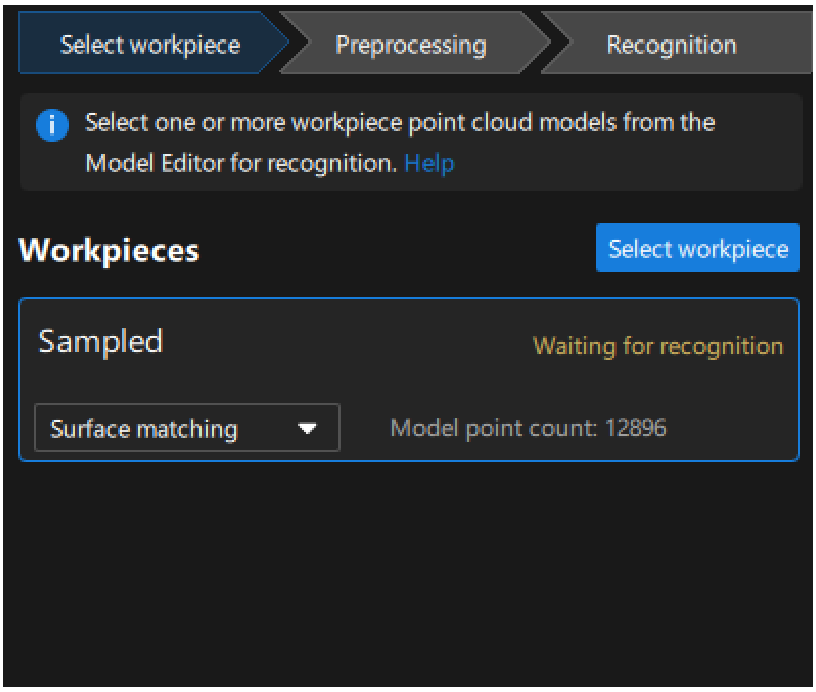

Save all when exiting?

6. Select this workpiece from the Workpiece Library.

   After closing the Matching Model and Pick Point Editor, select the saved point cloud model of the workpiece, and click [OK].



Subsequently, the target workpiece to recognize is displayed in the upper-right corner of the 3D Workpiece Recognition visualized configurator.



Select one or more workpiece point cloud models from the Model Editor for recognition. Help

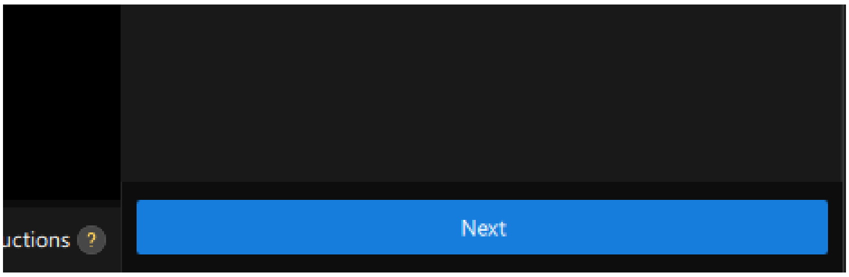
Sampled

Waiting for recognition

Surface matching

Model point count: 12896

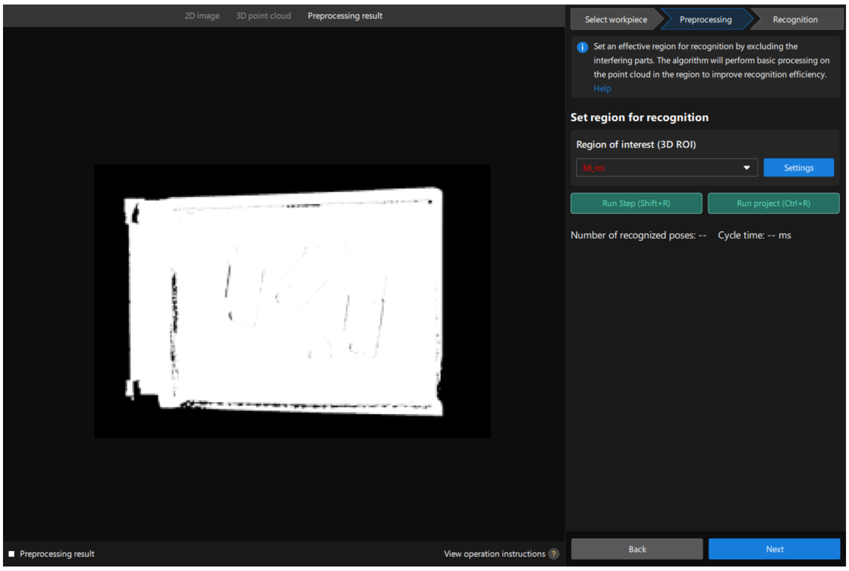
Now, you have selected the workpiece. Click **[Next]** on the bottom of the 3D Workpiece Recognition visualized configurator.



Preprocessing

Preprocessing is used to set an effective region for recognition to exclude the point cloud of unnecessary parts and keep only the point cloud of the workpiece, thus improving recognition efficiency.

The following figure displays the Preprocessing interface.



Set an effective region for recognition by excluding the interfering parts. The algorithm will perform basic processing on the point cloud in the region to improve recognition efficiency.

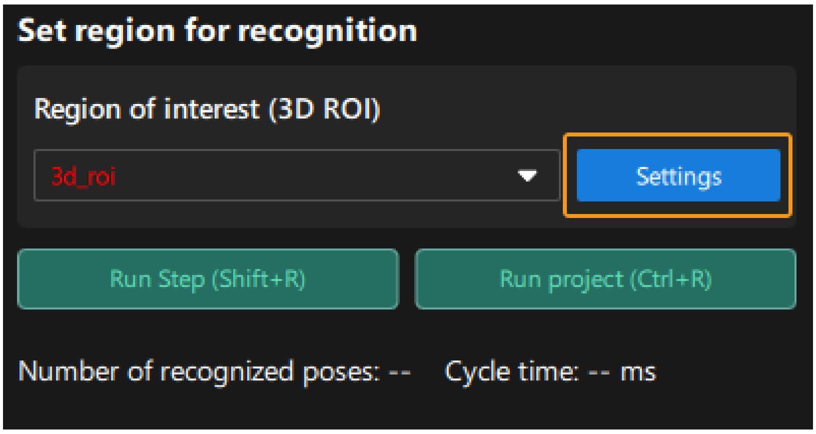
Set region for recognition

Region of interest (3D ROI): 3d_roi

Number of recognized poses: -- Cycle time: -- ms

1. Set the region for recognition.

   Click **[Settings]**.

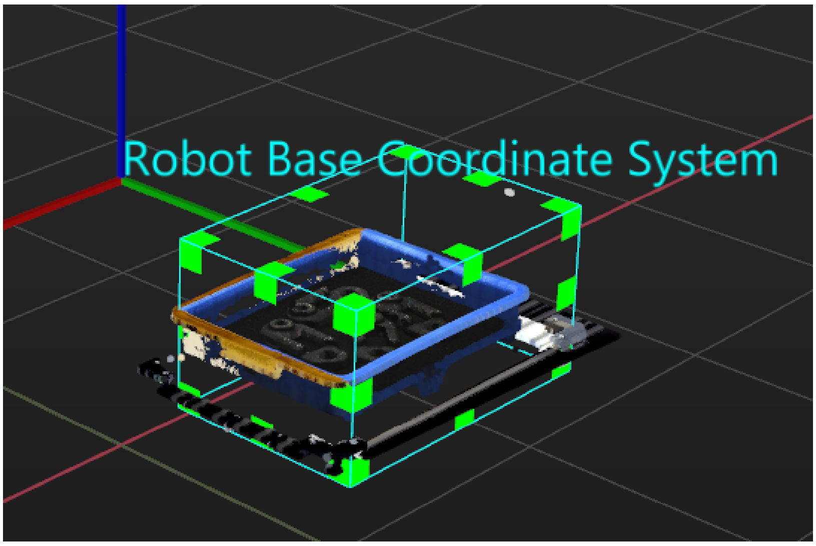
   

   Set region for recognition

   Region of interest (3D ROI): 3d_roi

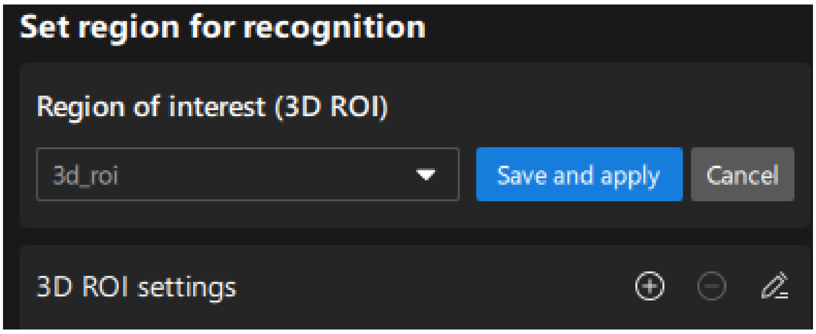
   Number of recognized poses: -- Cycle time: -- ms

   In visualized interface, set the region for recognition (3D ROI). Press and hold the `Ctrl` key, select the vertices of the 3D ROI, and drag the 3D ROI to the proper size. The following figure displays the set 3D ROI.

2. Save the region for recognition.

   Click **[Save and apply]** to save the region for recognition.

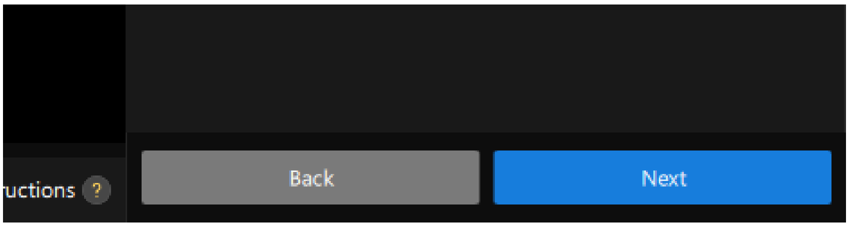


Set region for recognition

Region of interest (3D ROI)

3d_roi

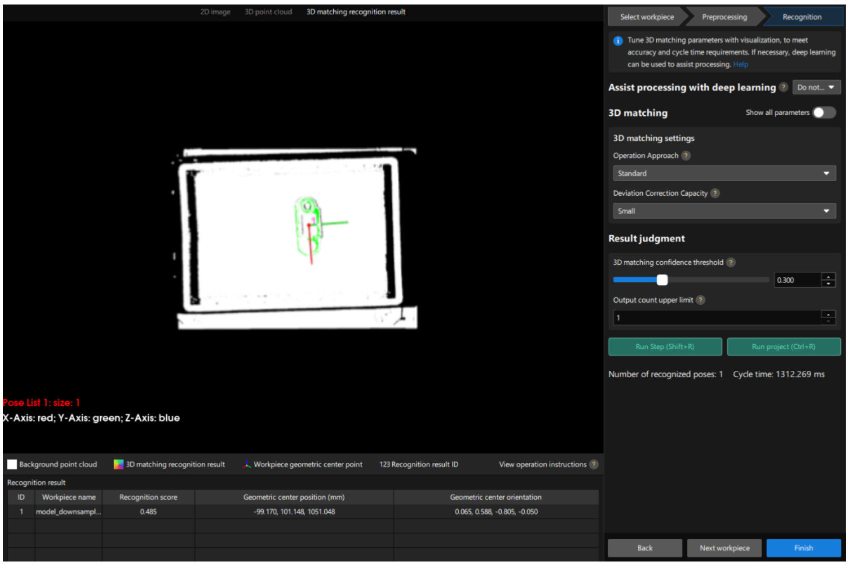
Now, you have finished the preprocessing procedure. Click **[Next]** at the bottom of the 3D Workpiece Recognition visualized configurator to enter the recognition procedure.



Recognize workpieces

In this procedure, you can adjust the 3D matching related parameters in a visualized manner, and output the workpieces' poses.

The following figure displays the Recognition interface.



2D image
3D point cloud
3D matching recognition result
Select workpiece
Preprocessing
Recognition
Background point cloud
3D matching recognition result
Workpiece geometric center point
123 Recognition result ID
View operation instructions ?
Run Step (Shift+R)
Run project (Ctrl+R)
Back
Next workpiece
Finish

Tune 3D matching parameters with visualization, to meet accuracy and cycle time requirements. If necessary, deep learning can be used to assist processing. Help

Assist processing with deep learning: Do not...

3D matching — Show all parameters

3D matching settings

Operation Approach: Standard

Deviation Correction Capacity: Small

Result judgment

3D matching confidence threshold: 0.300

Output count upper limit: 1

Number of recognized poses: 1   Cycle time: 1312.269 ms

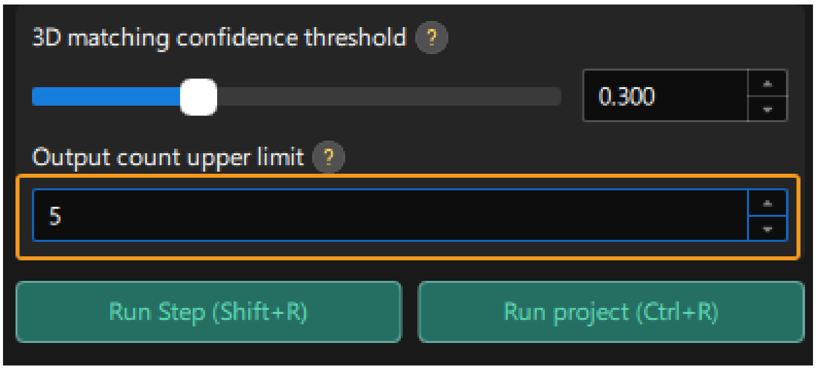
Pose List 1: size: 1

X-Axis: red; Y-Axis: green; Z-Axis: blue

Recognition result

ID	Workpiece name	Recognition score	Geometric center position (mm)	Geometric center orientation
1	model_downsampl...	0.485	-99.170, 101.148, 1051.048	0.065, 0.588, -0.805, -0.050

1. Since the robot needs to pick and place one workpiece based on the returned vision result in this tutorial, change the **Output count upper limit** parameter to 1 for this project.



3D matching confidence threshold ?

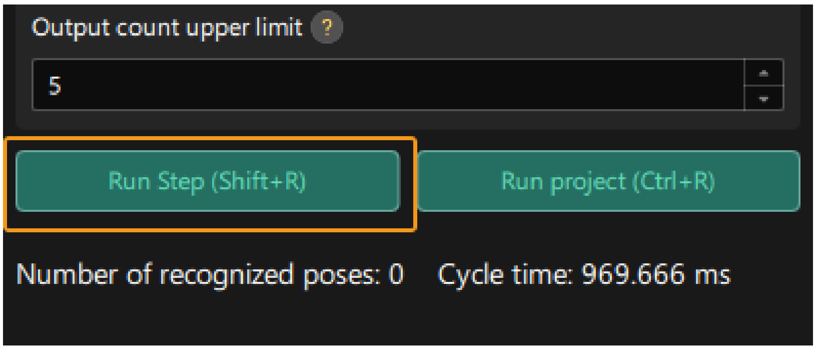
0.300

Output count upper limit ?

5

2. View the visualized output result.

   Click **[Run Step]**.

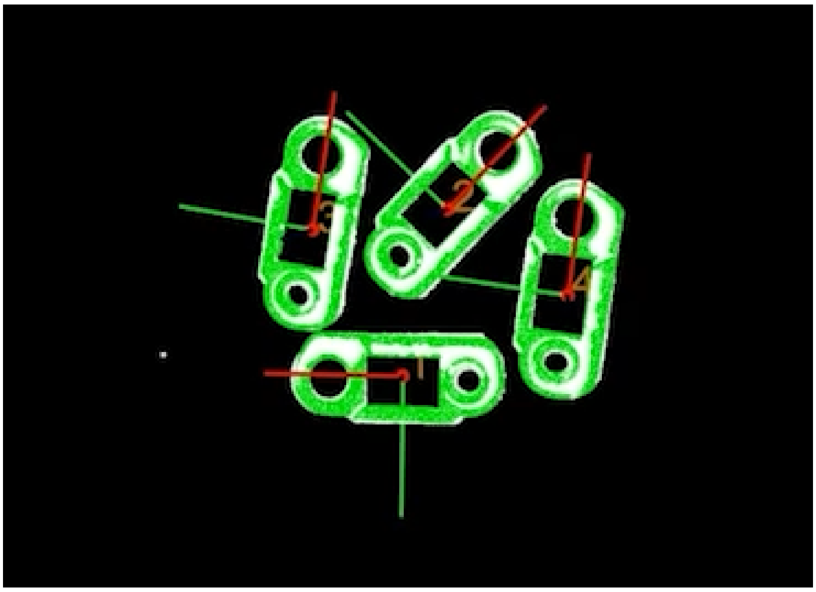


Output count upper limit ?

5

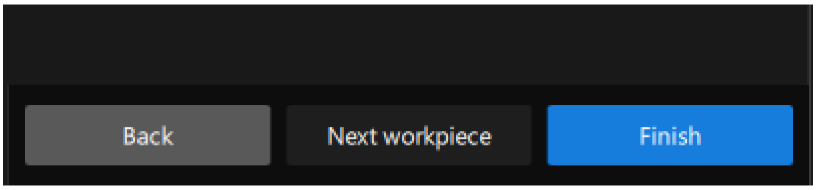
Number of recognized poses: 0   Cycle time: 969.666 ms

   You can view the visualized output result in the visualized area. As the figure below, the pose of one workpiece is output.

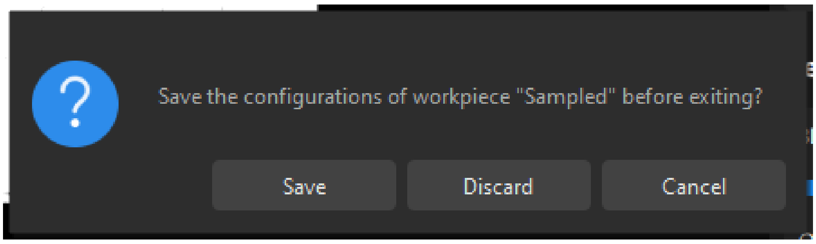


3. Save the configuration.

   Click the **[Finish]** button at the bottom of the 3D Workpiece Recognition visualized configurator.



In the pop-up window, click **[Save]**.


Save the configurations of workpiece "Sampled" before exiting?

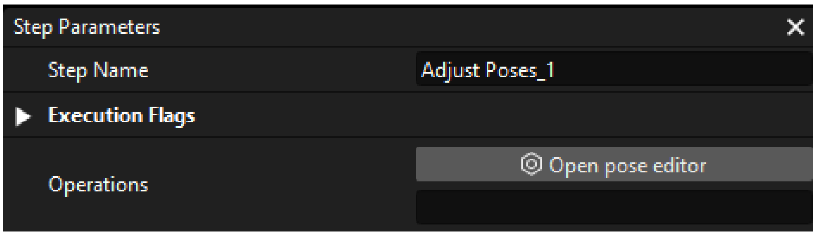
Now, you have recognized the workpiece and calculated its pick point.

Adjust poses

The pick points output by the “3D Workpiece Recognition” Step is in the camera reference frame. To facilitate robot picking, you need to adjust the workpieces’ poses to transform them from the camera reference frame to the robot reference frame.

1. Open the pose editor.

   Select the “Adjust Poses” Step, and click **[Open pose editor]** on the Step Parameters tab.

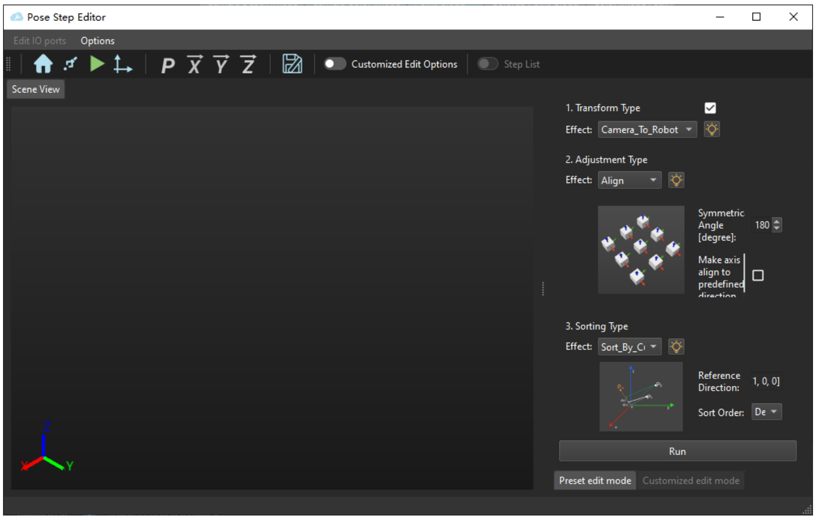

Step Parameters

Step Name: Adjust Poses_1

Execution Flags

Operations

   The following figure shows the pose editor.


Pose Step Editor

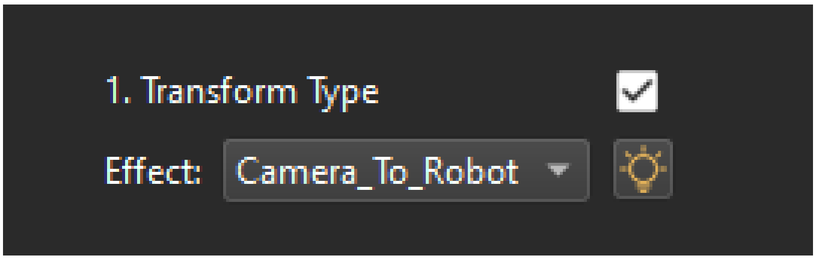
1. Transform Type ☑
Effect: Camera_To_Robot

2. Adjustment Type
Effect: Align
Symmetric Angle [degree]: 180
Make axis align to predefined direction ☐

3. Sorting Type
Effect: Sort_By_Cı
Reference Direction: 1, 0, 0]
Sort Order: De

2. Adjust the reference frame.

In the upper-left corner of the pose editor, set Effect to **Camera_To_Robot**.

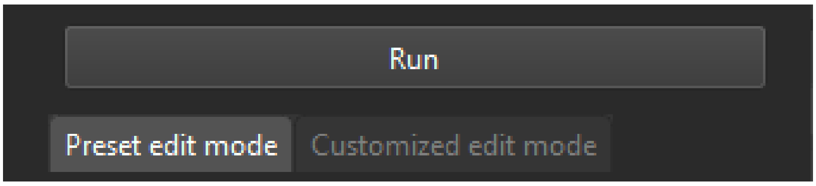


1. Transform Type ☑

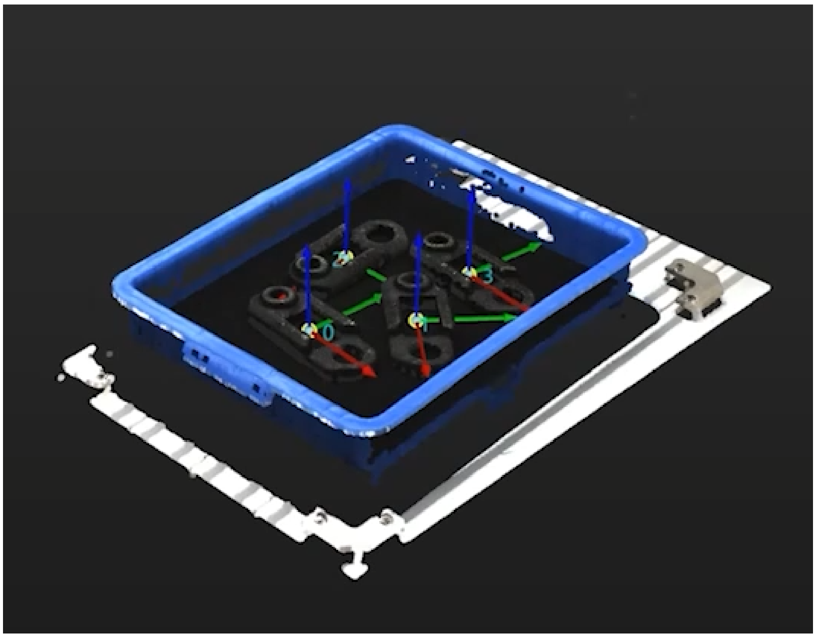
Effect: Camera_To_Robot

3. View the reference frame transformation result.

   At the bottom of the pose editor, click the **[Run]** button.

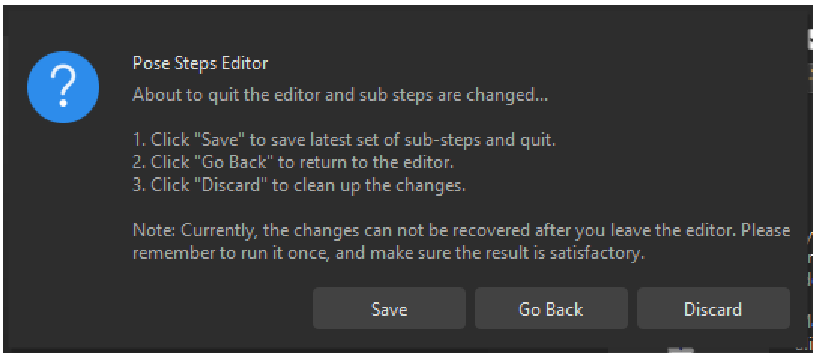


   You can see the transformed pick points in the Scene Viewer of the pose editor.



4. Save the configuration.

   Close the pose editor, and click **[Save]** in the pop-up window.



Pose Steps Editor

About to quit the editor and sub steps are changed...

1. Click "Save" to save latest set of sub-steps and quit.
2. Click "Go Back" to return to the editor.
3. Click "Discard" to clean up the changes.

Note: Currently, the changes can not be recovered after you leave the editor. Please remember to run it once, and make sure the result is satisfactory.

Now, the reference frame of the pick points has been transformed.

Procedure Out

The “Procedure Out” Step sends the results of the current project to the backend service.

Up to now, you have deployed the Mech-Vision project.

3.5. Pick and Place

In this tutorial, you will learn how to write a robot program to build a simple pick-and-place process.

In this tutorial, the robot program provides the following functions:

- When the program is run once, the Mech-Vision project is run once.
- Every time the Mech-Vision project is run, one vision point (TCP) can be obtained by the robot.
- The robot picks and places a workpiece based on the returned vision result and the programmed pick-and-place path.

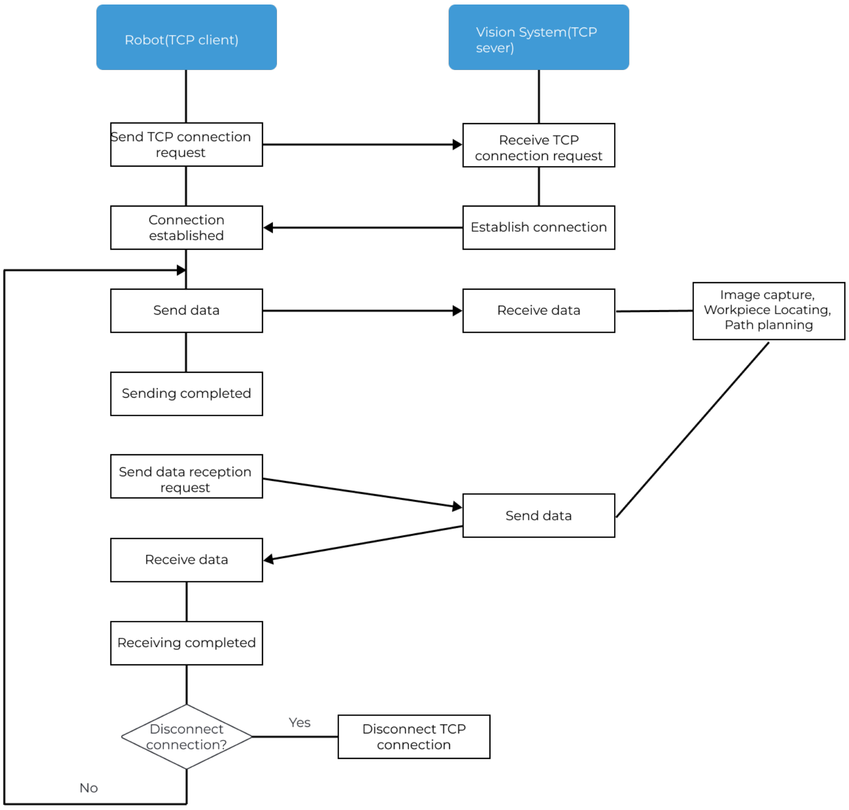
▶ https://www.youtube.com/watch?v=J1wru_bfjtI/PLVcMd7cW2rXVtrAejMyVQni2dUDv8bxje *(YouTube video)*

Video Tutorial: Configuration and Running

Introduction to Communication

This section describes the fundamental principles of communication between Yaskawa robots and the Mech-Mind Vision System briefly.

Yaskawa robots and the Mech-Mind Vision System use the Transmission Control Protocol (TCP) Socket protocol (in ASCII data format) for the Standard Interface communication. During the communication, the robot acts as a TCP client while the vision system acts as a TCP server. The process of communication between a robot and the vision system is shown below.



Introduction to Programming

This section provides some basic knowledge that you should know to write a pick-and-place program for Yaskawa robots.

The purpose of writing a robotic pick-and-place application program is to achieve Standard Interface communication between a robot and the vision system, as well as enabling the robot to perform pick-and-place tasks under vision guidance. This program contains a range of functions such as TCP connection establishment/disconnection, data transfer, robot motion control, digital output (DO) control, etc. When writing a pick-and-place program, you need to use two types of commands: Yaskawa robot commands and Standard Interface commands.

Yaskawa Robot Commands

YASKAWA robot commands refer to the commands supported by the YASKAWA INFORM robot language, including input/output commands, control commands, math commands, motion commands, shift commands, and additional commands. SectionYaskawa Robot Command Reference will briefly introduce the commands used for writing a pick-and-place program.

Standard Interface Commands

Standard Interface commands refer to the commands supported by the Standard Interface program loaded to the YASKAWA robot. These commands are used for Standard Interface communication with the vision system and include commands related to TCP socket communication, data exchange, and other functions. For the Standard Interface commands supported by YASKAWA robots, please refer to YASKAWA Standard Interface Functions.

The Standard Interface commands supported by Yaskawa robots are encapsulated in the frontend programs (Jobs) loaded to the robot. After loading frontend programs to the robot, you can use the YASKAWA command `CALL` to call a Standard Interface command encapsulated in a frontend program, for example: `CALL JOB:MM_OPEN_SOCKET`.

How to Write a Pick-and-Place Program?

The easiest way to write a pick-and-place program is to modify an existing example program and debug it.

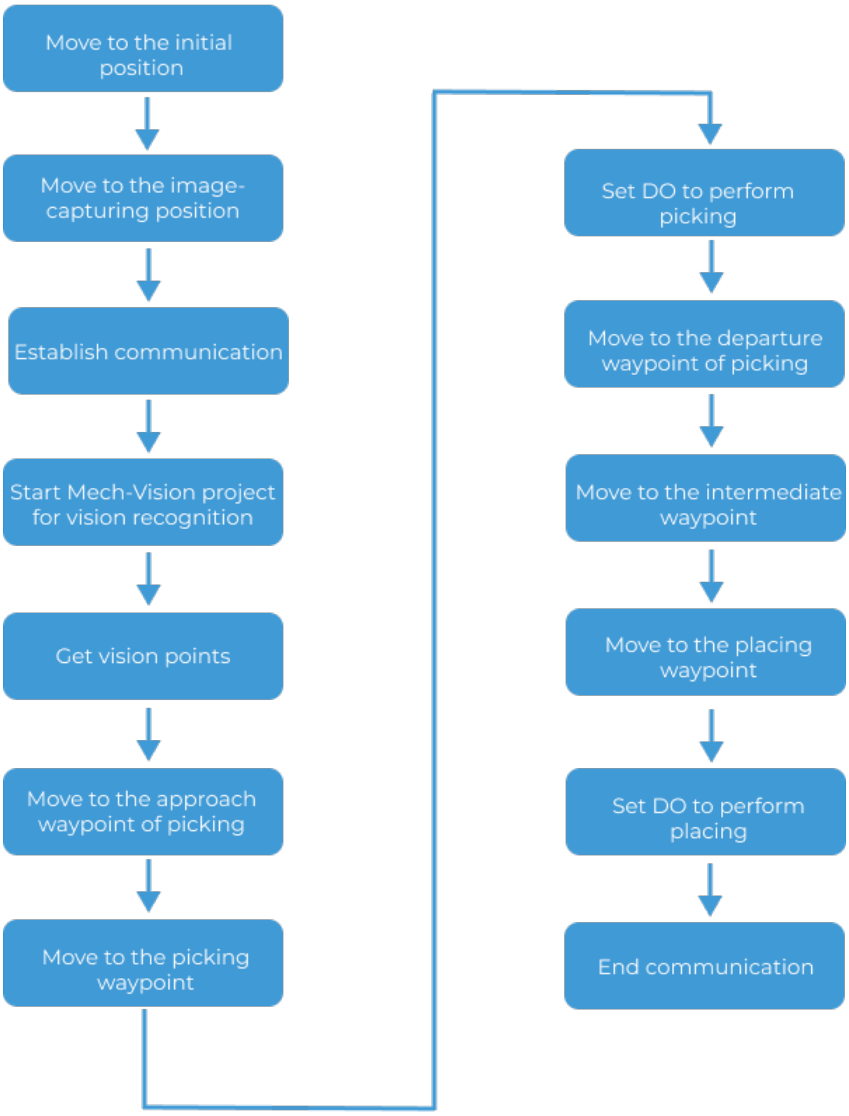
To facilitate the quick preparation of a pick-and-place program in collaboration with the Mech-Mind Vision System, Mech-Mind provides three pick-and-place example programs. This tutorial will modify the first example program and debug it. This example program enables a simple pick-and-place, where one vision point is obtained at a time from the vision system (only Mech-Vision used), and then picks and places the target object.

The code for this example program is displayed below:

```
NOP
'*****************************
'FUNCTION:simple pick and place
'with {product-vis}
'mechmind, 2022-5-27
'*****************************
'clear I50 to I69
CLEAR I050 20
'Initialize p variables
SUB P070 P070
SUB P071 P071
'set  200mm to Z
SETE P070 (3) 200000
'move to the home position
MOVJ VJ=80.00
'move to the camera position
MOVJ VJ=80.00 PL=0
'set ip address of IPC
CALL JOB:MM_INIT_SOCKET ("192.168.170.22;50000;1")
TIMER T=0.20
CALL JOB:MM_OPEN_SOCKET
'set vision recipe
CALL JOB:MM_SET_MODEL ("1;1")
TIMER T=0.20
'Run vision project
CALL JOB:MM_START_VIS ("1;0;2")
TIMER T=1.00
'get result from Vis
CALL JOB:MM_GET_VISDATA ("1;50;51;52")
TIMER T=0.20
PAUSE IF I052<>1100
'set the first pos to P071;
'set labels to I61;
'set speed to I62;
```

```
CALL JOB:MM_GET_POSE ("1;71;61;62")
SFTON P070
MOVJ P071 VJ=50.00
SFTOF
MOVL P071 V=80.0 PL=0
'enable gripper
DOUT OT#(1) ON
SFTON P070
MOVJ P071 VJ=50.00
SFTOF
MOVJ VJ=80.00
'drop point
MOVJ VJ=80.00 PL=0
'release gripper
DOUT OT#(1) OFF
CALL JOB:MM_CLOSE_SOCKET
END
```

The pick-and-place process built by the preceding example program is shown in the figure below.


Move to the initial position
Move to the image-capturing position
Establish communication
Start Mech-Vision project for vision recognition
Get vision points
Move to the approach waypoint of picking
Move to the picking waypoint
Set DO to perform picking
Move to the departure waypoint of picking
Move to the intermediate waypoint
Move to the placing waypoint
Set DO to perform placing
End communication

The following table describes each step in the pick-and-place process and the required modification.

Step	Code	Description

Initialize variables	`NOP`
	
	`'clear I50 to I69`
	`CLEAR I050 20`
	`'Initialize p variables`
	`SUB P070 P070`
	`SUB P071 P071`
	`'set  200mm to Z`
	`SETE P070 (3) 200000`
		This step initializes integer variables and position variables used in the program. Usually, no modification of the corresponding code block is required.

Step	Code	Description

Move to the initial position	`'move to the home position`
	`MOVJ VJ=80.00`
		This step moves the robot TCP to the initial position in the TEACH mode. The initial position is the starting point of the robot motion, and it should also be a safe position. At this position, the robot should be away from the objects to pick and the surrounding devices, and should not block the camera's field of view (FOV).
		Teach the initial position in this step.

Move to the image-capturing position	`'move to the camera position`
	`MOVJ VJ=80.00 PL=0`
		This step moves the robot TCP to the image-capturing position in the TEACH mode. The image-capturing position refers to the position where the robot captures images. At this position, the robot arm should not block the camera's FOV.
		Teach the image-capturing position in this step.

Establish communication	`'set ip address of IPC`
	`CALL JOB:MM_INIT_SOCKET("192.168.170.22;50000;1")`
	`TIMER T=0.20`
	`CALL JOB:MM_OPEN_SOCKET`
		This step initializes the parameters involved in communication and establishes TCP Socket communication with the vision system.
		In this step, you need to modify the corresponding parameters (the IP address and port of the IPC) of the "Initialize Communication" command (MM_INIT_SOCKET).

Start Mech-Vision project for vision recognition	`'set vision recipe`
	`CALL JOB:MM_SET_MODEL("1;1")`
	`'Run vision project`
	`CALL JOB:MM_START_VIS("1;0;2")`
		In this step, you need to switch the Mech-Vision parameter recipe (necessary for more than one workpiece), and then start Mech-Vision project to recognize the picking poses of workpieces. In this tutorial, only one type of workpiece is involved, so there is no need to switch the parameter recipe.
		In this step, you are required to comment out the "Switch Mech-Vision Recipe" command (MM_SET_MODEL) and modify the parameter of the "Start Mech-Vision Project" command.

Step	Code	Description

Get vision points	`'get result from Vis`
	`CALL JOB:MM_GET_VISDATA("1;50;51;52")`
	`TIMER T=0.20`
	`PAUSE IF I052<>1100`
	`'set the first pos to P071;`
	`'set labels to I61;`
	`'set speed to I62;`
	`CALL JOB:MM_GET_POSE("1;71;61;62")`
		This step obtains the vision point from Mech-Vision and saves the TCP, its corresponding label value, and speed value in variables.
		In this tutorial, the robot obtains a vision point from Mech-Vision at a time. You can use the code snippet in the example directly without modification.

Move to the approach waypoint of picking	`SFTON P070`
	`MOVJ P071 VJ=50.00`
	`SFTOF`
		This step moves the robot TCP to the approach waypoint of picking, which prevents the robot from colliding with scene objects (such as bins) when moving directly from the image-capturing position to the picking waypoint.
		The sample code moves the robot TCP to 200 mm above the picking waypoint (P071) by default (the approach waypoint of picking). If the distance between the approach waypoint of picking and the picking waypoint is too short, you can modify the Z-axis value of variable P070.
		`'set  200mm to Z`
		`SETE P070 (3) 200000`

Move to the picking waypoint	`MOVL P071 V=80.0 PL=0`	This step moves the robot TCP to the picking waypoint (P071). The example moves the robot TCP from the approach waypoint of picking to the picking waypoint in linear motion by default. You can use the code snippet in the sample directly without modification.

Step	Code	Description

Set DO to perform picking	`'enable gripper`
	`DOUT OT#(1) ON`
		This step expands the claw gripper to pick workpieces.
		The claw gripper used in this tutorial will expand to pick workpieces when the general output signal with ID 2 is connected. Therefore, this step requires modifying the general output signal ID to 2. In real application scenarios, please modify the general output signal ID according to the actual situation.

Move to the departure waypoint of picking	`SFTON P070`
	`MOVJ P071 VJ=50.00`
	`SFTOF`
		This step moves the robot TCP to the departure waypoint of picking, which prevents the robot from colliding with scene objects (such as bins) after picking the workpieces.
		The example program moves the robot TCP to 200 mm above the picking waypoint (P071) by default (the departure waypoint of picking). If the distance between the departure waypoint of picking and the picking waypoint is too short, you can modify the Z-axis value of variable P070.
		`'set  200mm to Z`
		`SETE P070 (3) 200000`
		ℹ The Z-axis value of variable P070 is shared by the approach waypoint of picking and the departure waypoint of picking as the offset from the Z-axis of the picking pose. If the Z-axis value of variable P070 is modified, both the approach waypoint of picking and the departure waypoint of picking will be affected.

Move to the intermediate waypoint	`MOVJ VJ=80.00 PL=0`	This step moves the robot TCP to the intermediate waypoint in the TEACH mode, which prevents the robot from colliding with scene objects. Teach the intermediate waypoint in this step. Add more intermediate waypoints according to your needs.

Step	Code	Description

Move to the placing waypoint	`'drop point`
	`MOVJ VJ=80.00 PL=0`
		This step moves the robot TCP to the placing waypoint in the TEACH mode.
		Teach the placing waypoint in this step.

Set DO to perform placing	`'enable gripper`
	`DOUT OT#(1) OFF`
		This step opens the claw gripper to place workpieces.
		The claw gripper used in this tutorial will open to place workpieces when the general output signal with ID 1 is connected. Therefore, this step requires modifying the state of the general output with signal ID 1 to ON. In real application scenarios, please modify the general output signal ID and state according to the actual situation.

End communication	`CALL JOB:MM_CLOSE_SOCKET`
	`END`
		This step ends TCP Socket communication. This line of code does not need to be modified.

For more information about the example program, please refer to Explanations to the Example Program.

Modify and Debug the Example Program

The robot pose in this example program is TCP. Before modifying and debugging the example program, please switch the tool number to the gripper No. (for example, 01) and complete the TCP settings of the tool.

Open the Example Program

To modify and debug the robot pick-and-place example program, you need to open the example program first:

1. In the **Online mode**, select JOB › SELECT JOB on the **Main Menu** to enter the JOB LIST interface.
2. Select the **mm_sample** program and press the `Select` key to open it.

The code snippet at the beginning of the example program initializes variables and usually does not need to be modified.

Move to the Initial Position

This step moves the robot TCP to the initial position in the TEACH mode. Teach the initial position for the robot here.

This step requires modifying the code snippet below.

```
'move to the home position
MOVJ VJ=80.00
```

Perform the following operations to teach the initial position for the robot:

1. Twist the mode key on teach pendant to `TEACH` to enter the **TEACH** mode.
2. Manually move the robot to the initial position so that the program can start executing from the beginning.
3. Press the ▼ key on the teach pendant to select the line "MOVJ VJ=80.00", press the `Modify` key, and then press the `Enter` key to confirm changing the initial position to the robot's current TCP.

"VJ=80.00" means the joint speed is 80%. This parameter can remain unchanged.

Move to the Image-capturing Position

This step moves the robot TCP to the image-capturing position in the TEACH mode. Teach the image-capturing position for the robot here.

This step requires modifying the code snippet below.

```
'move to the camera position
MOVJ VJ=80.00 PL=0
```

Perform the following operations to teach the image-capturing position for the robot:

1. In the **TEACH** mode, manually move the robot to the image-capturing position.
2. Press the ▼ key on the teach pendant to select the line "MOVJ VJ=80.00 PL=0", press the `Modify` key, and then press the `Enter` key to confirm changing the image-capturing position to the robot's current TCP.

"VJ=80.00" means the joint speed is 80%; "PL=0" means the position level is 0. These two parameters can remain unchanged. The position level is used to determine how close the robot is to the teaching position when it is in that position. The value range is from 0 to 8, with smaller values indicating closer proximity.

Establish Communication

This step begins the communication between the robot and the vision system. In this step, you need to modify the corresponding parameters (the IP address and port of the IPC) of the "Initialize Communication" command (MM_INIT_SOCKET).

The code snippet that needs to be modified is as follows.

```
'set ip address of IPC
CALL JOB:MM_INIT_SOCKET ("192.168.170.22;50000;1")
TIMER T=0.20
```

```
CALL JOB:MM_OPEN_SOCKET
```

Perform the following operations to modify the parameters of the “Initialize Communication” command:

1. In the **TEACH** mode, press the ▼ key on the teach pendant to select the line “CALL JOB:MM_INIT_SOCKET ("192.168.170.22;50000;1")” and press the `Select` key.
2. Use the soft keyboard on the teach pendant screen to modify "192.168.170.22" and "50000" to the IP address and port of the IPC respectively, and then press the `Confirm` key on the screen to confirm the modification.

Start Mech-Vision Project for Vision Recognition

This step starts the Mech-Vision project to perform vision recognition and output vision results.

In this step, you need to modify the parameters of the “Trigger Mech-Vision Project” command. In this tutorial, only one type of workpiece is involved, so you need to comment out the “Switch Mech-Vision Recipe” command.

The code snippet that needs to be modified is as follows.

```
CALL JOB:MM_SET_MODEL ("1;1")
'Run vision project
CALL JOB:MM_START_VIS ("1;0;2")
```

Perform the following operations to comment out the “Switch Mech-Vision Recipe” command and then modify the parameters of the “Trigger Mech-Vision Project” command:

1. In the **TEACH** mode, press the ▼ key on the teach pendant to select the line “CALL JOB:MM_SET_MODEL ("1;1")”, click the **Edit** tab, then select the **[Comment]** button and press the `Select` key on the teach pendant.

   When the project contains multiple workpieces, you need to use the parameter recipe. In this case, the second parameter needs to be modified to the parameter recipe No. corresponding to the current workpiece.

2. Press the ▼ key on the teach pendant to select the line “CALL JOB:MM_START_VIS ("1;0;2")” and press the `Select` key.
3. Use the soft keyboard on the teach pendant screen to modify the parameters of the "MM_START_VIS" command (for example, 1;1;0), and then press the `Confirm` key on the screen to confirm the modification.
   - The initial value “1” indicates the project ID of Mech-Vision. The project ID used in this tutorial is 1, therefore no modification is required.
   - The initial value “0” indicates the number of vision points obtained from Mech-Vision each time. In this tutorial, only one vision point is obtained at a time, therefore you need to modify this value to 1.
   - The initial value “2” indicates whether to send the robot’s pose. In this tutorial, there is no need to send the robot’s pose to Mech-Vision, so you need to modify this value to 0.

Get Vision Points

This step obtains the vision result from Mech-Vision and determines whether the result is obtained successfully. If the acquisition is successful, the pose of the vision point goes into the position variable P071; the label value corresponding to the pose goes into the integer variable I061; and the movement speed corresponding to the pose goes into the integer variable I062.

The code snippet corresponding to this step can be used directly without modification.

```
'get result from Vis
CALL JOB:MM_GET_VISDATA ("1;50;51;52")
PAUSE IF I052<>1100
'set the first pos to P071;
'set labels to I61;
'set speed to I62;
CALL JOB:MM_GET_POSE ("1;71;61;62")
```

Move to the Approach Waypoint of Picking

This step moves the robot TCP to the approach waypoint of picking.

- In order to avoid the robot’s end tool from colliding with the bin or other workpieces when reaching the picking waypoint, the robot TCP is usually moved to a certain distance directly above the picking waypoint (the approach waypoint of picking) and then moves in linear motion to the picking waypoint.
- You can change the Z-axis value of the picking waypoint pose to get the approach waypoint of picking pose.

The code snippet corresponding to this step can be used directly and usually does not need to be modified. In this tutorial, the Z-axis offset is 200 mm, which can be modified as needed.

```
'set  200mm to Z
SETE P070 (3) 200000
...
SFTON P070
MOVJ P071 VJ=50.00
SFTOF
```

Move to the Picking Waypoint

The step moves the robot TCP from the approach waypoint of picking to the picking waypoint in linear motion.

The code snippet corresponding to this step can be used directly without modification.

```
MOVL P071 V=80.0 PL=0
```

Set DO to Perform Picking

> The end tool used in this tutorial is an expanding claw gripper, which is controlled by an output signal. Connecting the general output signal with ID 2 will expand the claw gripper, while connecting the general output signal with ID 1 will open the claw gripper.

This step expands the claw gripper by connecting the output signal with ID 2 to pick workpieces. Therefore you need to modify the output signal ID to 2.

The code snippet that needs to be modified is as follows in this step.

```
'enable gripper
DOUT OT#(1) ON
```

Perform the following operations to modify the output signal ID to 2:

1. In the **TEACH** mode, press the ▼ key on the teach pendant to select the line “DOUT OT#(1) ON” and press the `Select` key.
2. Modify the output signal ID from 1 to 2, and then press the `Confirm` key on the screen to confirm the modification.

Move to the Departure Waypoint of Picking

The step moves the robot TCP from the picking waypoint to the departure waypoint of picking in linear motion.

> - In order to avoid the robot’s end tool from colliding with the bin or other workpieces when leaving after picking the workpieces, the robot TCP is usually moved to a certain distance directly above the picking waypoint (the departure waypoint of picking) and then leaves from this position for placement.
> - You can change the Z-axis value of the picking waypoint pose to get the departure waypoint of picking pose.
> - Usually the approach waypoint of picking and departure waypoint of picking are at the same position. In this tutorial, the approach waypoint of picking and departure waypoint of picking are both located 200 mm directly above the picking waypoint.

The code snippet corresponding to this step can be used directly and usually does not need to be modified.

```
'set  200mm to Z
SETE P070 (3) 200000
...

SFTON P070
MOVJ P071 VJ=50.00
SFTOF
```

Move to the Intermediate Waypoint

This step moves the robot TCP to the intermediate waypoint in the TEACH mode.

In order to prevent the robot from collision, an intermediate waypoint is usually added between the departure waypoint of picking and the placing waypoint. In this tutorial, only one intermediate waypoint is involved. You can add more waypoints to optimize the placing path in real application scenarios.

This step requires modifying the code snippet below.

```
MOVJ VJ=80.00
```

Perform the following operations to teach the intermediate waypoint for the robot:

1. In the **TEACH** mode, manually move the robot to a proper intermediate waypoint.
2. Press the ▼ key on the teach pendant to select the line "MOVJ VJ=80.00", press the `Modify` key, and then press the `Enter` key to confirm changing the intermediate waypoint to the robot's current TCP.

Move to the Placing Waypoint

This step moves the robot TCP to the placing waypoint in the TEACH mode.

This step requires modifying the code snippet below.

```
'drop point
MOVJ VJ=80.00 PL=0
```

Perform the following operations to teach the placing waypoint for the robot:

1. In the **TEACH** mode, manually move the robot to the placing waypoint.
2. Press the ▼ key on the teach pendant to select the line "MOVJ VJ=80.00 PL=0", press the `Modify` key, and then press the `Enter` key to confirm changing the placing waypoint to the robot's current TCP.

Set DO to Perform Placing

This step opens the claw gripper by connecting the output signal with ID 1 to place workpieces.

The code snippet corresponding to this step is as follows. You need to connect the output signal with ID 1, that is, change OFF to ON.

```
'release gripper
DOUT OT#(1) OFF
```

End Communication

This tutorial shows an example of a single pick-and-place. After finishing that, disconnect the communication by calling the "MM_CLOSE_SOCKET" command.

```
CALL JOB:MM_CLOSE_SOCKET
```

So far, the modification of the example program is completed. You can test it now.

Run and Test

1. Twist the mode key on the teach pendant to `PLAY` to enter the **REPEAT** mode.
2. Press the `SERVO ON READY` key on the teach pendant, and then press the blue **[Run]** button.

   The robot will automatically run the pick-and-place program to pick and place a workpiece.

> Please pay close attention to the robot's motion trajectory. In an emergency, press the emergency stop key.

Now you have completed a simple robot pick-and-place program.

Reference

Yaskawa Robot Command Reference

For a more detailed introduction to Yaskawa robot commands, please refer to the "YRC1000/YRC1000micro/DX200 INFORM Manual" that comes with the robot.

Command	Description	Grammar	Example
NOP	This is the program's initial command and does not perform any operations.	NOP	`NOP`
'	Specify comment.	' the string to be commented	`'Run vision project`
END	End the program.	END	`END`
CLEAR	Clear the values of variables after the specified index in Data 1 to 0, as specified by Data 2.	CLEAR <Data 1> <Data 2>	`CLEAR I050 20` Clear variables I50 to I69 (20 in total) to 0.
SUB	Subtract Data 2 from Data 1 and save the result to Data 1.	SUB <Data 1> <Data 2>	`SUB P070 P070` Subtract the variable P070 from the position variable P070 and save the result to the variable P070.
SETE	Set Data 2 to the elements of position variables in Data 1.	SETE <Data 1> <Data 2>	`SETE P070 (3) 200000` Increase the Z-axis value of variable P070 by an offset of 200,000 (in micrometers, equivalent to 200 millimeters).

Com mand	Description	Grammar	Example
MOVJ	Move the robot to the teaching position by using MOVJ instructions (in joint motion). This command is used within a range that does not limit the trajectory of the robot when it moves to the destination.	MOVJ file number of the robot teaching position VJ = joint speed (%)	`MOVJ VJ=80.00` The pose corresponds to when the robot joint moves to the teaching position.
MOVL	Move the robot to the teaching position by using the MOVL instructions (in linear motion). The robot moves to the pose in linear motion, which is recorded in the program after completing the teaching.	MOVL file number of the robot teaching position V = TCP speed (mm/s) PL = Position Level (0~8)	`MOVL P071 V=80.0 PL=0` The robot moves to the specified pose by P071 in linear motion.
CALL	Call the specified program.	CALL JOB: Name of the called program	`CALL JOB:MM_OPEN_SOCKET` Start TCP connection by calling the standard Interface command “MM_OPEN_SOCKET”.
TIME R	Stop the robot for a specified time.	TIMER T= time (s)	TIMER T=0.20 The robot stops for 0.2 seconds.
PAUS E	Pause the execution of the program.	PAUSE IF	`PAUSE IF I052<>1100` If the I052 variable value is not equal to 1100, program execution is paused.
IF	This command is used to determine various conditions when the robot is moving. Use this command after other commands for processing.	IF <Object 1> Operator (=, <>, etc.) <Object 1>	`PAUSE IF I052<>1100` If the I052 variable value is not equal to 1100, program execution is paused.

Command	Description	Grammar	Example
SFTON	The robot starts to move in parallel. The parallel moving distances are the incremental values of X-axis, Y-axis, and Z-axis in the frame, which are set in position variables.	SFTON No. of the position variable	SETE P070 (3) 200000 ... SFTON P070 MOVJ P071 VJ=50.00 SFTOF The robot moves to 200 mm above the picking waypoint (P071) in joint motion, that is, the approach waypoint of picking.
SFTOF	The robot finishes moving in parallel. This command is typically used in pairs with the SFTON command.	SFTOF	SFTOF
DOUT	This command is used to connect and disconnect the general output signal.	DOUT OT#(ID of the output signal) ON/OFF	DOUT OT#(1) ON Connect the general output signal with ID 1. DOUT OT#(1) OFF Disconnect the general output signal with ID 1.

Explanations to the Example Program

-
  ```
  NOP
  ```

  Description: This is the program’s initial command, which marks the start of the program.

-
  ```
  'clear I50 to I69
  CLEAR I050 20
  ```

  Description: Clear the values of integer variables I50 to I69 to zero. In this tutorial, variables I50 to I69 are mainly used to save states, number and label values.

-
  ```
  'Initialize p variables
  SUB P070 P070
  SUB P071 P071
  ```

  Description: Reset the values of position variables P070 and P071 to 0. In this tutorial, the variable P071 is used to save the picking pose. The Z-axis value of the P070 variable saves the offset of the approach waypoint of picking and the departure waypoint of picking on the Z-axis relative to the pick point.

-
  ```
  'set  200mm to Z
  SETE P070 (3) 200000
  ```

  Description: Save the Z-axis offset of the approach waypoint of picking and departure waypoint of picking relative to the picking waypoint as the Z-axis value of the P070 variable, in microns. In this tutorial, the default offset is 200000 microns, that is, 200 millimeters.

-
  ```
  'move to the home position
  MOVJ VJ=80.00
  ```

  Description: This command moves the robot TCP to the initial position in the TEACH mode. "VJ=80.00" means the joint speed is 80%. Teach the initial position here.

-
  ```
  'move to the camera position
  MOVJ VJ=80.00 PL=0
  ```

  Description: This command moves the robot TCP to the image-capturing position in the TEACH mode. "VJ=80.00" means the joint speed is 80%; "PL=0" means the position level is 0. The position level is used to determine how close the robot is to the teaching position when it is in that position. The value range is from 0 to 8, with smaller values indicating closer proximity.

-
  ```
  'set ip address of IPC
  CALL JOB:MM_INIT_SOCKET ("192.168.170.22;50000;1")
  ```

  Description: This command initializes communication parameters (the IP address and port of the IPC).

-
  ```
  TIMER T=0.20
  ```

  Description: Set the robot to stop for 0.2 seconds.

-
  ```
  CALL JOB:MM_OPEN_SOCKET
  ```

  Description: This command establishes TCP Socket communication with the vision system.

-
  ```
  'set vision recipe
  CALL JOB:MM_SET_MODEL ("1;1")
  ```

  Description: This command switches the parameter recipe of the project. When the project contains multiple workpieces, you need to use the parameter recipe.

-
  ```
  'Run vision project
  CALL JOB:MM_START_VIS ("1;0;2")
  ```

  Description: Start Mech-Vision Project for Vision Recognition. You need to set the parameters of the “MM_START_VIS” command correctly.

-
  ```
  'get result from Vis
  CALL JOB:MM_GET_VISDATA ("1;50;51;52")
  ```

  Description: This command is used to obtain vision points, and save the states of whether the vision points have been sent, the number of returned poses, and the command status code to variables I050~I052 respectively.

-
  ```
  PAUSE IF I052<>1100
  ```

  Description: If the command status code (I052 variable value) is not equal to 1100 (normal code), program execution is paused.

-
  ```
  'set the first pos to P071;
  'set labels to I61;
  'set speed to I62;
  CALL JOB:MM_GET_POSE ("1;71;61;62")
  ```

  Description: Save the first pose along with its corresponding label value and motion speed into variables P071, I061, and I062 for future reference from the vision result obtained.

-
  ```
  SFTON P070
  MOVJ P071 VJ=50.00
  SFTOF
  ```

  Description: Use the SFTON command to move the robot TCP in joint motion from the image-capturing position to the approach waypoint of picking. The Z-axis offset of the approach waypoint of picking relative to the picking waypoint is determined by the Z-axis value of variable P070.

-
  ```
  MOVL P071 V=80.0 PL=0
  ```

  Description: This command moves the robot TCP to the picking waypoint (the pose specified by the variable P071) in linear motion. The TCP speed is 80 mm/s and the position level is 0.

-
  ```
  'enable gripper
  ```

```
DOUT OT#(1) ON
```

Description: Control the gripper to pick workpieces by connecting the output signal with ID 1. In real application scenarios, you may modify the output signal ID according to the actual situation.

-
  ```
  SFTON P070
  MOVJ P071 VJ=50.00
  SFTOF
  ```

  Description: Use the SFTON command to move the robot TCP in linear motion from the picking waypoint to the departure waypoint of picking. The Z-axis offset of the departure waypoint of picking relative to the picking waypoint is determined by the Z-axis value of variable P070.

-
  ```
  MOVJ VJ=80.00
  ```

  Description: This command moves the robot TCP to the intermediate waypoint in the TEACH mode, which prevents the robot from colliding with scene objects. Teach the intermediate waypoint here.

-
  ```
  'drop point
  MOVJ VJ=80.00 PL=0
  ```

  Description: This command moves the robot TCP to the intermediate waypoint in the TEACH mode, which prevents the robot from colliding with scene objects. Teach the intermediate waypoint here.

-
  ```
  'release gripper
  DOUT OT#(1) OFF
  ```

  Description: Control the gripper to place workpieces by disconnecting the output signal with ID 1. In real application scenarios, you may modify the output signal ID and state according to the actual situation.

-
  ```
  CALL JOB:MM_CLOSE_SOCKET
  ```

  Description: This command ends TCP Socket communication with the vision system.

-
  ```
  END
  ```

  Description: End the program.

4. Getting Started Tutorial: Vision-Guided Robotic Depalletizing of Cartons (Master-Control Communication)

In this tutorial, you will learn how to deploy a simple 3D vision-guided robotic application of depalletizing cartons in the Master-Control communication mode.

Application Overview

- Camera: Mech-Eye DEEP camera, mounted in eye to hand (ETH) mode
- Robot: ABB_IRB_1300_11_0_9
- Workobject: single-case cartons
- Used software: Mech-Vision 1.7.4, Mech-Viz 1.7.4, Mech-Center 1.7.4, and Mech-Eye Viewer 2.1.0
- Communication mode: Master-Control communication

> If you are using a different camera model, robot brand, or workobject than in this example, please refer to the reference information provided in the corresponding steps to make adjustments.

▶ https://www.youtube.com/watch?v=9QrLGzeLI3Y/PLVcMd7cW2rXWbHhTQX8m2R1r6PhX6vrCI *(YouTube video)*

Getting Started Tutorial: Vision-Guided Robotic Depalletizing Overview

Explanation of Terms

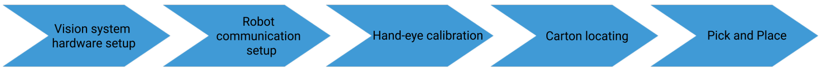
Pallet	A platform device used to hold goods (such as cartons)
Depalletizing	A task that a robot performs to disassemble objects from a pallet as required under the vision guidance
Palletizing	A task that a robot performs to place objects at specified positions on a pallet under the vision guidance
Single-case depalletizing	Depalletize single-case cartons (with the same dimensions) from a pallet
Mixed-case depalletizing	Depalletize single-case cartons (with different dimensions) from a pallet

Single-case depalletizing

Mixed-case depalletizing

How to Deploy a Vision-Guided Robotic Application?

The deployment of the vision-guided robotic application can be divided into five phases, as shown in the figure below:



The following table describes the five phases of deploying a vision-guided robotic application.

No.	Phase	Description
1	Vision system hardware setup	Install and connect hardware of the Mech-Mind Vision System.
2	Robot communication setup	Load the robot master-control program and the configuration files to the robot system and set up the communication between the vision system and the robot, thus helping the Mech-Mind Vision System obtain control over the robot.
3	Hand-eye calibration	Perform the automatic hand-eye calibration in the eye-to-hand setup, to establish the transformation relationship between the camera reference frame and the robot reference frame.
4	Carton locating	Use the case project “Single_Case_Cartons” to locate cartons and provide guidance for accurate robot picking.
5	Pick and place	Build the workflow for a Mech-Viz project to plan a collision-free robot path for picking and placing cartons.

Next, follow subsequent sections to complete the application deployment.

4.1. Vision System Hardware Setup

In this tutorial, you will learn how to build the Mech-Mind Vision System.

Follow the steps below to build the Mech-Mind Vision System: **Check the contents of the package** → **Install the hardware** → **Connect the network** → **Upgrade the software (optional)** → **Confirm** that the vision system can capture images normally.

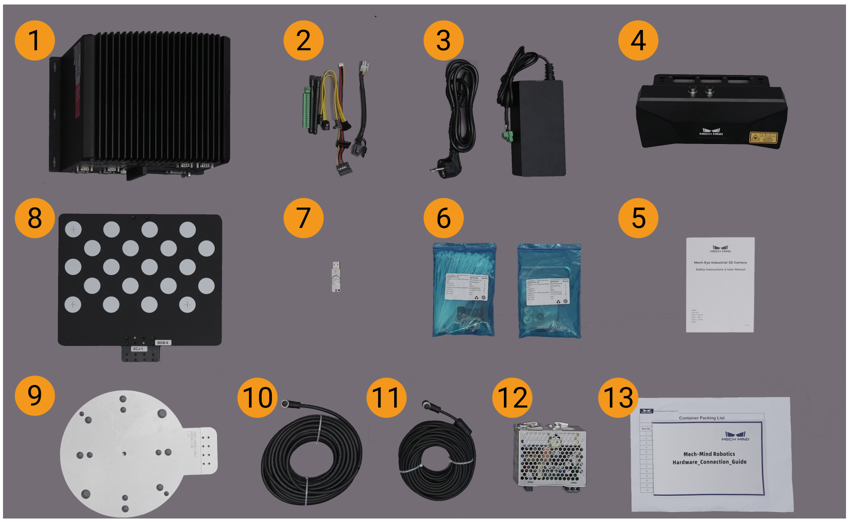
▶ https://www.youtube.com/watch?v=N8iwcoSjEdU/PLVcMd7cW2rXWbHhTQX8m2R1r6PhX6vrCI *(YouTube video)*

Video Tutorial: Vision System Hardware Setup

Check Contents of the Package

1. Make sure that the package is intact when you receive it.
2. Check the contents against the "packing list" in the package to ensure that no devices or accessories are missing or damaged.

The following figure shows the devices and accessories included in a vision system shipment. The table below is for reference only. Please take the "packing list" in the package as final.



No.	Category	Name	Function
1	IPC and accessories	IPC	Provides running environment for Mech-Mind software
2		IPC accessories	Provides IPC accessories, such as antennas for WIFI connection
3		IPC power cable and adapter	Supplies power to IPC
4	Camera and accessories	Mech-Eye Industrial 3D Camera	Captures images
5		User manual	Mech-Eye Industrial 3D Camera User Manuals
6		Camera accessory box	Mounts the camera

No.	Category	Name	Function
7	Project accessories	License dongle	Authorizes the Software Suite license
8		Calibration board	Calibrates the camera
9		Flange adapter	Connects the calibration board to the robot flange
10		Camera DC power cable (standard: 20 meters)	Connects the camera and the DIN rail power supply; Longer power cables are an option
11		Camera Ethernet cable (standard: 20 meters)	Connects the camera and the IPC; longer camera Ethernet cables are an option
12		DIN rail power supply (standard)	Supplies power to Mech-Eye Industrial 3D Camera; The camera power adaptor is an option
13	Packing list		Lists all the devices and accessories in the package

Contact Mech-Mind if any items are missing or damaged.

Prepare Other Materials

In this tutorial, besides the items in the package, you still need to prepare the materials shown in the following table by yourself.

Item	Function
Monitor	Provides display for the IPC
HDMI cable	Connects the monitor and the IPC
RJ45 Ethernet cable	Connects the IPC and the robot controller

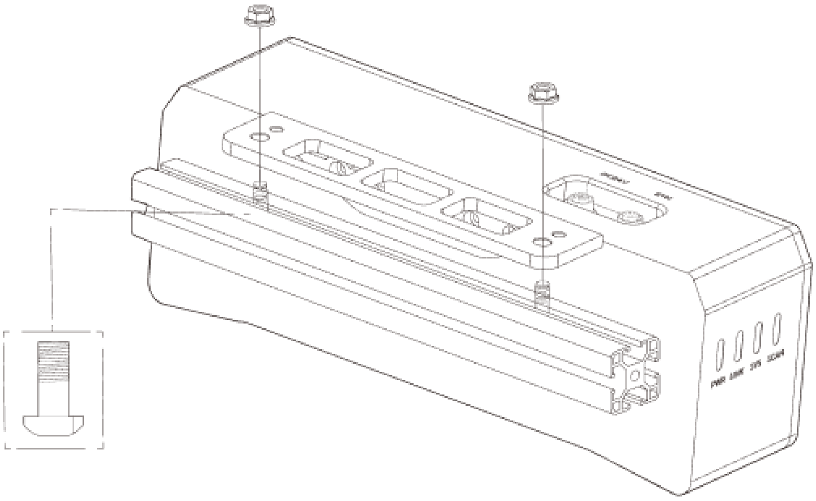
In this tutorial, the IPC and robot controller are directly connected through an RJ45-to-RJ45 Ethernet cable, and the IPC and the camera are directly connected through the camera Ethernet cable. Alternatively, you can use a router to connect the IPC and the robot controller, and the IPC and the camera, which is not covered in this topic.

Install Hardware

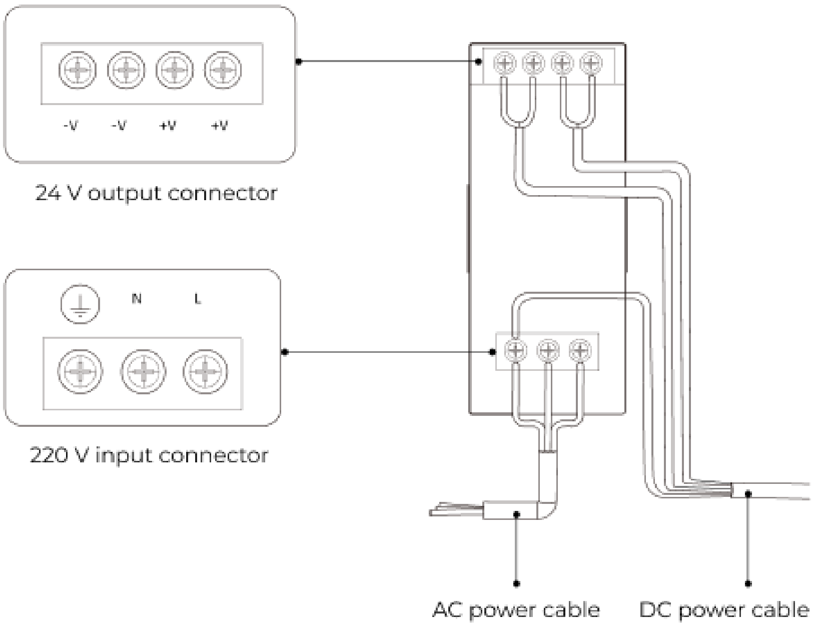
Mount the Camera

In this tutorial, the camera is mounted on the camera mounting frame (that is the eye to hand (ETH) mounting mode). In addition, the camera can also be mounted onto the end terminal of the robot (that is the eye in hand (EIH) mounting mode).

1. Get the camera mounting bolts and wrench from the camera accessory box.
2. Tighten the two bolts with the wrench as shown below.

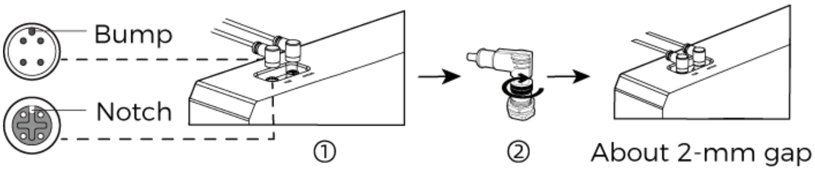


3. Please remove the lens protection film after mounting the camera.
4. Power the camera with the DIN rail power supply.
   - Connect the DC power cable:
     - Connect the +V wire to the +V connectors of the 24 V output connectors;
     - Connect the -V wire to the -V connectors of the 24 V output connectors;
     - Connect the PE wire to the 220 V input connector ⏚.



5. Install the Ethernet cable of the camera.

   Make sure the bump of the M12 connector and the notch of the ETH port, and tighten the nut after plugging in the cable.

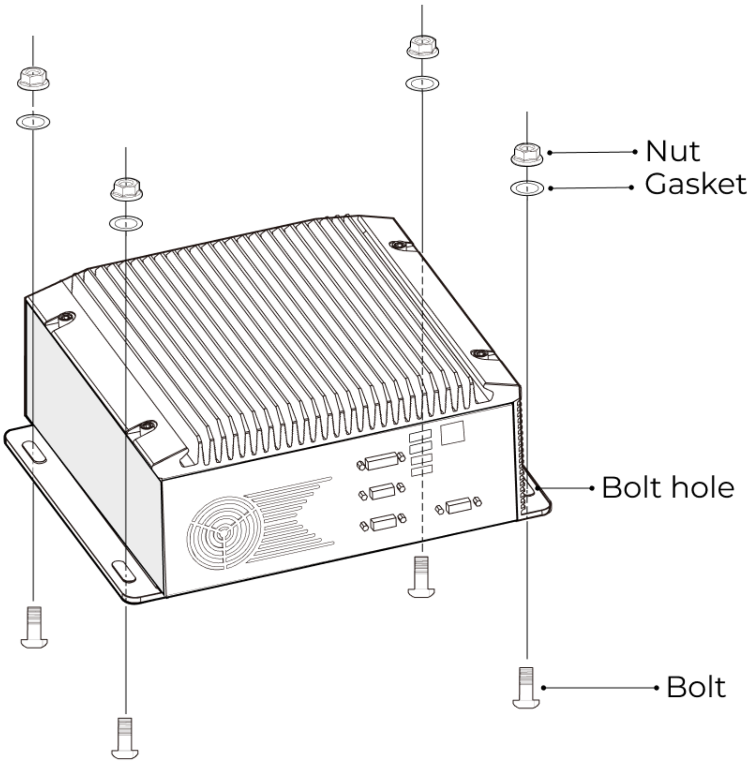


Mount the IPC

> The IPC is normally mounted in the control cabinet of the robot. The environment in which the IPC is mounted requires good heat dissipation, ventilation, and dust protection. It should be mounted at a location where Ethernet cables, HDMI cable, and USB ports can be easily mounted and serviced.

To mount the IPC, follow these steps:

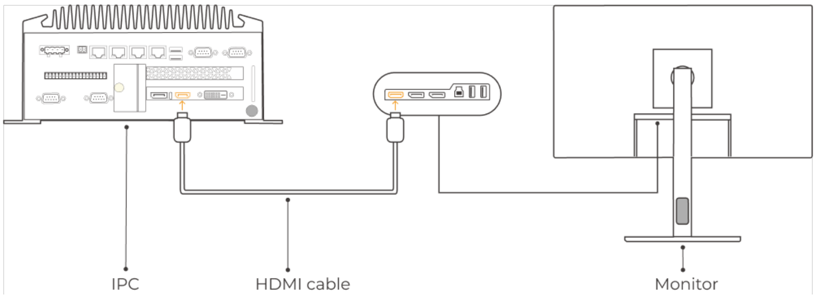
1. Prepare the wrench, bolts, nuts, and gaskets that are not included in the package beforehand.
2. If the robot controller is designed with mounting holes for the IPC in it, secure the IPC in the controller: place the bolt, gasket, and nut one by one, and tighten the two bolts with a wrench, as shown below.



> If the location of the robot controller is already fixed, skip this step and just place the IPC inside the controller.

3. Connect the IPC and the monitor with the HDMI cable.

   Plug one end of the HDMI cable into the HDMI port of the monitor, and the other end into the HDMI port of the IPC, as shown below.



4. Connect the IPC to the power supply unit with the power adaptor.

   Plug the power cable of the power adaptor into the power connector of the IPC. Connect the adaptor to the power supply on the other end.

5. Insert the license dongle.

   Plug the license dongle into a USB port of the IPC.

6. After the IPC is connected to the power supply, switch on the IPC.
   - If the IPC is started normally, the power indicator should be solid on.
   - If the IPC cannot be started, contact Mech-Mind Technical Support.

Connect the Network

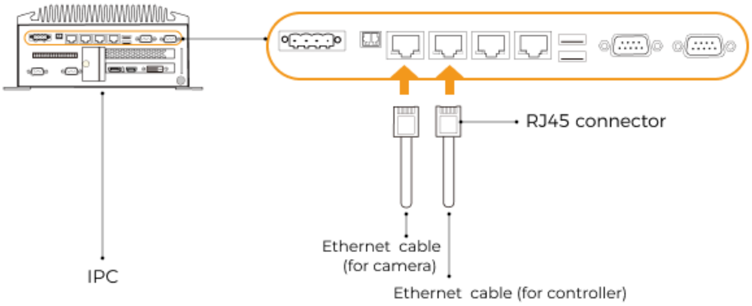
In this section, you will learn how to connect the network between the IPC and the camera, and between the IPC and the robot.

In the following sections, the following IP addresses will be used for network settings. Please adjust the network settings according to your actual network environment.

Device		IP Address
IPC	Ethernet port connecting to the camera	192.168.100.10
	Ethernet port connecting to the robot controller	192.168.200.10
Camera		192.168.100.20
Robot		192.168.200.20 (already set on the robot)

Connect the IPC and the Camera, and the IPC and the Robot Controller

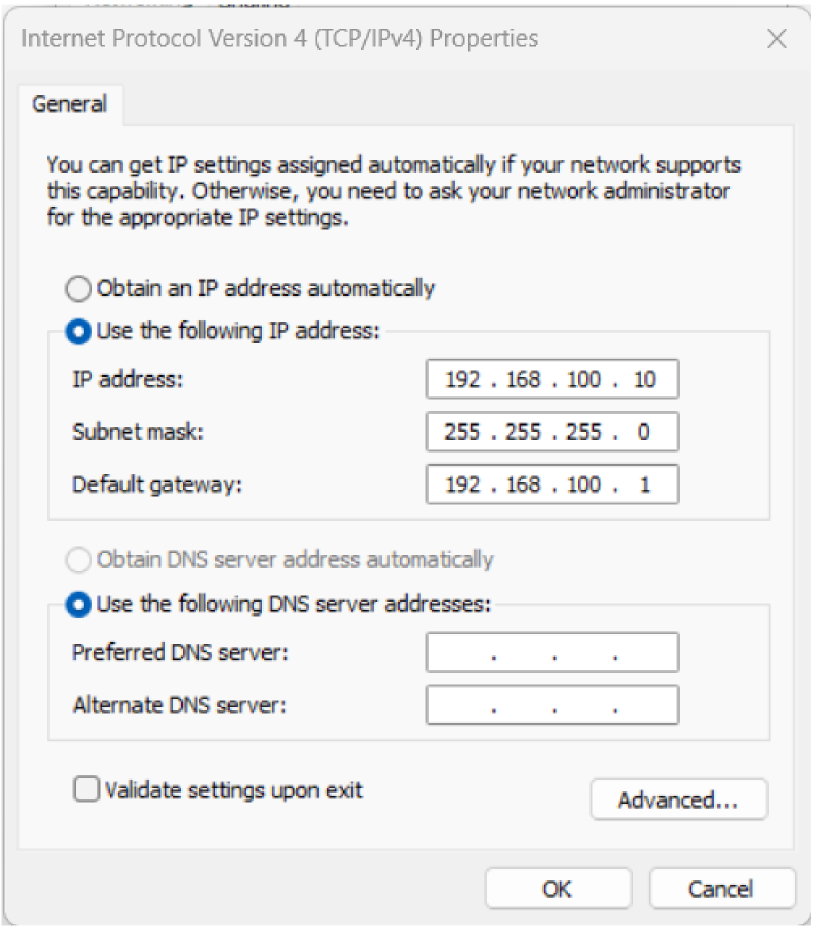
1. Plug the other end of the Ethernet cable connected to the camera into an Ethernet port of the IPC.



2. Use an RJ45-to-RJ45 Ethernet cable to plug one end into the Ethernet port of the robot controller and the other end into an Ethernet port of the IPC.

Set the IP Addresses on the IPC

1. Select Control Panel › Network and Internet › Network and Sharing Center › Change adapter settings on the IPC. The **Network Connections** page will be displayed.
2. Right-click the Ethernet port connected to the camera, and select **Rename** to rename the Ethernet port, such as “To_camera”.
3. Right-click the Ethernet port connected to the camera, and select **Properties** to enter the **Ethernet Properties** page.
4. Select the **Internet Protocol Version 4 (TCP/IPv4)** checkbox, and then click the **[Property]** button to enter the **Internet Protocol Version 4 (TCP/IPv4) Properties** page.
5. Select the **Use the following IP address** radio button, set the **IP address** parameter to “192.168.100.10”, **Subnet mask** to “255.255.255.0”, and **Default gateway** to “192.168.100.1”, and then click the **[OK]** button.



Internet Protocol Version 4 (TCP/IPv4) Properties

General

You can get IP settings assigned automatically if your network supports this capability. Otherwise, you need to ask your network administrator for the appropriate IP settings.

○ Obtain an IP address automatically

● Use the following IP address:

IP address: 192 . 168 . 100 . 10

Subnet mask: 255 . 255 . 255 . 0

Default gateway: 192 . 168 . 100 . 1

○ Obtain DNS server address automatically

● Use the following DNS server addresses:

Preferred DNS server: . . .

Alternate DNS server: . . .

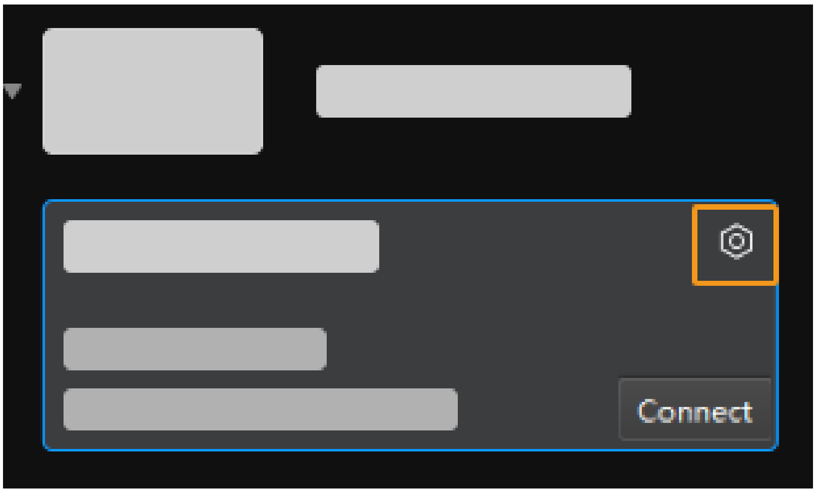
□ Validate settings upon exit

6. Repeat steps 2 to 5 to rename the Ethernet port connected to the robot controller (for example, “To_robot”), and set the IP address for this Ethernet port. For example, set the IP address of this Ethernet port to “192.168.200.10”.

> The IP address of the robot and that of the IPC Ethernet port connected to the robot controller must be on the same subnet.

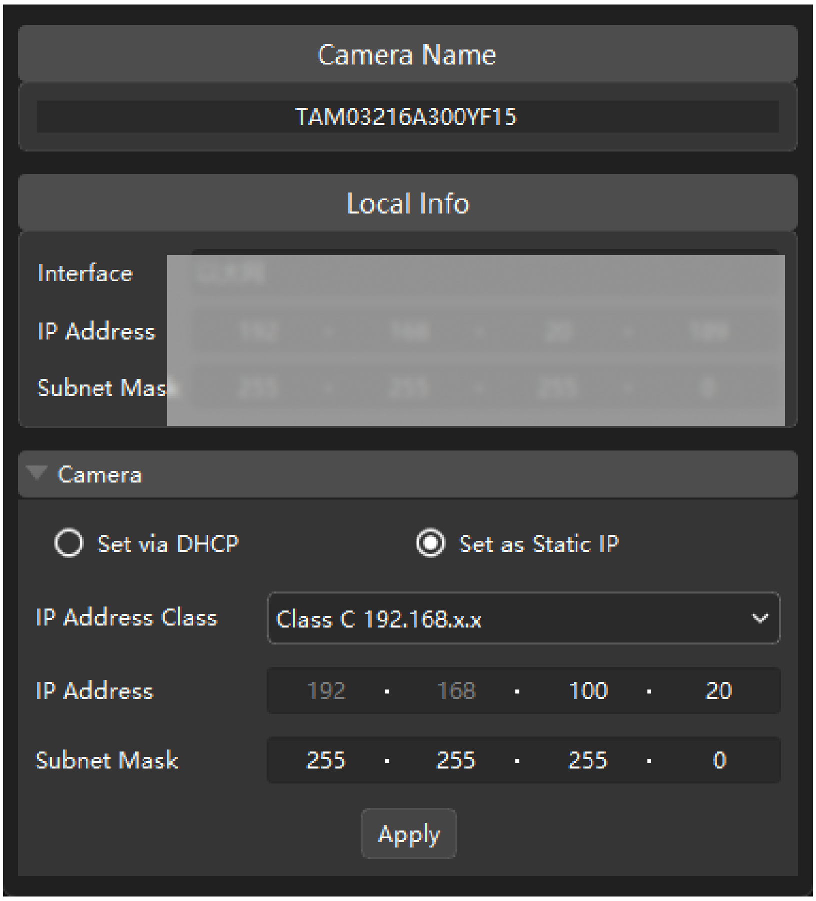
Set the Camera IP Address

1. Double-click EYE on the desktop of the IPC to open and run Mech-Eye Viewer.
2. Select the camera in the camera list, and hover the cursor on the camera. Click ⚙ to open the **Config Camera Network** dialog box.



If the camera cannot be found or connected, please refer to Camera Troubleshooting.

3. In the **Camera** area, select the **Set as Static IP** radio button, set **IP Address Class** to “Class C 192.168.x.x”, **IP Address** to **192.168.100.20**, and **Subnet Mask** to “255.255.255.0” and then click **[Apply]**.



Camera Name

TAM03216A300YF15

Local Info

Interface [redacted]

IP Address

Subnet Mask

Camera

○ Set via DHCP ● Set as Static IP

IP Address Class: Class C 192.168.x.x

IP Address: 192 . 168 . 100 . 20

Subnet Mask: 255 . 255 . 255 . 0

The IP address of the camera and that of the IPC Ethernet port connected to the camera must be on the same subnet.

Test the Network Connectivity

1. Press `Win` + `R` to open the **Run** dialog box.

2. Type cmd in the **Run** dialog box, and then click **[OK]**.
3. Type **ping XXX.XXX.XX.XX** in the command prompt window and press **[Enter]** to execute the command.

💡 Replace XXX.XXX.XX.XX with the actual IP address of the camera/robot.

If the network connectivity is normal, you should receive the following response:

```
Pinging XXX.XXX.XX.XX with 32 bytes of data:
Reply from XXX.XXX.XX.XX: bytes=32 time<1ms TTL=128
Reply from XXX.XXX.XX.XX: bytes=32 time<1ms TTL=128
Reply from XXX.XXX.XX.XX: bytes=32 time<1ms TTL=128
Reply from XXX.XXX.XX.XX: bytes=32 time<1ms TTL=128
```

Upgrade the Software (Optional)

The IPC purchased from Mech-Mind already has the latest version of Mech-Mind software installed.

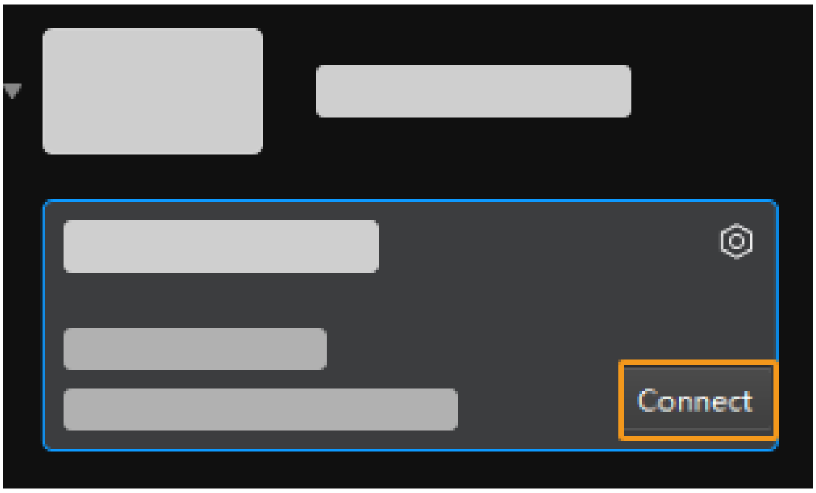
Please check if all software on the IPC is running with the latest version. If so, skip this section; if not, follow the sections below to update the software to the latest versions.

- Downloading and installing Mech-Eye SDK
- Downloading and installing Mech-Vision

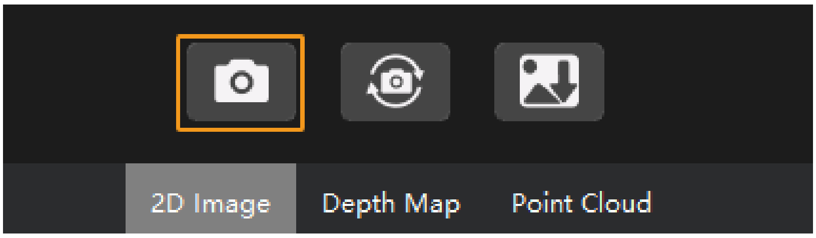
Confirm That the Vision System Can Capture Images Normally

After testing the network connectivity between the IPC and the camera/robot, please confirm that the vision system can capture images normally:

1. Place workpieces at the center of the camera’s field of view (FOV), and make sure that the workpieces on the edge and top are all within the FOV.
2. Open and start Mech-Eye Viewer by double-clicking EYE on the desktop of the IPC.
3. Select the camera in the camera list and click **[Connect]**.

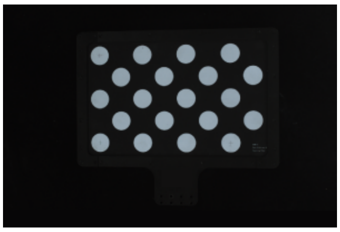
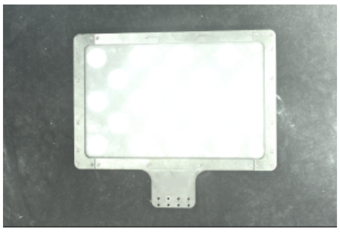
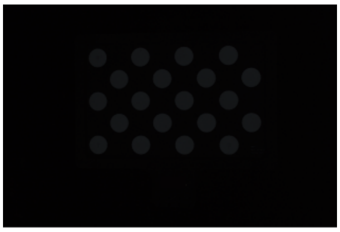
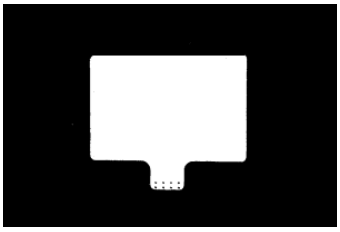
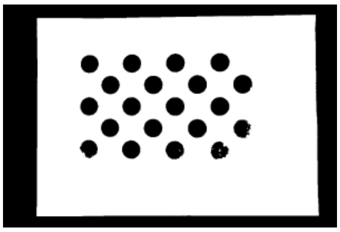
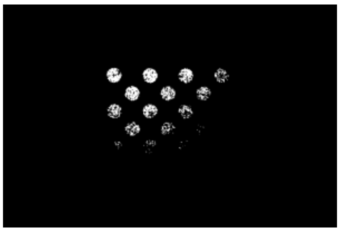


4. After the camera is connected, click **[Capture once]**.



5. Make sure that the captured images meet the following standard.

   - 2D image: the captured 2D image is not significantly overexposed (too white to recognize objects) or underexposed (too dark to recognize object details).
   - Depth map and point cloud: All relevant parts of the object can be recognized.

	Normal	Overexposed	Underexposed
2D image			
Point cloud			

If the captured images do not meet the required standard, please use Mech-Eye Viewer to adjust camera parameters.

Up to now, you have learned how to build the vision system.

4.2. Robot Communication Setup

In this tutorial, you will learn how to load the Master-Control program files to the ABB robot and configure the Master-Control communication.

- In this section, you will load the robot Master-Control program and the configuration files to the robot system to establish the communication between the vision system and the robot, thus helping the Mech-Mind Vision System obtain control over the robot.
- If you are using the robots of other brands, you can find instructions on setting up the Master-Control communication configuration with the desired robot from the section Master-Control Communication.

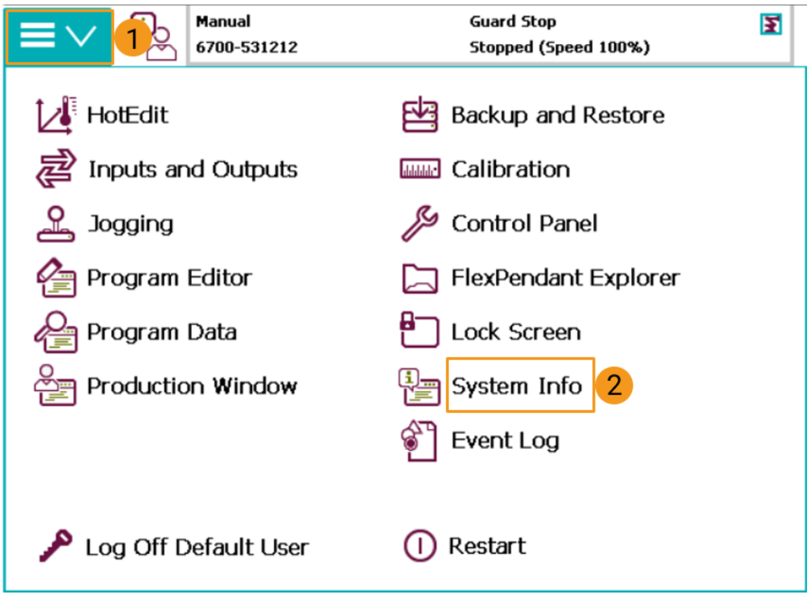
▶ https://www.youtube.com/watch?v=kPFvHWFKlYA/PLVcMd7cW2rXWbHhTQX8m2R1r6PhX6vrCI *(YouTube video)*

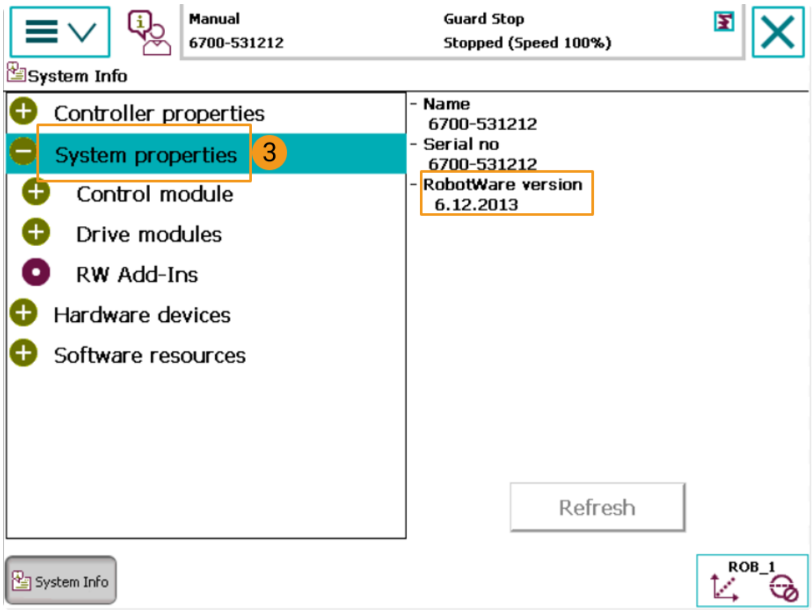
Video Tutorial: Robot Interface Configuration

Preparation

Check the Controller and RobotWare Version

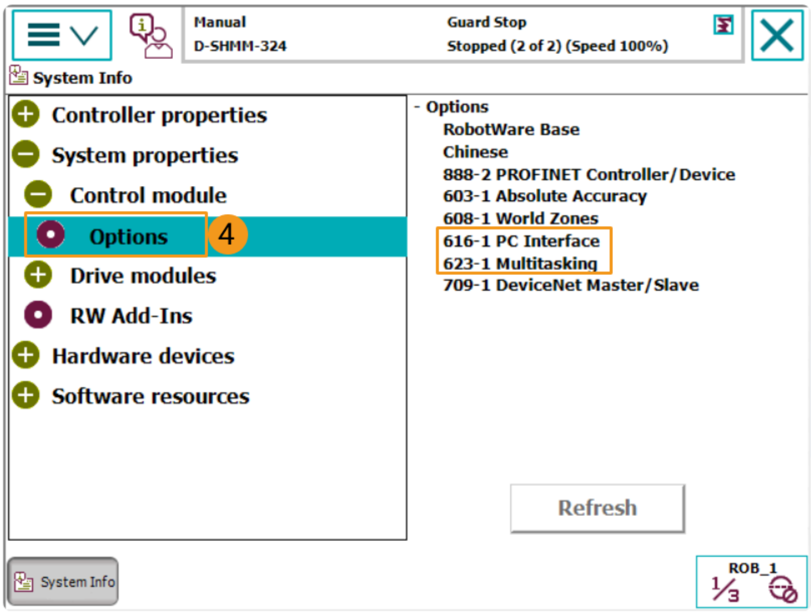
1. Make sure that D652 or DSQC1030 IO board has been installed on the robot controller.
2. Confirm that the RobotWare version is 6.0 or above on the teach pendant.





- Name
  6700-531212
- Serial no
  6700-531212
- RobotWare version
  6.12.2013

3. Confirm that the following control modules have been installed on the teach pendant.
   - 623-1 Multitasking
   - 616-1 PCInterface



- Options
  - RobotWare Base
  - Chinese
  - 888-2 PROFINET Controller/Device
  - 603-1 Absolute Accuracy
  - 608-1 World Zones
  - 616-1 PC Interface
  - 623-1 Multitasking
  - 709-1 DeviceNet Master/Slave

If the preceding conditions cannot be met, the vision side cannot communicate with the robot through the master-control mode. Please contact the vendor of your robot.

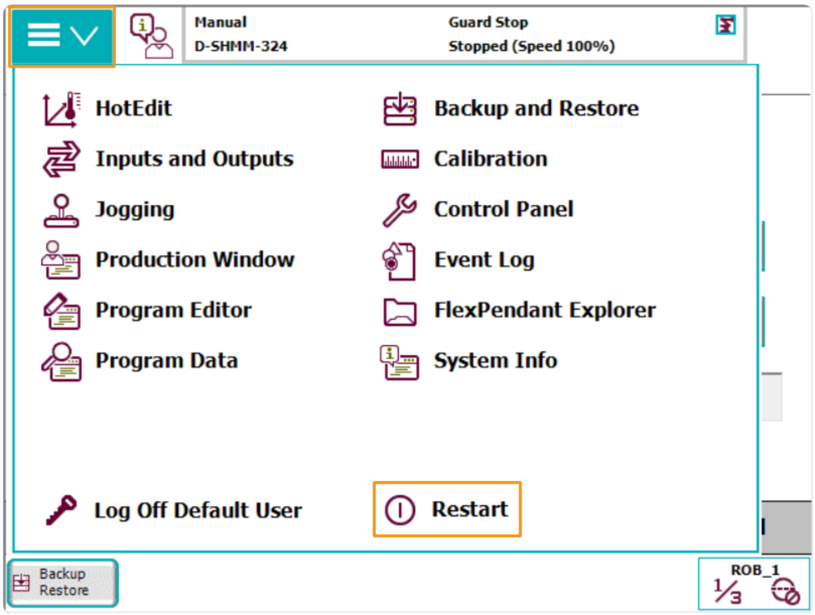
Reset the Robot System

After checking the software and hardware status of the controller, reset the robot system. If you are using a new robot, skip this section.

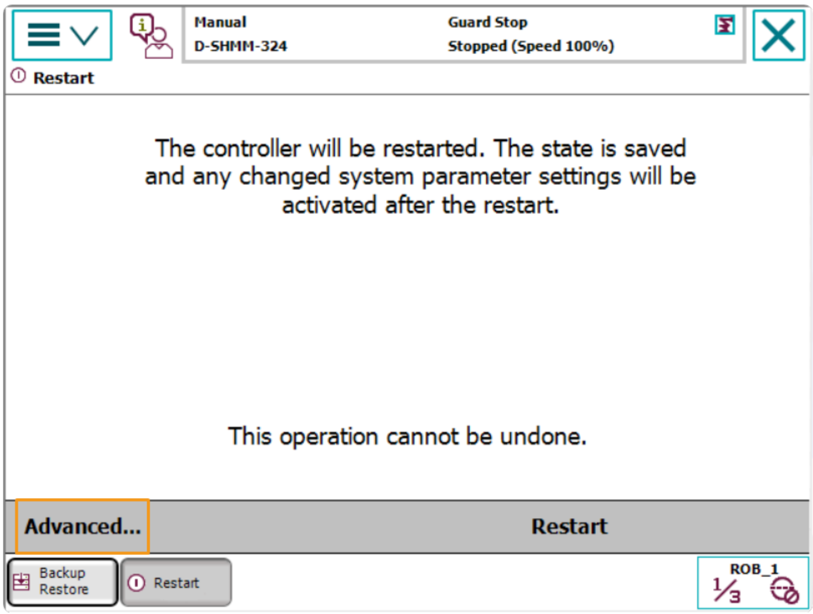
Resetting the system will restore the factory settings. Please make sure that you have completed the backup operation.

To reset the robot system, follow these steps:

1. On the teach pendant, go to the home page using the menu in the upper-left corner, and then press [Restart].



2. Press [Advanced...].



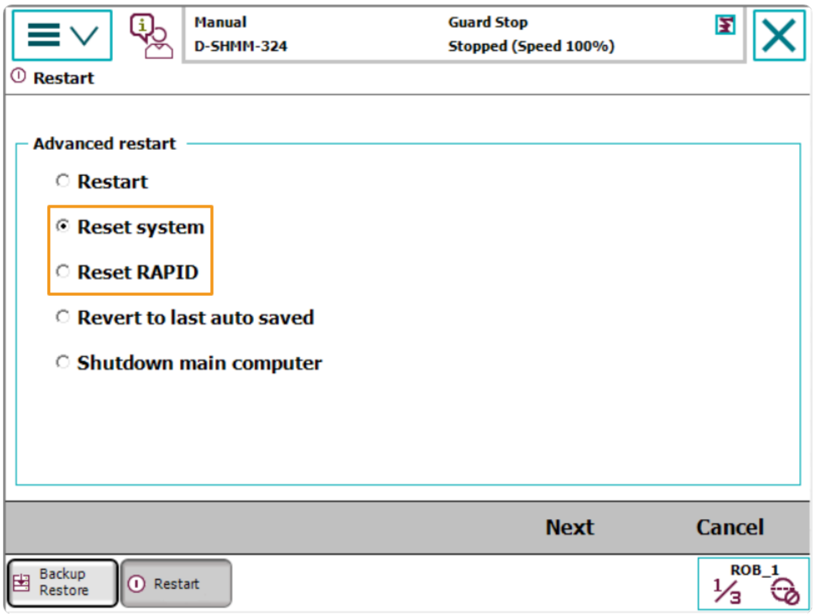
Manual
D-SHMM-324
Guard Stop
Stopped (Speed 100%)

Restart

The controller will be restarted. The state is saved and any changed system parameter settings will be activated after the restart.

This operation cannot be undone.

3. Select **Reset system** and then press [**Next**].



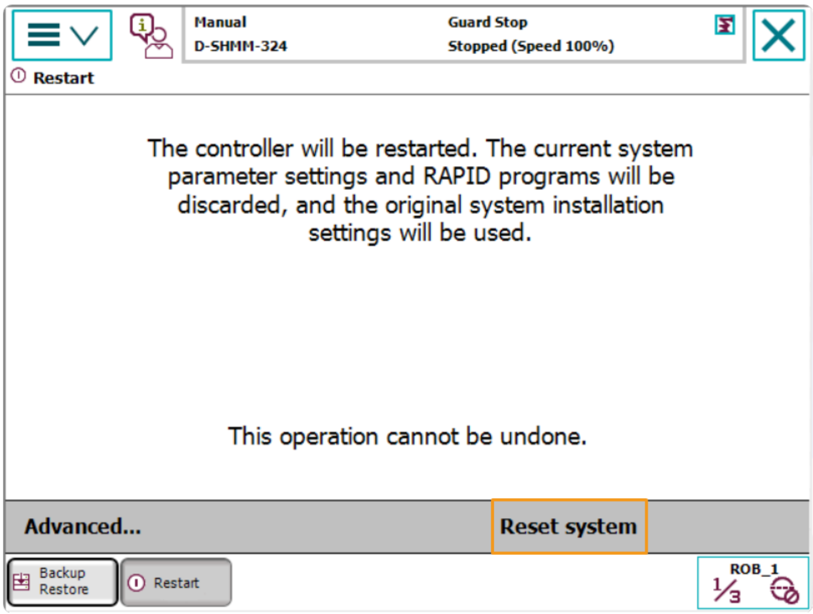
Manual
D-SHMM-324
Guard Stop
Stopped (Speed 100%)

Restart

Advanced restart

○ Restart
● Reset system
○ Reset RAPID
○ Revert to last auto saved
○ Shutdown main computer

4. Press [**Reset system**].



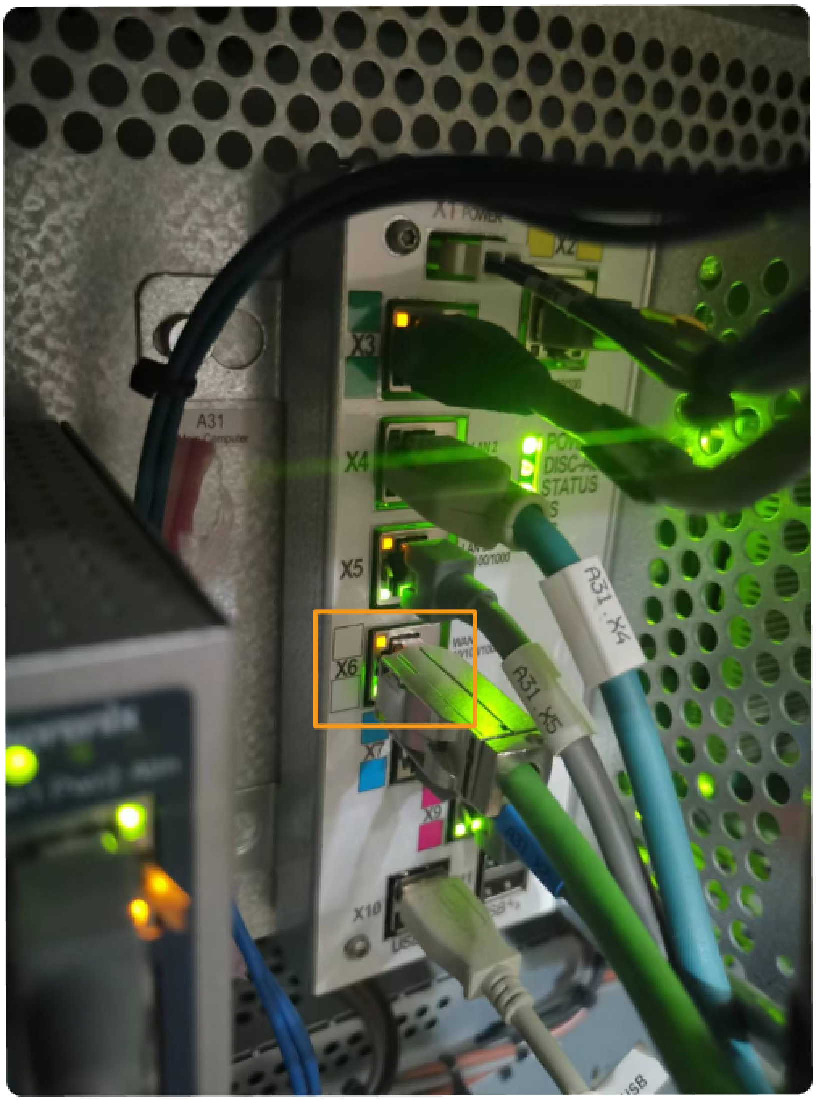
The controller will be restarted. The current system parameter settings and RAPID programs will be discarded, and the original system installation settings will be used.

This operation cannot be undone.

Resetting the system takes 1 to 2 minutes. Resetting is completed when the home page is displayed again on the teach pendant.

Connect the Network

1. Connect the IPC Ethernet Cable to the X6 (WAN) port on the robot controller, as shown below.



2. Make sure that the IP address of the ABB robot and that of the IPC are in the same subnet.

Prepare the Program Files

1. On the IPC, open the `Mech-Center/Robot_Server/Robot_FullControl/abb/server on ABB` folder in the installation directory of the Mech-Vision and Mech-Viz software.
2. Copy this folder to the USB flash drive, and insert it into the computer installing the RobotStudio software.

> RobotStudio is the simulation and offline programming software for ABB robots. This software can be installed on the IPC or another computer. In this example, RobotStudio is installed on another computer.

File description:

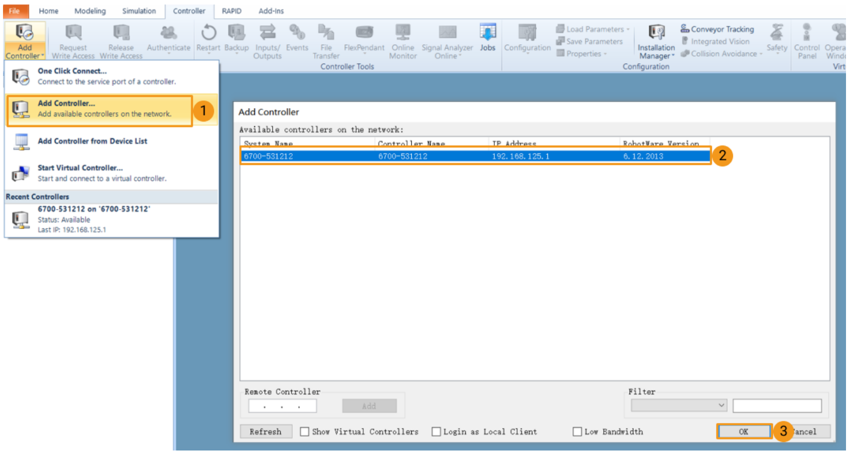
- "MM" folder: includes the robot program modules.
- "config": robot configuration files.
  - If the D652 IO board is used on site, you should use the D652.cfg and SYS.cfg files.
  - If the DSQC1030 IO board is used on site, you should use the DSQC1030.cfg and SYS.cfg files.

- If neither the D652 IO board nor the DSQC1030 IO board is used on site, you should use the EIO.cfg and SYS.cfg files.

Load the Program Files to the Robot

Open RobotStudio and Connect to the Controller

Click the **Controller** menu of RobotStudio, and select Add Controller › Add Controller on the toolbar. In the prompted **Add Controller** dialog box, select the controller, and click the **[OK]** button.

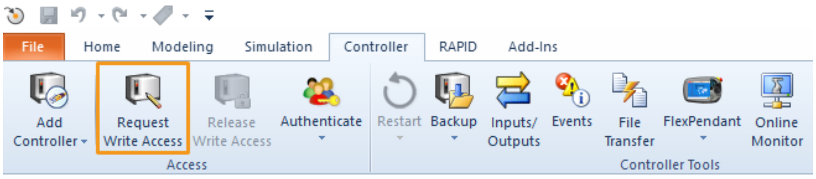


Available controllers on the network:

System Name	Controller Name	IP Address	RobotWare Version
6700-531212	6700-531212	192.168.125.1	6.12.2013

Obtain Write Access to the Robot

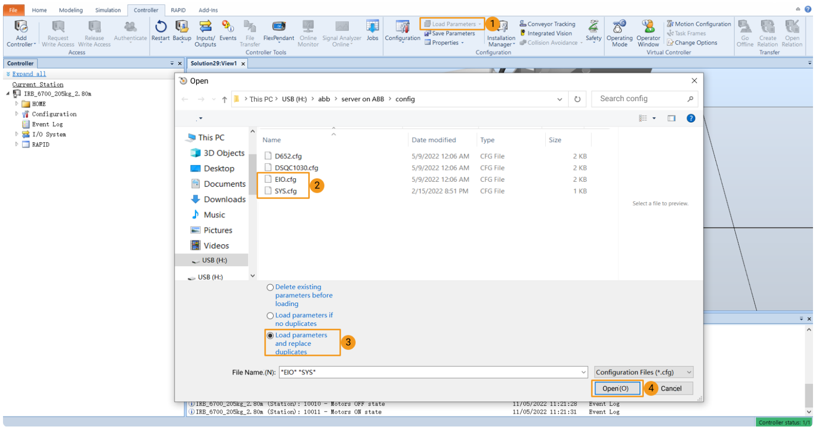
1. On the toolbar, click **Request Write Access** to request the write access to the teach pendant.



2. In the prompted **Request for Write Access** dialog box on the teach pendant, tap **[Grant]**.

Load the Robot Configuration Files

1. Click the **Controller** menu of RobotStudio, and select Load Parameters on the toolbar. Select the configuration files to import from the USB flash drive, select the **Load parameters and replace duplicates** radio button, and then click the **[Open]** button.



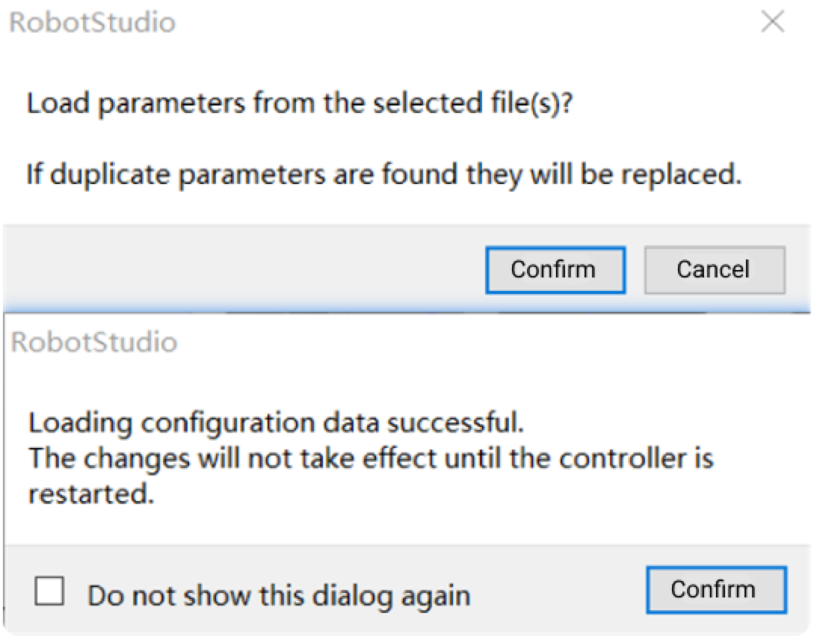
Name	Date modified	Type	Size
D652.cfg	5/9/2022 12:06 AM	CFG File	2 KB
DSQC1030.cfg	5/9/2022 12:06 AM	CFG File	2 KB
EIO.cfg	5/9/2022 12:06 AM	CFG File	2 KB
SYS.cfg	2/15/2022 8:51 PM	CFG File	1 KB

○ Delete existing parameters before loading
○ Load parameters if no duplicates
● Load parameters and replace duplicates

File Name.(N): "EIO" "SYS"   Configuration Files (*.cfg)

```
IRB_6700_205kg_2.80m (Station): 10010 - Motors OFF state    11/05/2022 11:21:28    Event Log
IRB_6700_205kg_2.80m (Station): 10011 - Motors ON state     11/05/2022 11:21:31    Event Log
```

2. In the prompted dialog boxes, click [Confirm].



RobotStudio

Load parameters from the selected file(s)?

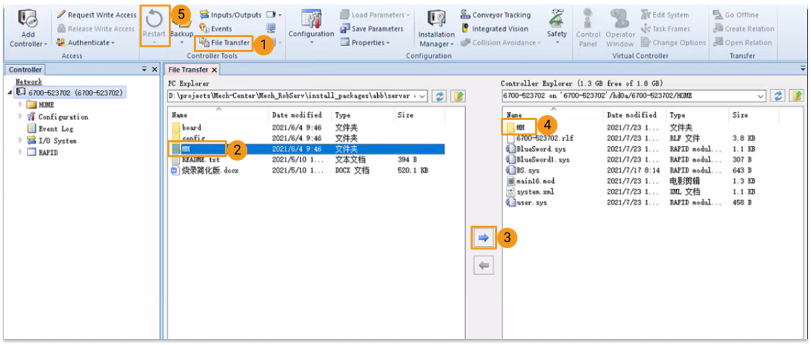
If duplicate parameters are found they will be replaced.

RobotStudio

Loading configuration data successful.
The changes will not take effect until the controller is restarted.

Load the Robot Program Modules

1. Click the **Controller** menu of RobotStudio, and select **File Transfer** on the toolbar. On the left panel of the **File Transfer** interface, select the “MM” folder, and then click the **Transfer** button to transfer this folder to the HOME directory of the robot system.



PC Explorer

D:\projects\Mech-Center\Mech_RobServ\install_packages\abb\server

Name	Date modified	Type	Size
board	2021/6/4 9:46	文件夹	
config	2021/6/4 9:46	文件夹	
MM	2021/6/4 9:46	文件夹	
README.txt	2021/5/10 1...	文本文档	394 B
烧录简化版.docx	2021/5/10 1...	DOCX 文档	520.1 KB

Controller Explorer (1.3 GB free of 1.8 GB)

6700-523702 on '6700-523702' /hd0a/6700-523702/HOME

Name	Date modified	Type	Size
MM	2021/7/23 1...	文件夹	
6700-523702.rlf	2021/7/23 1...	RLF 文件	3.8 KB
BlueSword.sys	2021/7/23 1...	RAPID modul...	1.1 KB
BlueSword1.sys	2021/7/23 1...	RAPID modul...	307 B
BS.sys	2021/7/17 8:14	RAPID modul...	643 B
main16.mod	2021/7/23 1...	电影剪辑	1.3 KB
system.xml	2021/7/23 1...	XML 文档	1.1 KB
user.sys	2021/7/23 1...	RAPID modul...	458 B

2. Click the **Controller** menu of RobotStudio, and select **Restart** on the toolbar to restart the robot system.

Till now, you have loaded the master-control program and the configuration files to the robot.

Test Whether Master-Control Communication Can Be Established

After the robot system has been rebooted, perform the following steps to test whether the master-control communication can be established with the robot:

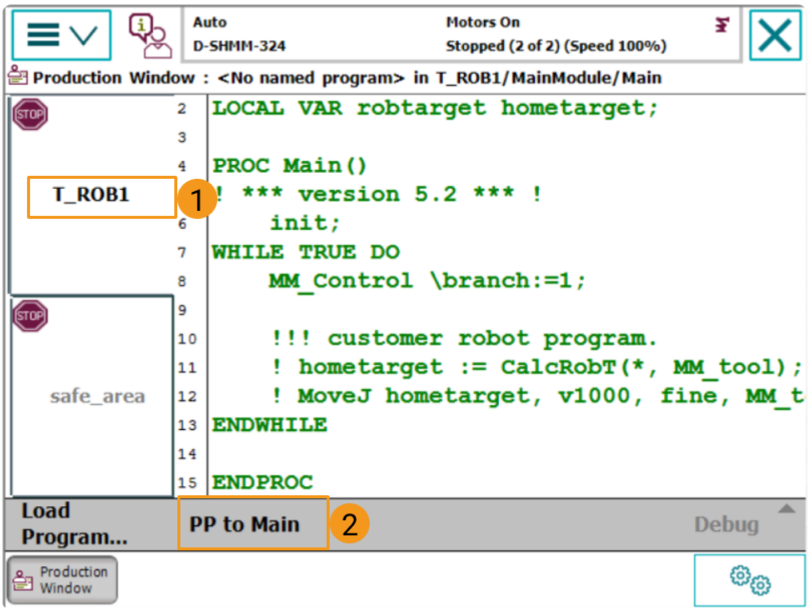
Switch the Robot to Automatic Mode

1. In the robot controller, switch the robot to the automatic mode by using the switch key.
2. In the prompted dialog box on the teach pendant, tap [Confirm].
3. In the robot controller, press the motor power button to power up the robot. When the robot is

powered on, this button is solid on.

Run the Main Program

1. On the teach pendant, move the PP of tasks T_ROB1 to main, and tap [PP to Main].

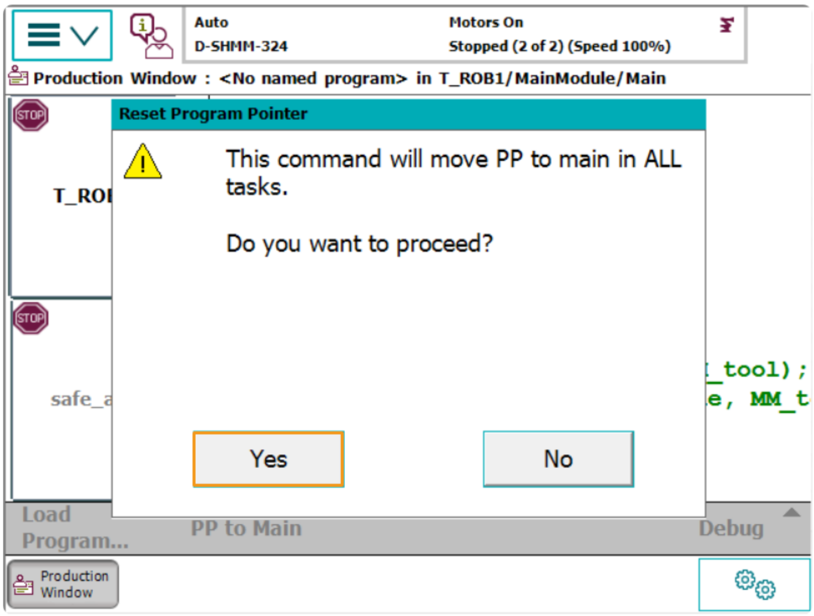


Auto
D-SHMM-324
Motors On
Stopped (2 of 2) (Speed 100%)

Production Window : <No named program> in T_ROB1/MainModule/Main

```
LOCAL VAR robtarget hometarget;

PROC Main()
! *** version 5.2 *** !
    init;
WHILE TRUE DO
    MM_Control \branch:=1;

    !!! customer robot program.
    ! hometarget := CalcRobT(*, MM_tool);
    ! MoveJ hometarget, v1000, fine, MM_t
ENDWHILE

ENDPROC
```

2. In the prompted dialog box, click [Yes].



Auto
D-SHMM-324
Motors On
Stopped (2 of 2) (Speed 100%)

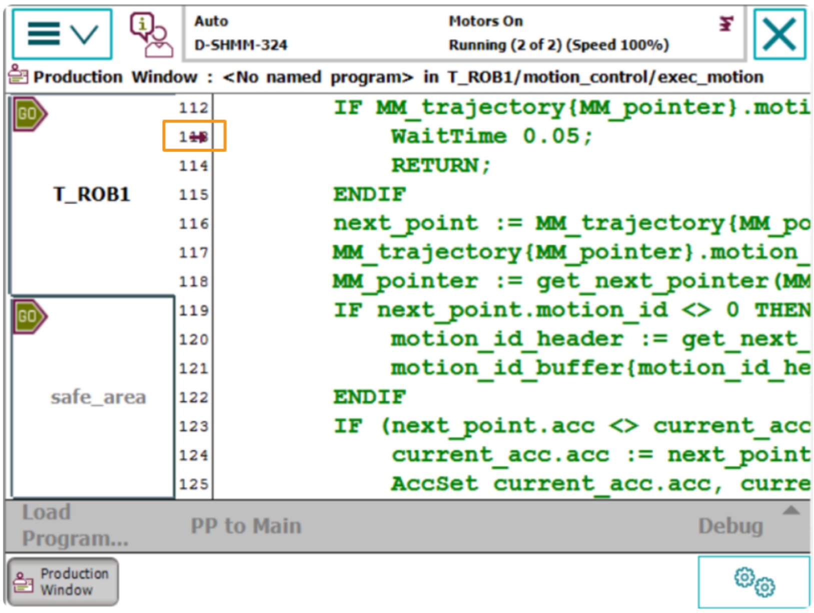
Production Window : <No named program> in T_ROB1/MainModule/Main

Reset Program Pointer

This command will move PP to main in ALL tasks.

Do you want to proceed?

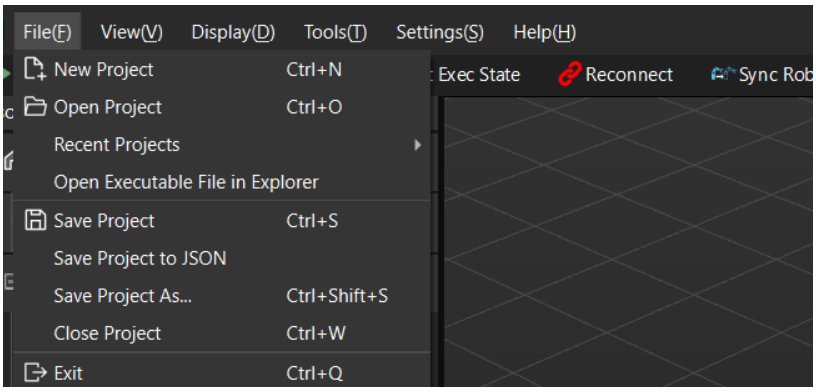
3. Press the run button on the right of the teach pendant.


Auto
D-SHMM-324
Motors On
Running (2 of 2) (Speed 100%)
Production Window : <No named program> in T_ROB1/motion_control/exec_motion
T_ROB1
safe_area
Load Program...
PP to Main
Debug
Production Window

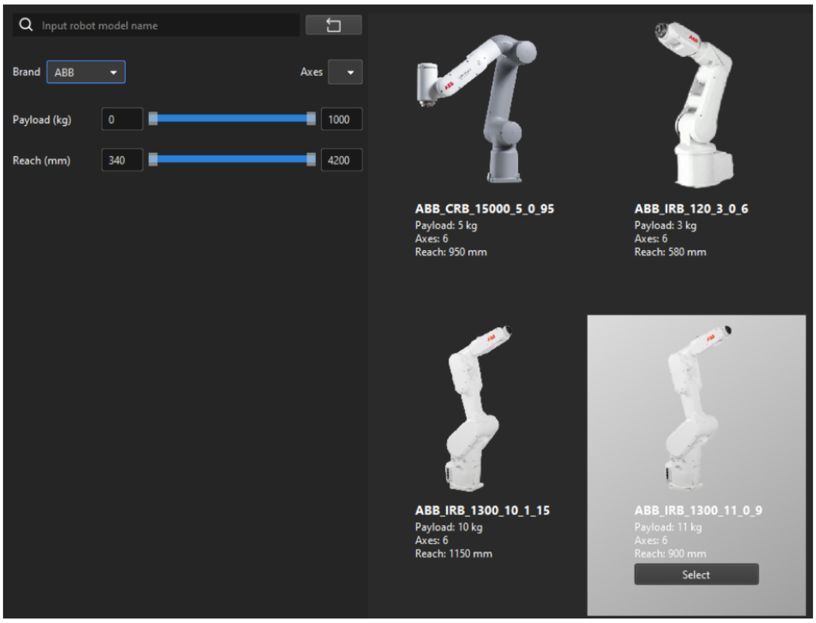
```
IF MM_trajectory{MM_pointer}.moti
    WaitTime 0.05;
    RETURN;
ENDIF
next_point := MM_trajectory{MM_po
MM_trajectory{MM_pointer}.motion_
MM_pointer := get_next_pointer(MM
IF next_point.motion_id <> 0 THEN
    motion_id_header := get_next_
    motion_id_buffer{motion_id_he
ENDIF
IF (next_point.acc <> current_acc
    current_acc.acc := next_point
    AccSet current_acc.acc, curre
```

Create a Mech-Viz Project

1. Open Mech-Viz, select File › New Project.



2. Set **Brand** to “ABB”, select the robot model “ABB_IRB_1300_11_0_9” on the right panel, and then click **[Select]**.



Brand: ABB
Axes:
Payload (kg): 0 – 1000
Reach (mm): 340 – 4200

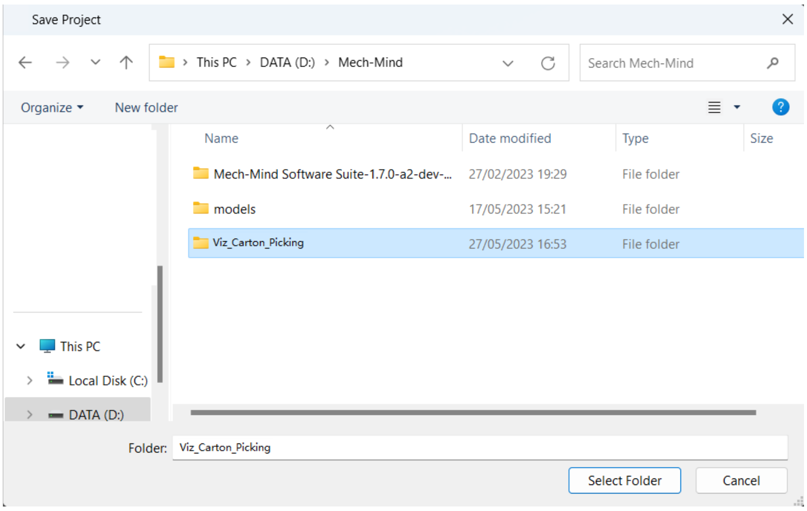
ABB_CRB_15000_5_0_95
Payload: 5 kg
Axes: 6
Reach: 950 mm

ABB_IRB_120_3_0_6
Payload: 3 kg
Axes: 6
Reach: 580 mm

ABB_IRB_1300_10_1_15
Payload: 10 kg
Axes: 6
Reach: 1150 mm

ABB_IRB_1300_11_0_9
Payload: 11 kg
Axes: 6
Reach: 900 mm

3. Press `Ctrl` + `S`, create a new folder named "Viz_Carton_Picking", select it, and then click [**Select folder**].

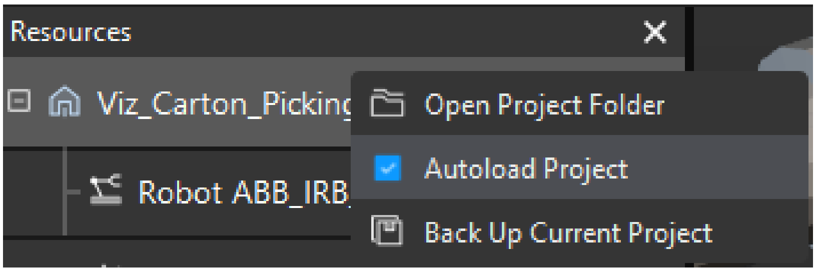


Name	Date modified	Type	Size
Mech-Mind Software Suite-1.7.0-a2-dev-...	27/02/2023 19:29	File folder	
models	17/05/2023 15:21	File folder	
Viz_Carton_Picking	27/05/2023 16:53	File folder	

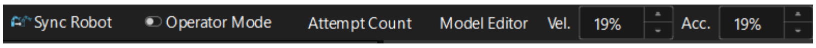
Folder: Viz_Carton_Picking

After the Mech-Viz project is saved successfully, in the **Resources** panel, the project name is displayed as "Viz_Carton_Picking".

4. Right-click the project name, and select **Autoload project** checkbox.



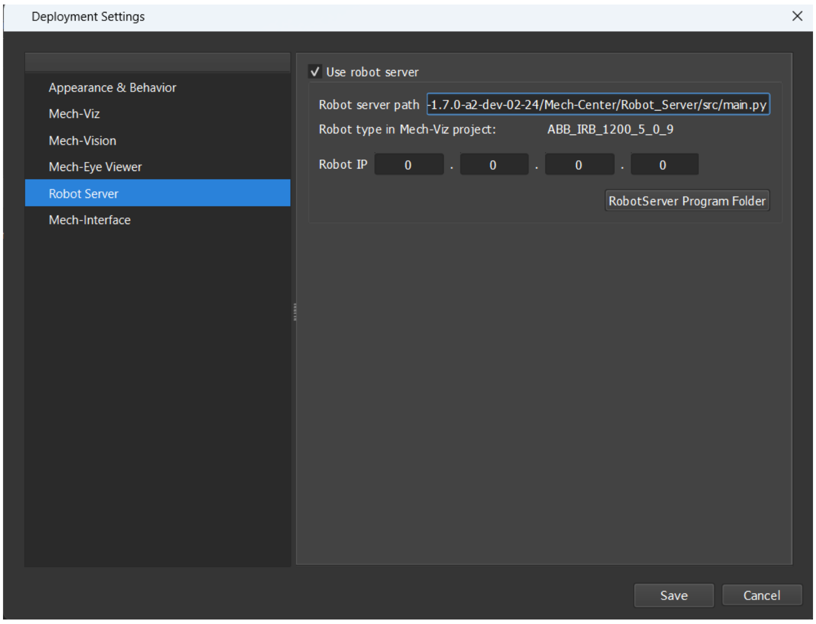
5. For the robot motion safety, on the toolbar, set **Vel.** (velocity) and **Acc.** (acceleration) to a small value, such as 5%.



6. Press `Ctrl` + `S` to save the project.

Configure Master-Control Communication Settings on Mech-Center

1. Open Mech-Center, select Deployment Settings › Robot Server, and then select **Enable Robot Server** checkbox.
2. Make sure that the robot model matches the real robot, set **Robot IP** to the IP address of the real robot, and then click **[Save]**.



☑ Use robot server

Robot server path -1.7.0-a2-dev-02-24/Mech-Center/Robot_Server/src/main.py

Robot type in Mech-Viz project: ABB_IRB_1200_5_0_9

Robot IP 0 . 0 . 0 . 0

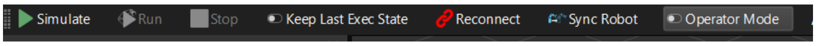
Connect to the Robot

In Mech-Center, click **Connect Robot** on the toolbar.

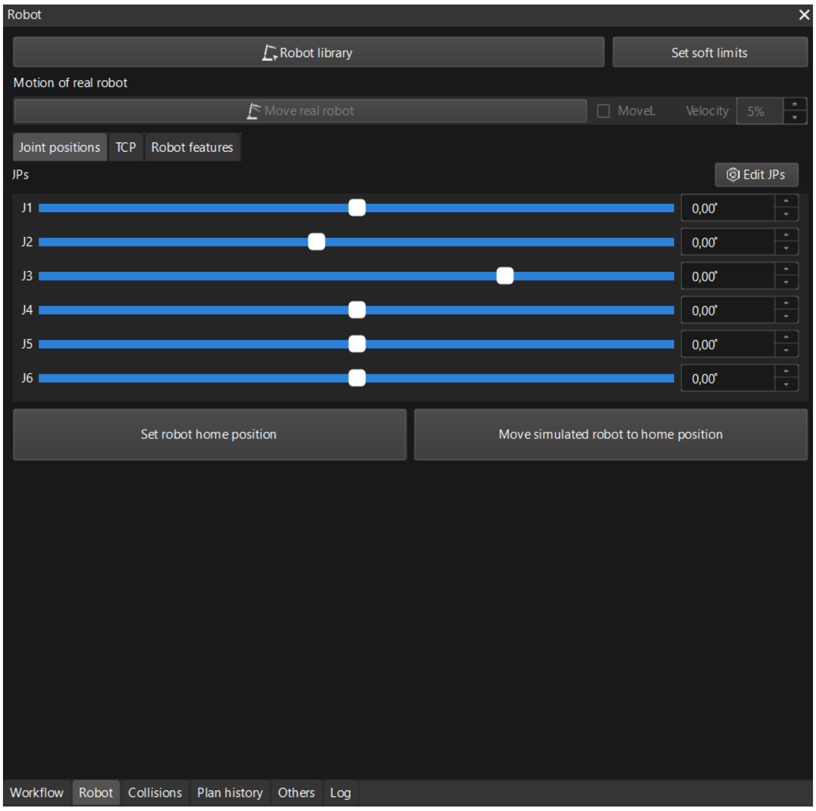
- If the robot is successfully connected, a robot icon with the robot model will be displayed in the Service Status bar, and the corresponding message is printed in the Log panel.
- If the connection fails, please check if any of the operations so far was incorrect.

Move the Robot

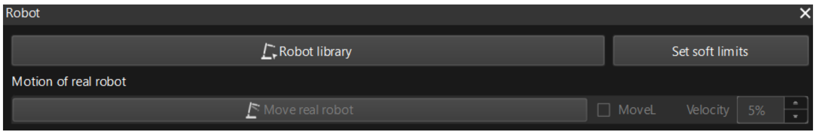
1. In Mech-Viz, click **Sync Robot** in the toolbar to synchronize the pose of the simulated robot to that of the real robot. Then, click **Sync Robot** again to disable the synchronization.



2. On the **Robot** tab, change the joint position of J1 slightly (for example, from 0° to 3°). The simulated robot will move accordingly.



3. Click [**Move real robot**].



When moving the robot, please ensure the safety of personnel. In the case of an emergency, press the emergency stop button on the teach pendant!

If you observe that the real robot moves to the pose of the simulated robot, the master-control communication has been established successfully.

4.3. Hand-Eye Calibration

In this tutorial, you will perform the automatic hand-eye calibration in the eye-to-hand setup.

The hand-eye calibration establishes **the transformation relationship between the camera and robot reference frames (that is camera extrinsic parameters)**. With this relationship, the object pose determined by the vision system can be transformed into that in the robot reference frame, which guides the robot in performing its tasks.

▶ https://www.youtube.com/watch?v=u4m1Jsklfk8/PLVcMd7cW2rXWbHhTQX8m2R1r6PhX6vrCI *(YouTube video)*

Video Tutorial: Hand-Eye Calibration (Master-Control)

Preparation before Calibration

In this section, you will mount the calibration board, adjust the camera parameters, and complete the pre-calibration configuration.

Mount the Calibration Board

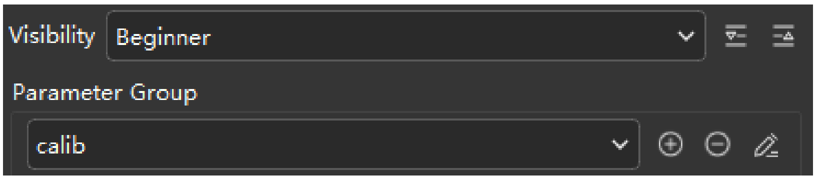
In the eye-to-hand setup, the calibration board needs to be mounted to the robot flange adapter.

Follow these steps:

1. Take the calibration board and flange adapter out of the system delivery package.
2. Use screws, gaskets, and nuts to fasten the flange adapter to the flange adapter.
3. Use screws, gaskets, and nuts to secure the calibration board to the flange adapter.
4. After mounting the calibration board, move the robot to the top of the lowest workpiece in the work area and into the camera's field of view (FOV).

Adjust Camera Parameters

1. In Mech-Eye Viewer, connect to the camera, and then set **Parameter Group** to "calib".

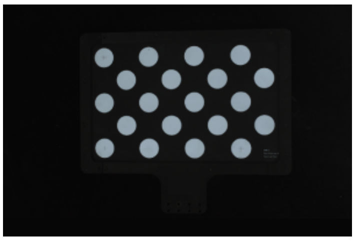
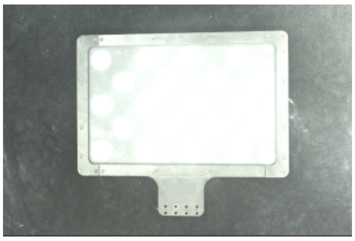
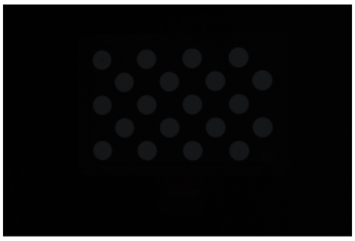
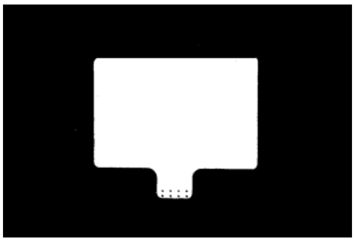
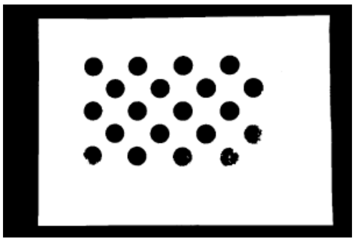
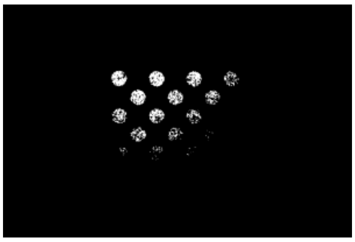


Visibility Beginner

Parameter Group

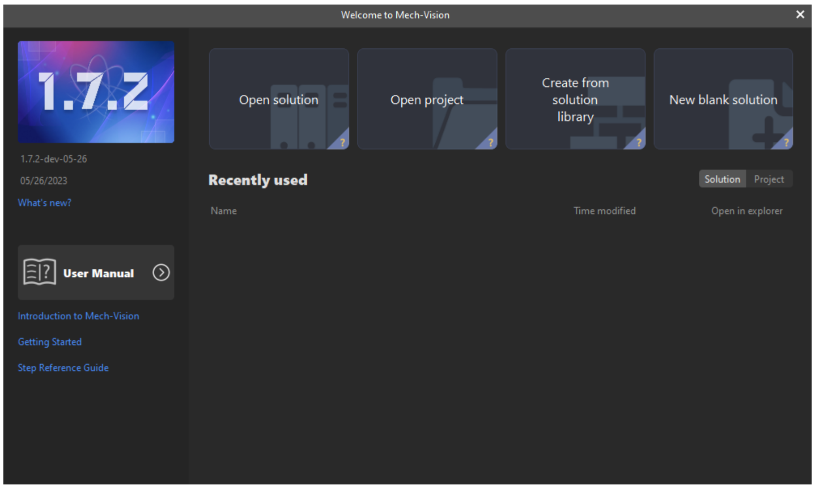
calib

2. Adjust the 2D parameters to make sure that the 2D image of the calibration board is clear and neither overexposed nor underexposed.
3. Adjust 3D parameters to make sure that the obtained point clouds of the circles on the calibration board are complete and have clear contours. It is recommended to set the **Surface Smoothing** and **Outlier Removal** parameters of **Point Cloud Processing** to **Normal** to reduce point cloud fluctuation.

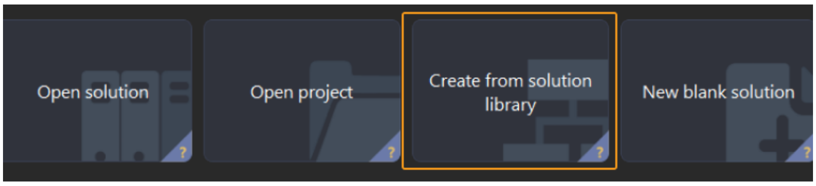
	Normal	Overexposed	Underexposed
2D image			
Point cloud			

Create a Mech-Vision Project and Save It

1. Open Mech-Vision. When the welcome interface as shown below is displayed, it indicates that Mech-Vision is started successfully.

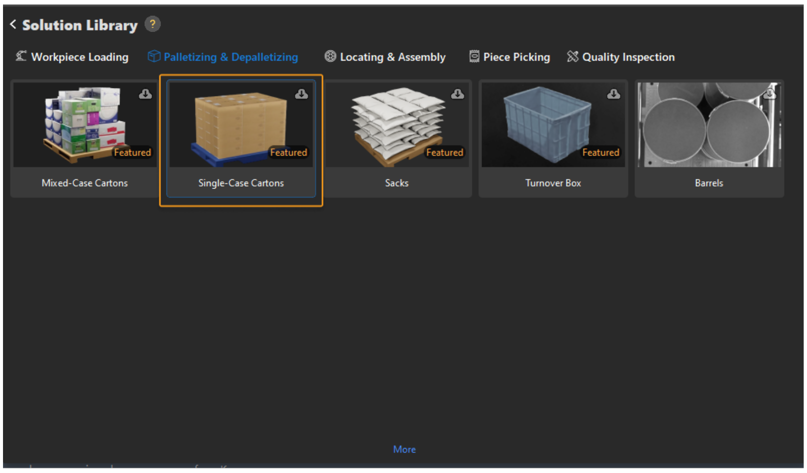


2. In the Welcome interface of Mech-Vision, click [**Create from solution library**] to open the Solution Library.



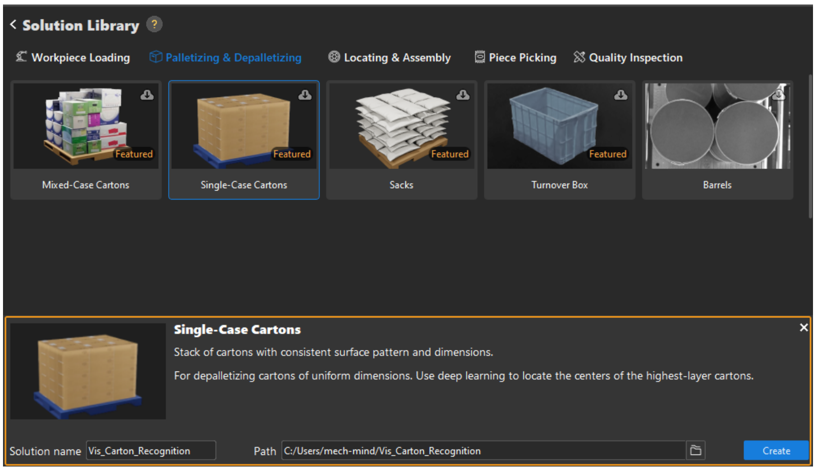
Solution Library is a resource library that provides typical solutions or projects (with example data) coming from different application scenarios.

3. Select the **Single-Case Cartons** project from the Solution Library.



If you cannot find this project in the Solution Library, click [**More**] at the bottom of the Solution Library interface.

4. After this project is selected, the information introducing this project will be displayed on the lower part of the Solution Library interface. Set the solution name and path, and then click [**Create**].



Single-Case Cartons

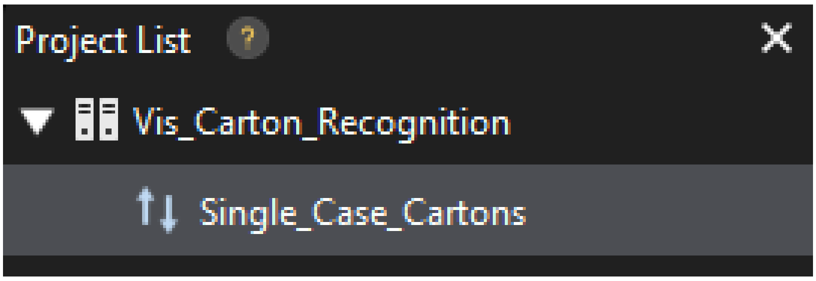
Stack of cartons with consistent surface pattern and dimensions.

For depalletizing cartons of uniform dimensions. Use deep learning to locate the centers of the highest-layer cartons.

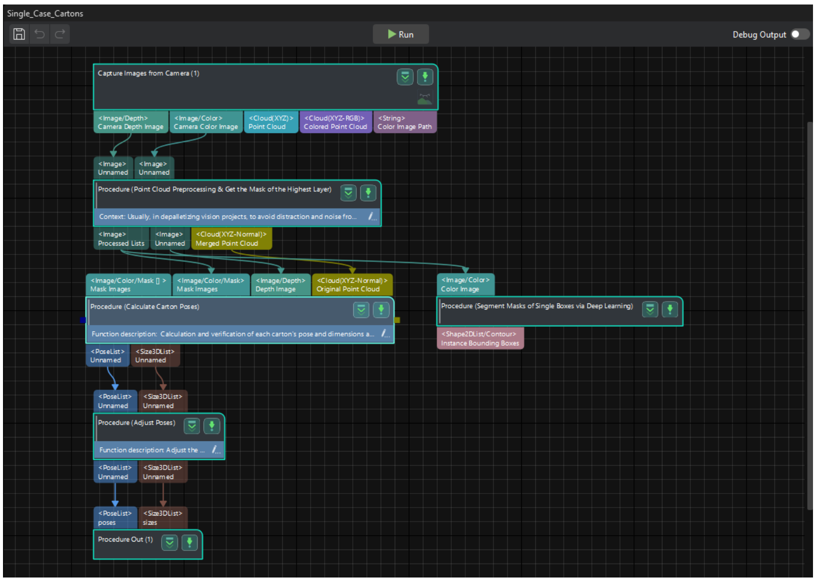
Solution name Vis_Carton_Recognition Path C:/Users/mech-mind/Vis_Carton_Recognition

After the project is created, the created solution and project will be displayed in the project list in the upper-left corner of the Mech-Vision main interface.

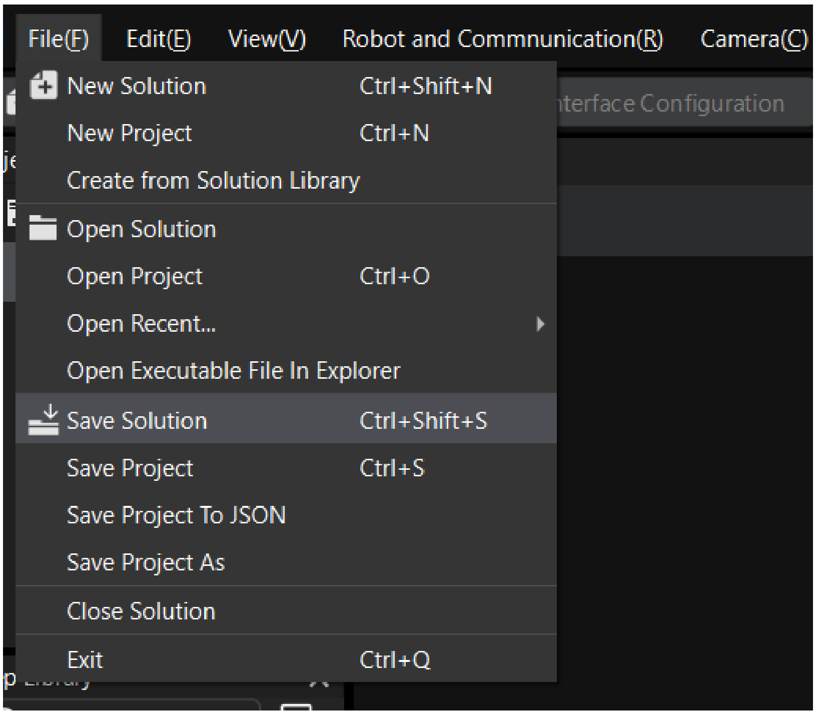
- A **solution** is a set of configurations and data related to robots and robot communication, vision processing, path planning, etc. that are required for the machine vision application.
- A **project** is a workflow of vision processing in the solution. Normally, a solution only contains one project, but it may contain more than one project in complex application scenarios. In this tutorial, the solution only needs one project.



In the Graphical Programming Workspace of the main interface, the workflow of the “Single-Case Cartons” project will be displayed.

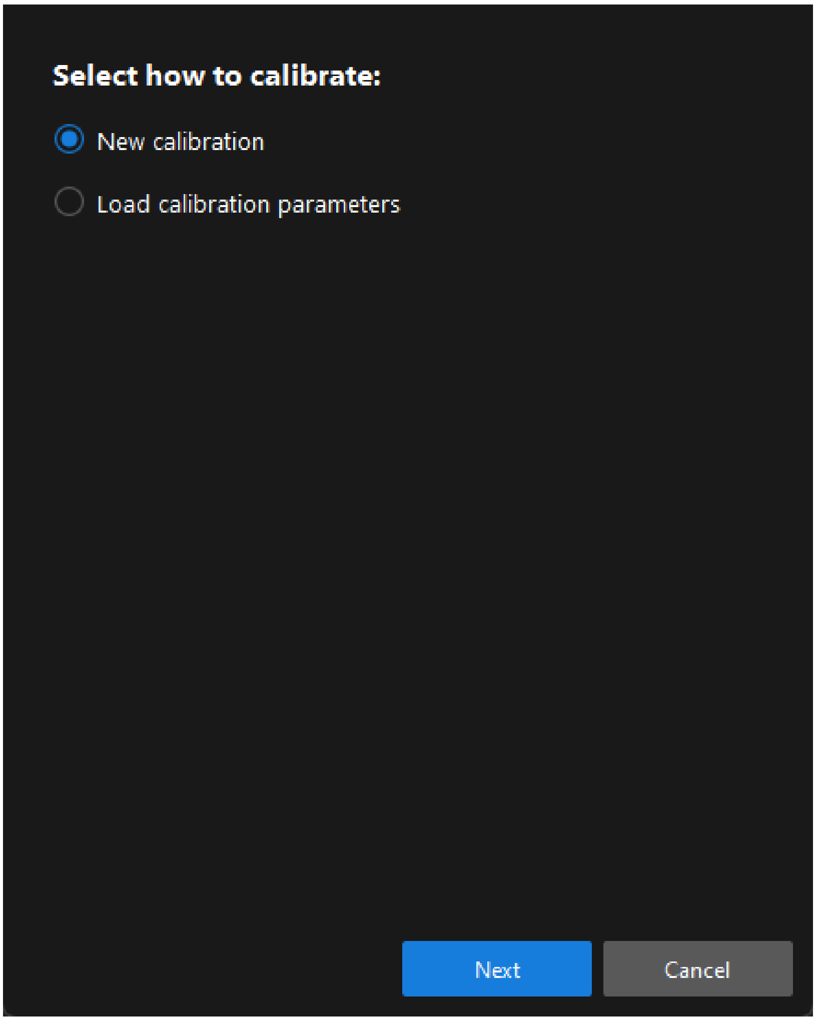

Single_Case_Cartons
Run
Debug Output
Capture Images from Camera (1)
<Image/Depth> Camera Depth Image
<Image/Color> Camera Color Image
<Cloud(XYZ)> Point Cloud
<Cloud(XYZ-RGB)> Colored Point Cloud
<String> Color Image Path
<Image> Unnamed
<Image> Unnamed
Procedure (Point Cloud Preprocessing & Get the Mask of the Highest Layer)
Context: Usually, in depalletizing vision projects, to avoid distraction and noise fro...
<Image> Processed Lists
<Image> Unnamed
<Cloud(XYZ-Normal)> Merged Point Cloud
<Image/Color/Mask [] > Mask Images
<Image/Color/Mask> Mask Images
<Image/Depth> Depth Image
<Cloud(XYZ-Normal)> Original Point Cloud
<Image/Color> Color Image
Procedure (Calculate Carton Poses)
Function description: Calculation and verification of each carton's pose and dimensions a...
Procedure (Segment Masks of Single Boxes via Deep Learning)
<Shape2DList/Contour> Instance Bounding Boxes
<PoseList> Unnamed
<Size3DList> Unnamed
<PoseList> Unnamed
<Size3DList> Unnamed
Procedure (Adjust Poses)
Function description: Adjust the ...
<PoseList> Unnamed
<Size3DList> Unnamed
<PoseList> poses
<Size3DList> sizes
Procedure Out (1)

5. On the menu bar, select File › Save Solution.



Pre-calibration Configuration

1. In Mech-Vision, click the [**Camera Calibration (Standard)**] button on the toolbar. The **Configuration before Calibration** window will be prompted.
2. In the **Select how to calibrate** window, select the **New calibration** radio button, and then click the [**Next**] button.

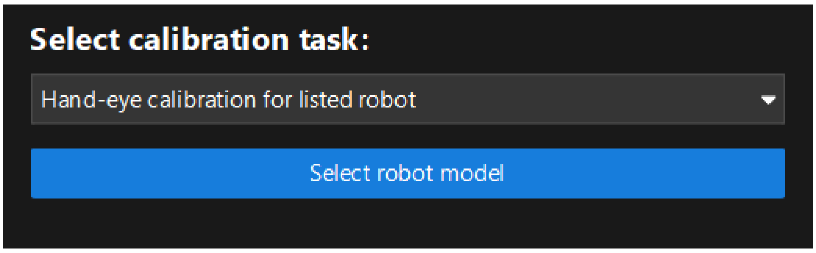


Select how to calibrate:

● New calibration

○ Load calibration parameters

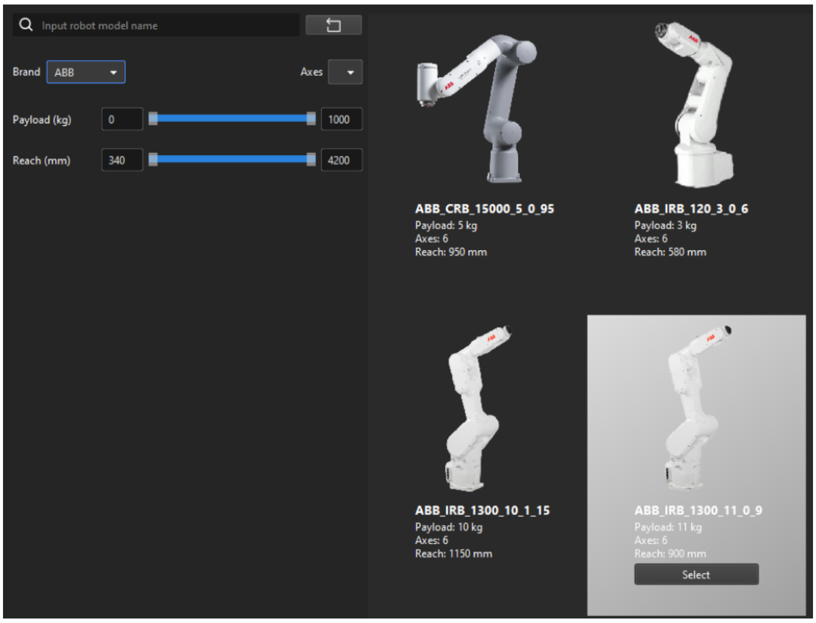
3. In the **Select calibration task** window, select **Hand-eye calibration for listed robot** from the drop-down list box, and click the **[Select robot model]** button.



Select calibration task:

Hand-eye calibration for listed robot

4. Click the **Brand** dropdown box, and select "ABB". Then, at the right panel, select model "ABB_IRB_1300_11_0_9", click the **[Select]** button, and then click the **[Next]** button.



Input robot model name

Brand: ABB

Axes:

Payload (kg): 0 – 1000

Reach (mm): 340 – 4200

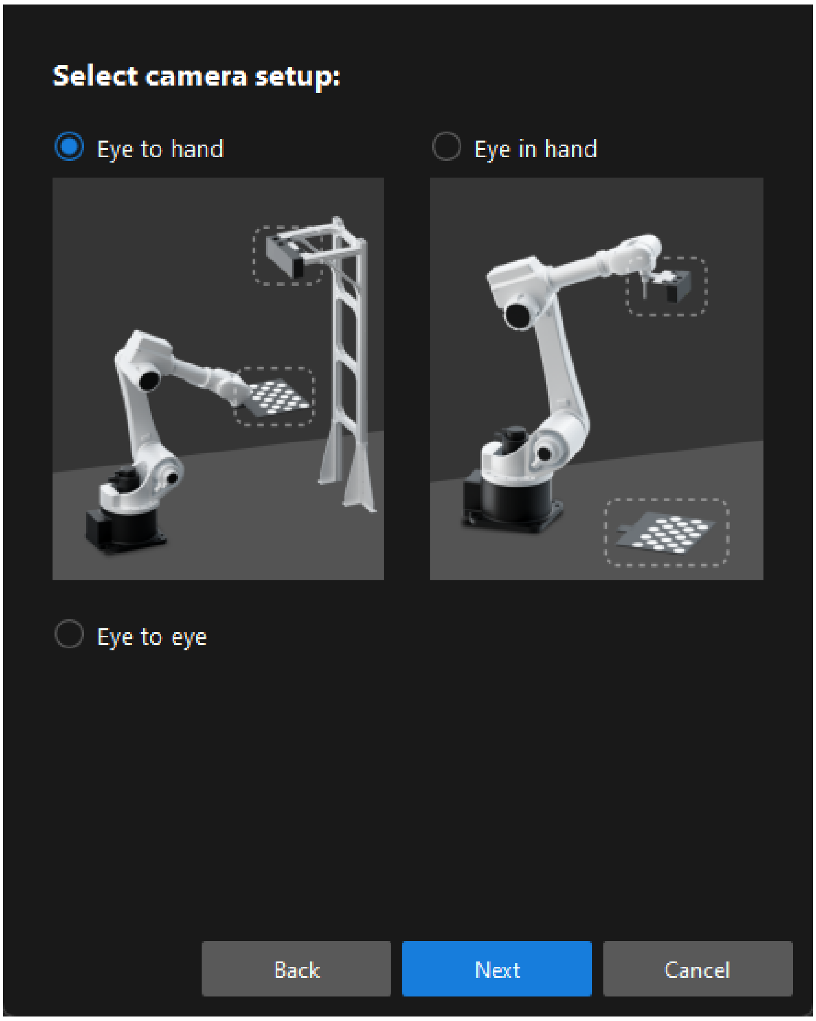
ABB_CRB_15000_5_0_95
Payload: 5 kg
Axes: 6
Reach: 950 mm

ABB_IRB_120_3_0_6
Payload: 3 kg
Axes: 6
Reach: 580 mm

ABB_IRB_1300_10_1_15
Payload: 10 kg
Axes: 6
Reach: 1150 mm

ABB_IRB_1300_11_0_9
Payload: 11 kg
Axes: 6
Reach: 900 mm

5. In the **Select camera setup** window, select the **Eye to hand** radio button, and then click the **[Next]** button.

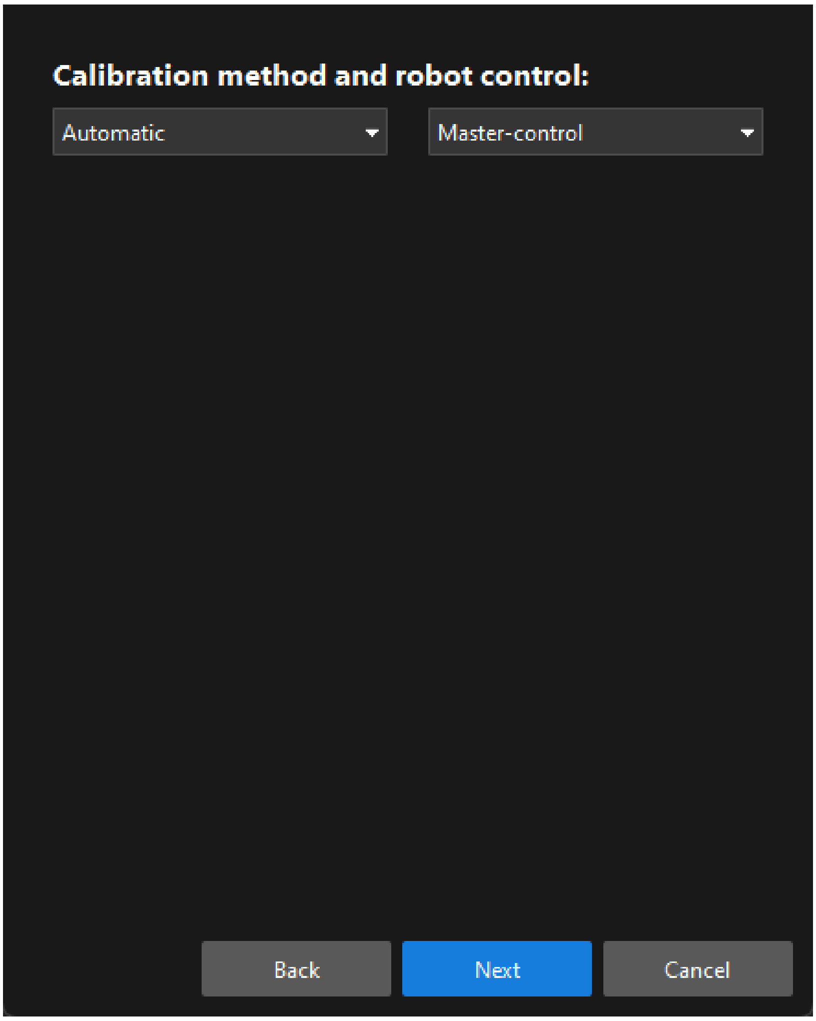


Select camera setup:

● Eye to hand ○ Eye in hand

○ Eye to eye

6. In the **Calibration method and robot control** window, select **Automatic** and **Master-control**, and then click the **[Next]** button.

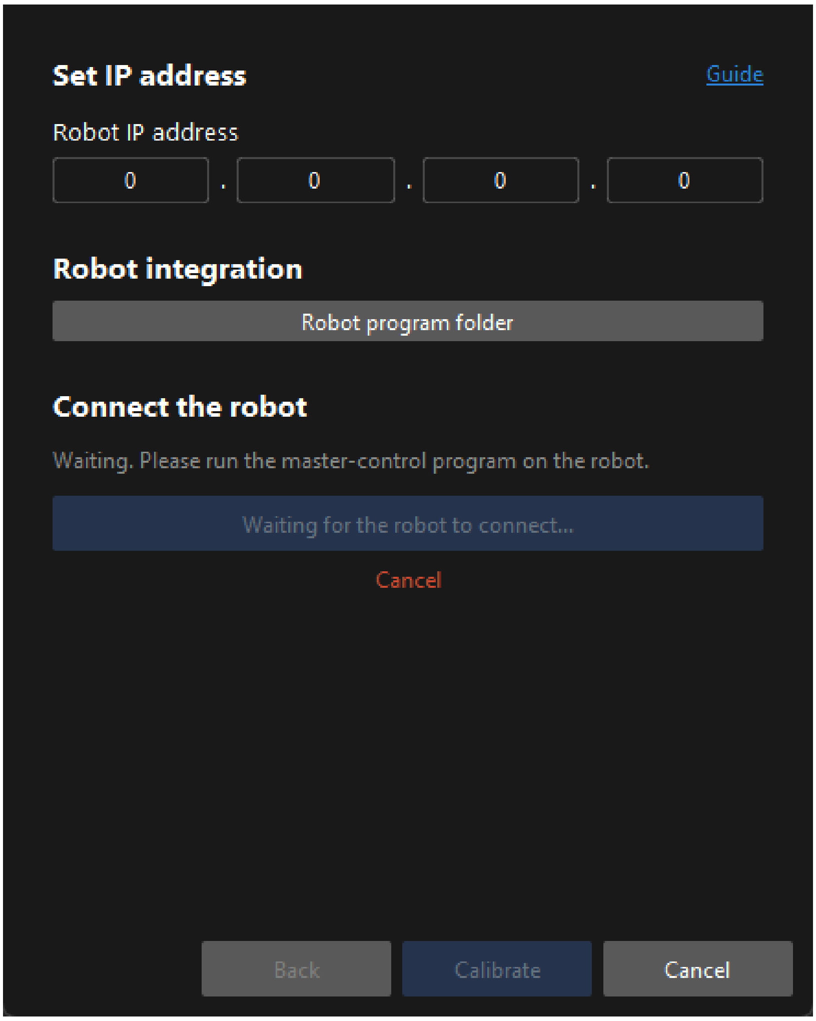


Calibration method and robot control:

Automatic

Master-control

7. In the **Communication Settings** window, set the **Robot IP address** parameter to the real robot's IP address.

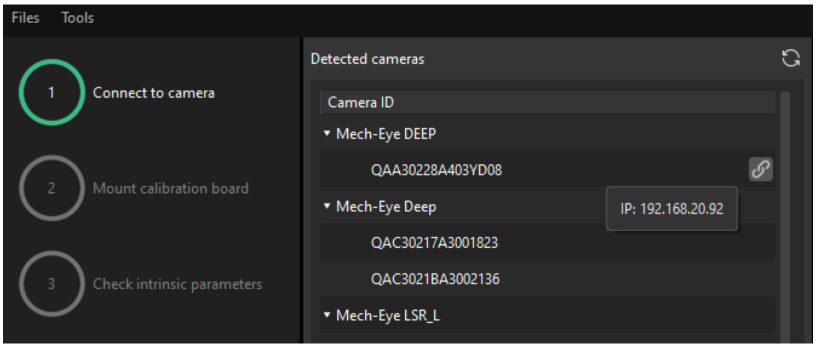


8. On the robot teach pendant, confirm that the MM main program has been started.
9. Return to Mech-Vision, and click the [**Connect the robot**] button in the **Connect the robot** area. The button will turn into **Waiting for the robot to connect...**
10. Wait until the "Connected" status message is displayed in the **Connect the robot** area, and then click the [**Calibrate**] button. The **Calibration (Eye to Hand)** window will be prompted.

Calibration Procedure

Connect to a Camera

1. In the **Connect to Camera** step, find the camera to connect in the **Detected Cameras** list, and click the 🔗 button.

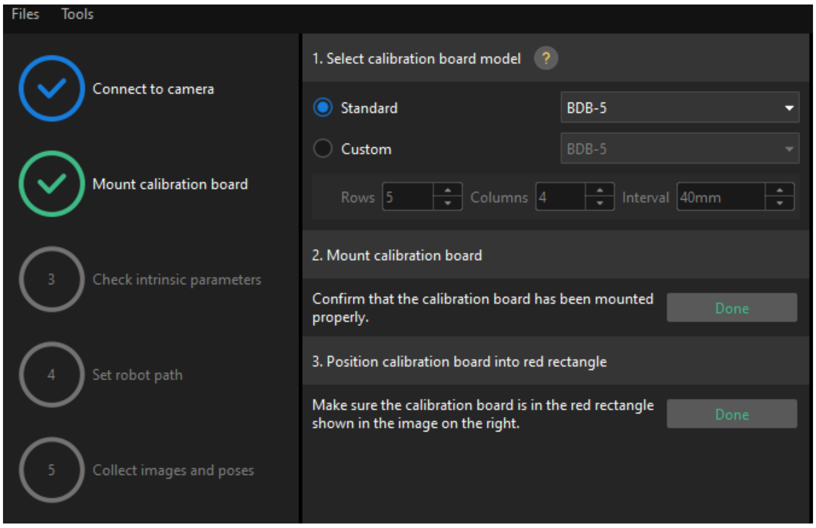


2. After the camera is connected, click the **[Capture once]** or **[Capture live]** button.
3. In the right **Image viewer** panel, ensure that the captured 2D image and depth map meet the calibration requirements and click the **[Next]** button on the bottom bar.

> If the captured image does not meet the calibration requirements, you need to open the Mech-Eye Viewer software to adjust the 2D and 3D exposure parameters and re-capture images.

Mount the Calibration Board

1. In the **Mount calibration board** step, select the **Standard** radio button and then select the calibration board model according to its model nameplate in the **1. Select calibration board type** area.
2. Make sure that the calibration board has been attached to the robot flange securely, and then click the **[Confirm]** button in the **2. Mount calibration board** area.
3. Confirm that the calibration board is in the center of the camera's field of view (the red rectangle), and then click the **[Confirm]** button in the **3. Position calibration board into red rectangle** area.



1. Select calibration board model

● Standard BDB-5

○ Custom BDB-5

Rows 5 Columns 4 Interval 40mm

2. Mount calibration board

Confirm that the calibration board has been mounted properly.

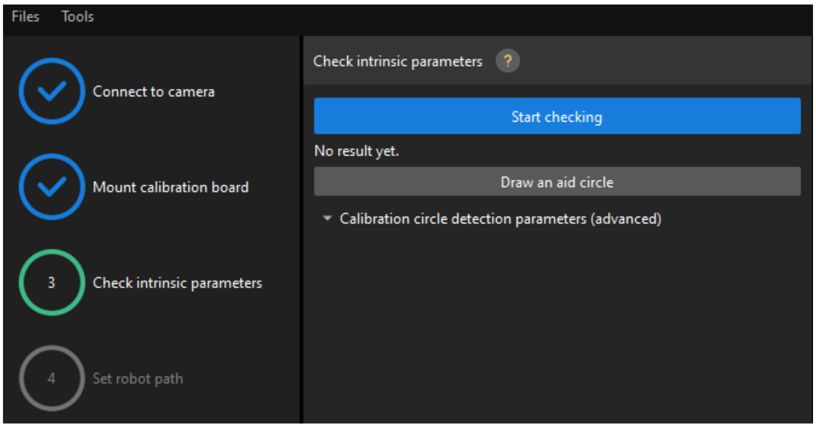
3. Position calibration board into red rectangle

Make sure the calibration board is in the red rectangle shown in the image on the right.

4. After all the operations related to the calibration board are completed, click the **[Next]** button on the bottom bar.

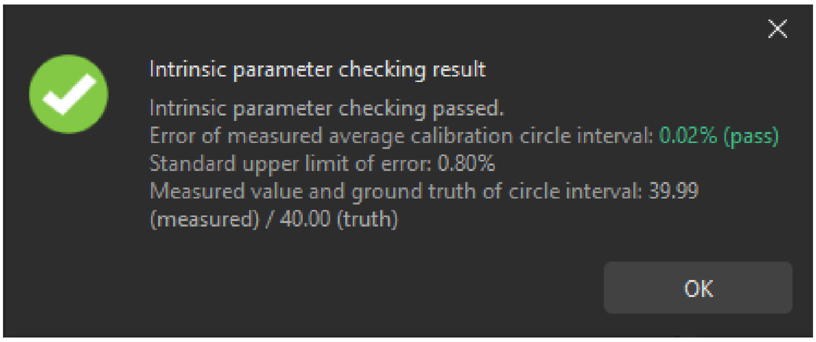
Check Intrinsic Parameters

1. In the **Check intrinsic parameters** step, click the **[Start checking]** button.


Files
Tools
Connect to camera
Mount calibration board
3
Check intrinsic parameters
4
Set robot path
Check intrinsic parameters
Start checking
Draw an aid circle
Calibration circle detection parameters (advanced)

No result yet.

2. Confirm that the camera intrinsic parameters check passes, and then click the [**Next**] button on the bottom bar.



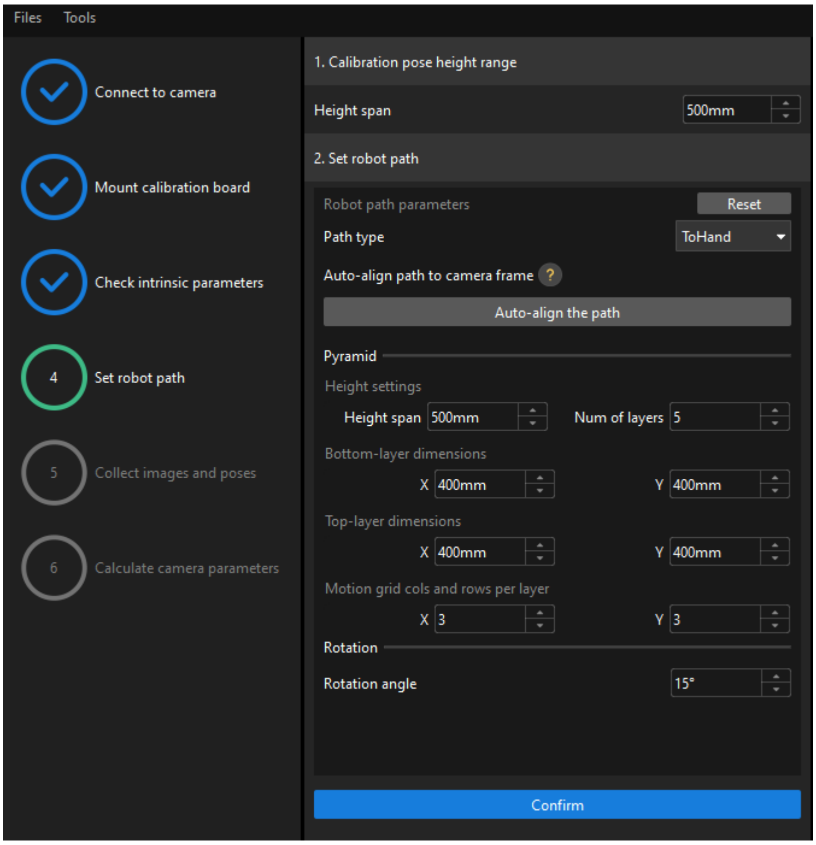
Intrinsic parameter checking result

Intrinsic parameter checking passed.
Error of measured average calibration circle interval: 0.02% (pass)
Standard upper limit of error: 0.80%
Measured value and ground truth of circle interval: 39.99 (measured) / 40.00 (truth)

If the camera intrinsic parameter check fails, adjust detection parameters by drawing aid circles or manually adjusting the calibration circle detection parameters.

Set Robot Path

1. In the **Set robot path** step, specify the **Height span** parameter. Please set this parameter according to the working distance range of the calibration board in the camera height direction.



1. Calibration pose height range

Height span 500mm

2. Set robot path

Robot path parameters

Path type ToHand

Auto-align path to camera frame ?

Pyramid

Height settings

Height span 500mm Num of layers 5

Bottom-layer dimensions

X 400mm Y 400mm

Top-layer dimensions

X 400mm Y 400mm

Motion grid cols and rows per layer

X 3 Y 3

Rotation

Rotation angle 15°

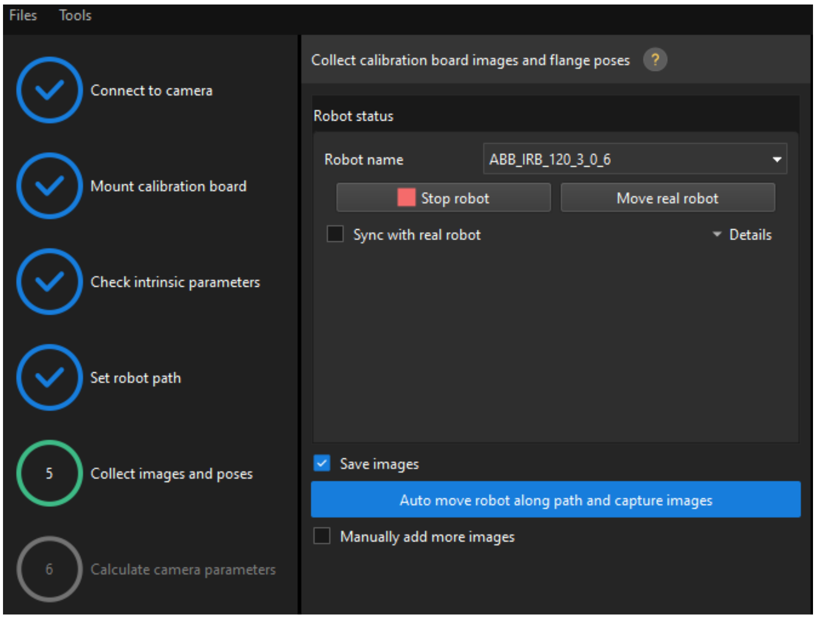
2. Set the **Path type** to **ToHand**, specify the pyramid parameters **Height span**, **Num of layers**, **Bottom-layer dimensions X/Y**, **Top-layer dimensions X/Y**, and **Motion grid cols and rows per layer**, specify the satellite parameter **Rotation angle**, and then click the **[Confirm]** button.

   > The set robot path should cover the work area.

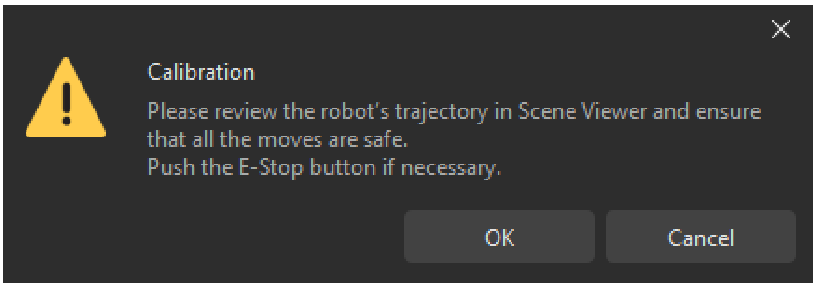
3. In the right **Scene Viewer** panel, confirm that the waypoints of the automatically generated motion path will not collide with obstacles in the environment, and then click the **[Next]** button.

Collect Images and Poses

1. In the **Collect images and poses** step, select the **Save images** checkbox.



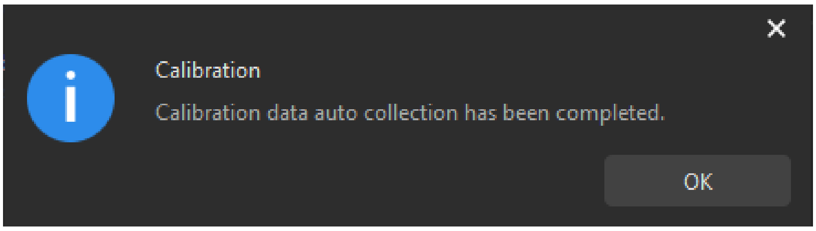
2. Click the [**Control robot to auto move along path and capture images**] button.
3. Read the safety window carefully and click the [**OK**] button.



Calibration

Please review the robot's trajectory in Scene Viewer and ensure that all the moves are safe.
Push the E-Stop button if necessary.

4. Wait until the robot finishes moving along the preset path and the camera finishes capturing images on all waypoints. Captured images can be seen in the right **Marker Image and Pose List** panel during this process.
5. After the automatic image capturing finishes, click the [**OK**] button, and then click the [**Next**] button on the bottom bar.

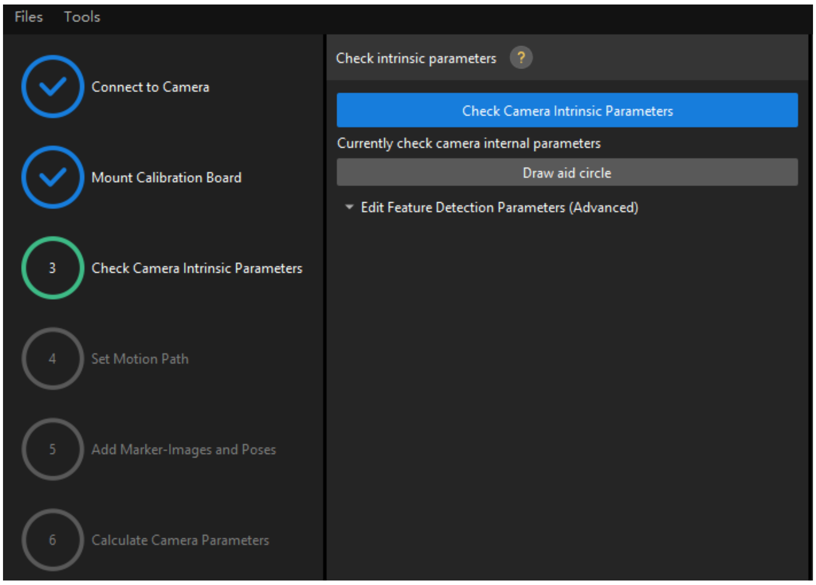


Calibration

Calibration data auto collection has been completed.

Calculate Camera Parameters

1. In the **Calculate camera parameters** window, click the [**Calculate camera extrinsic parameters**] button.


Files
Tools
Connect to Camera
Mount Calibration Board
3
Check Camera Intrinsic Parameters
4
Set Motion Path
5
Add Marker-Images and Poses
6
Calculate Camera Parameters
Check intrinsic parameters
?
Check Camera Intrinsic Parameters
Draw aid circle
Edit Feature Detection Parameters (Advanced)

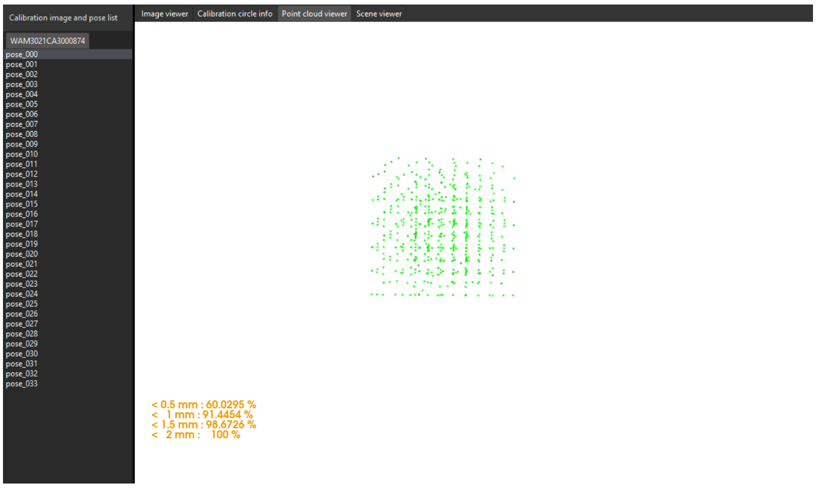
Currently check camera internal parameters

2. In the prompted window indicating calibration success, click the [OK] button.
3. View the calibration error of the point cloud in the right **Point cloud viewer** panel.

> The error point cloud shows the deviation between the calculated value and the actual value of the circles on the calibration board.

4. Confirm that the calibration accuracy meets the project requirements.

   Determine the error value with 100% to obtain the rough calibration accuracy. For example, the calibration accuracy in the following figure is within 2.5 mm.



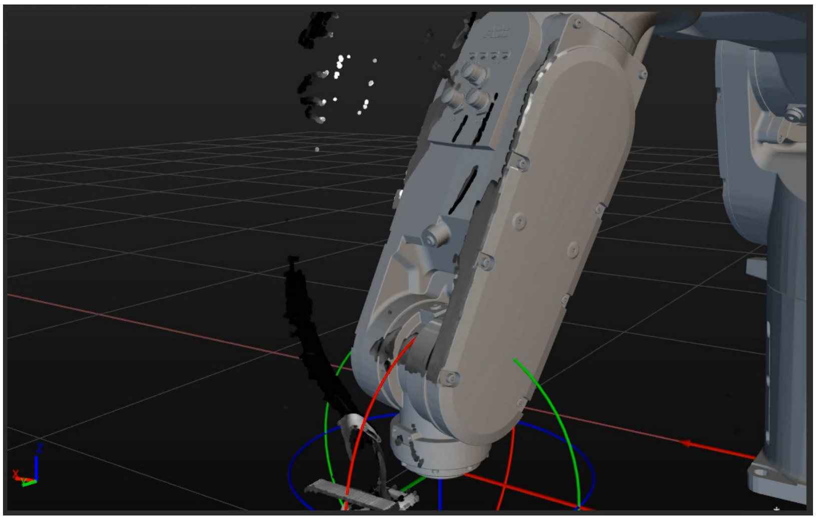
```
< 0.5 mm : 60.0295 %
<   1 mm : 91.4454 %
< 1.5 mm : 98.6726 %
<   2 mm :    100 %
```

   To improve the calibration accuracy, please refer to the section Calibration Result Check and Analysis.

Validate and Save the Calibration Result

1. After calibration, move the robot arms into the FOV of the camera.
2. In the **Calculate camera parameters** step, click the [**Re-calculate camera extrinsic parameters**] button. This operation triggers the camera to capture images.
3. Click **Scene viewer** to check the coincidence degree between the point cloud of the robot and the robot model.

If the point cloud of the robot approximately coincides with the robot model, the calibration is successful.



4. Click the [**Save**] button on the bottom bar. In the prompted **Save Calibration Files** dialog box, click the [**OK**] button. The camera calibration result will be automatically saved in the "calibration" directory of the project.

4.4. Carton Locating

Before using this tutorial, you should have created a Mech-Vision solution using the "Single-Case Cartons" case project in the "Hand-Eye Calibration" section.

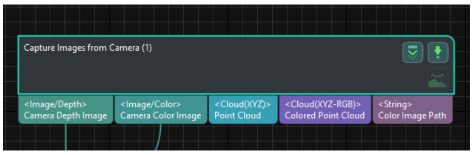
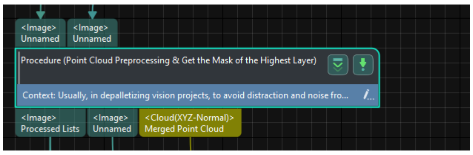
In this tutorial, you will first learn about the project workflow, and then how to deploy the project by adjusting Step parameters, to recognize the cartons' poses and output the vision result.

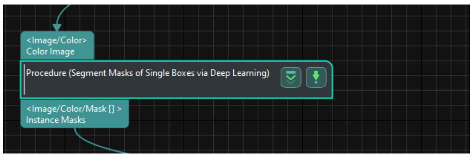
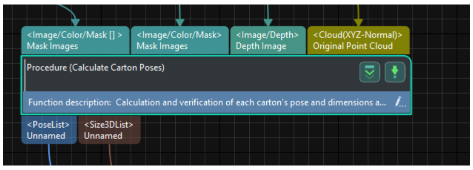
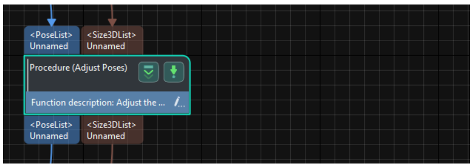
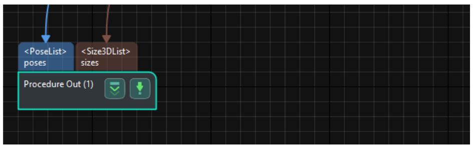
▶ https://www.youtube.com/watch?v=A-3KNbNCkMY/PLVcMd7cW2rXWbHhTQX8m2R1r6PhX6vrCI *(YouTube video)*

Video tutorial: Carton Locating

Introduction to the Project Workflow

The following table describes each Procedure in the project workflow.

No.	Step/Procedure	Image	Description
1	Capture Images from Camera	 Capture Images from Camera (1) <Image/Depth> Camera Depth Image <Image/Color> Camera Color Image <Cloud(XYZ)> Point Cloud <Cloud(XYZ-RGB)> Colored Point Cloud <String> Color Image Path	Connect to the camera and capture images of cartons
2	Point Cloud Preprocessing & Get the Mask of the Highest Layer	 Procedure (Point Cloud Preprocessing & Get the Mask of the Highest Layer) Context: Usually, in depalletizing vision projects, to avoid distraction and noise fro... <Image> Unnamed <Image> Unnamed <Image> Processed Lists <Image> Unnamed <Cloud(XYZ-Normal)> Merged Point Cloud	Perform preprocessing of the cartons' point cloud and get the mask of the highest layer

No.	Step/Procedure	Image	Description
3	Segment Masks of Individual Cartons Using Deep Learning	 <Image/Color> Color Image Procedure (Segment Masks of Single Boxes via Deep Learning) <Image/Color/Mask [] > Instance Masks	Use deep learning inference to segment masks of individual cartons based on the input mask of cartons on the highest layer, which facilitates obtaining the point cloud of an individual carton based on the corresponding mask
4	Calculate Carton Poses	 <Image/Color/Mask [] > Mask Images <Image/Color/Mask> Mask Images <Image/Depth> Depth Image <Cloud(XYZ-Normal)> Original Point Cloud Procedure (Calculate Carton Poses) Function description: Calculation and verification of each carton's pose and dimensions a... <PoseList> Unnamed <Size3DList> Unnamed	Recognize cartons' poses, and verify or adjust the recognition results based on the input carton dimensions
5	Adjust poses	 <PoseList> Unnamed <Size3DList> Unnamed Procedure (Adjust Poses) Function description: Adjust the ... <PoseList> Unnamed <Size3DList> Unnamed	Transform the reference frame of carton poses and sort the poses of multiple cartons by rows and columns
6	Output	 <PoseList> poses <Size3DList> sizes Procedure Out (1)	Output the cartons' poses for the robot to pick

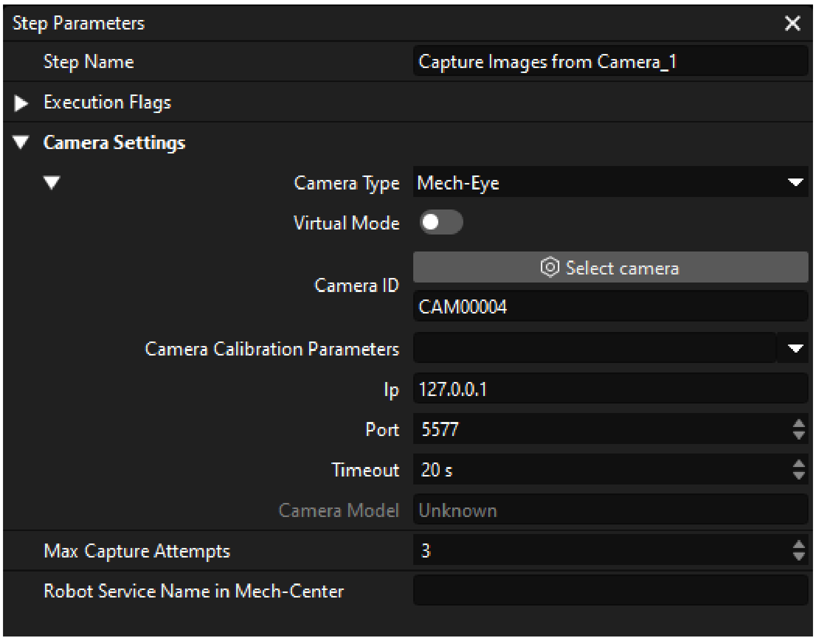
Adjust Step Parameters

In this section, you will deploy the project by adjusting the parameters of each Step or Procedure.

Capture Images from Camera

The "Single-Case Cartons" case project contains virtual data. Therefore, you need to disable the Virtual Mode and connect to the real camera in the "Capture Images from Camera" Step.

1. Select the "Capture Images from Camera" Step, disable the **Virtual Mode** option, and click **[Select camera]** on the Step parameters tab.



Step Parameters

Step Name: Capture Images from Camera_1

▶ Execution Flags

▼ Camera Settings

Camera Type: Mech-Eye

Virtual Mode: ○

Camera ID: Select camera / CAM00004

Camera Calibration Parameters:

Ip: 127.0.0.1

Port: 5577

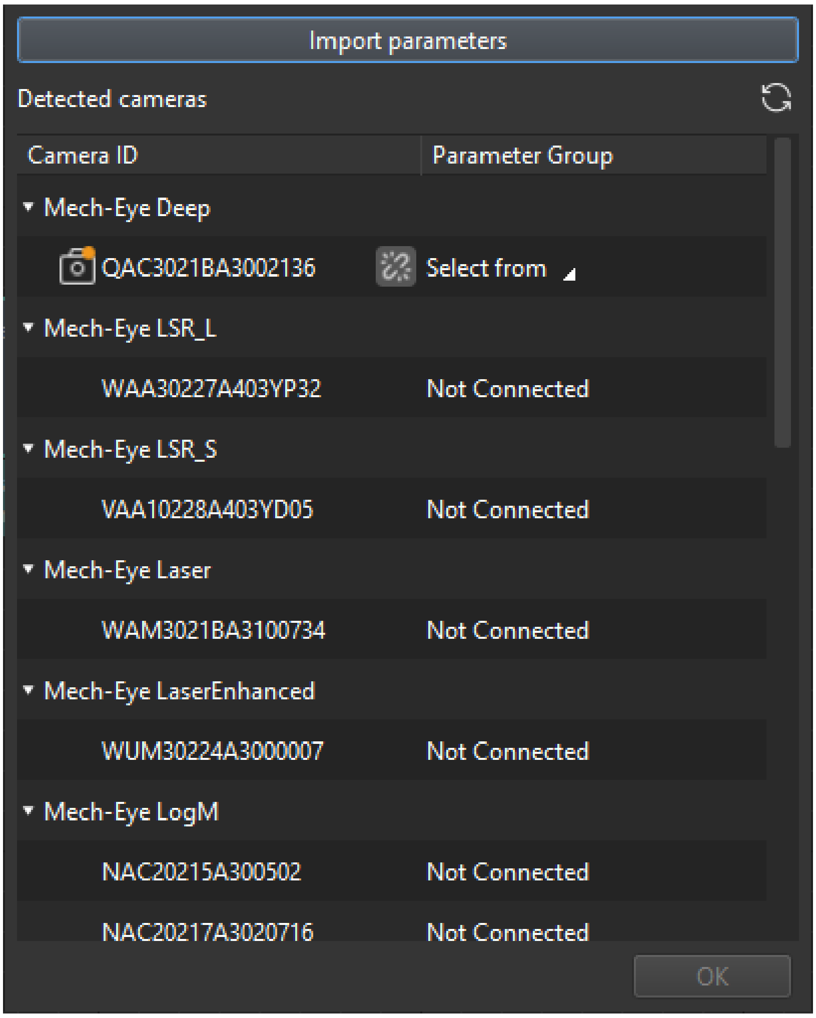
Timeout: 20 s

Camera Model: Unknown

Max Capture Attempts: 3

Robot Service Name in Mech-Center:

2. In the prompted window, click 🔗 on the right of the desired camera's serial No. to connect the camera. After the camera is connected, the icon 🔗 will turn into ⛓.



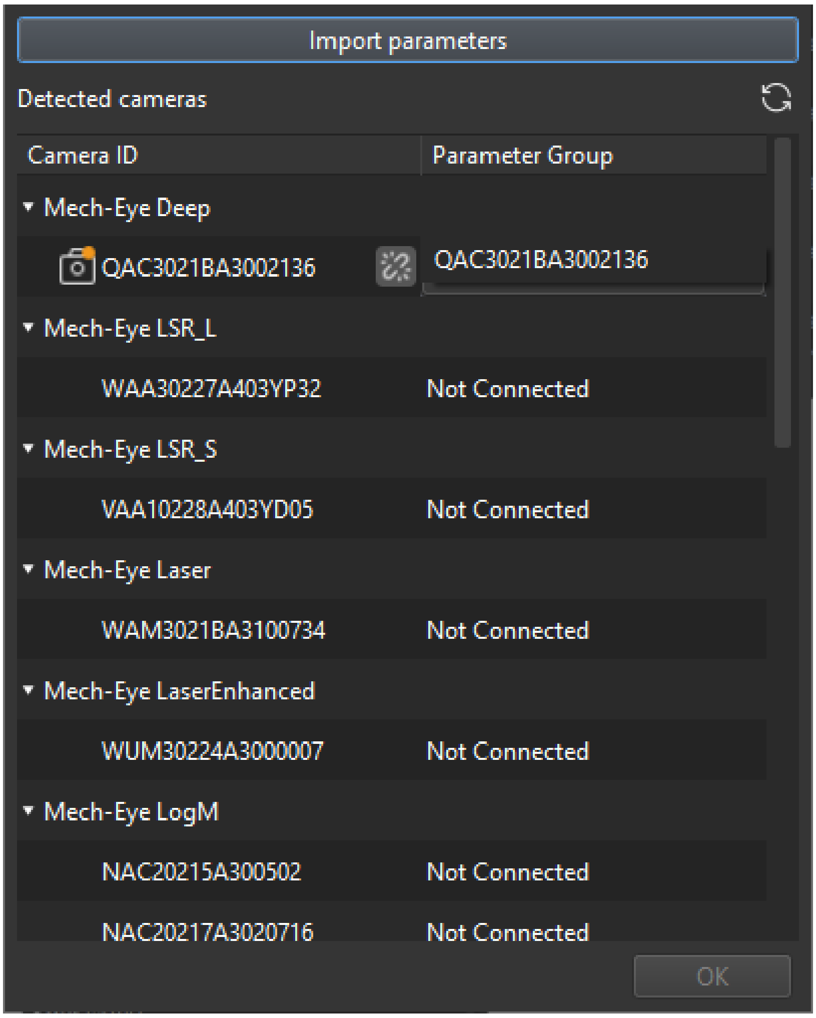
Import parameters

Detected cameras

Camera ID	Parameter Group
▼ Mech-Eye Deep	
QAC3021BA3002136	Select from
▼ Mech-Eye LSR_L	
WAA30227A403YP32	Not Connected
▼ Mech-Eye LSR_S	
VAA10228A403YD05	Not Connected
▼ Mech-Eye Laser	
WAM3021BA3100734	Not Connected
▼ Mech-Eye LaserEnhanced	
WUM30224A3000007	Not Connected
▼ Mech-Eye LogM	
NAC20215A300502	Not Connected
NAC20217A3020716	Not Connected

OK

After the camera is connected, click [Select from] to select the calibrated parameter group.

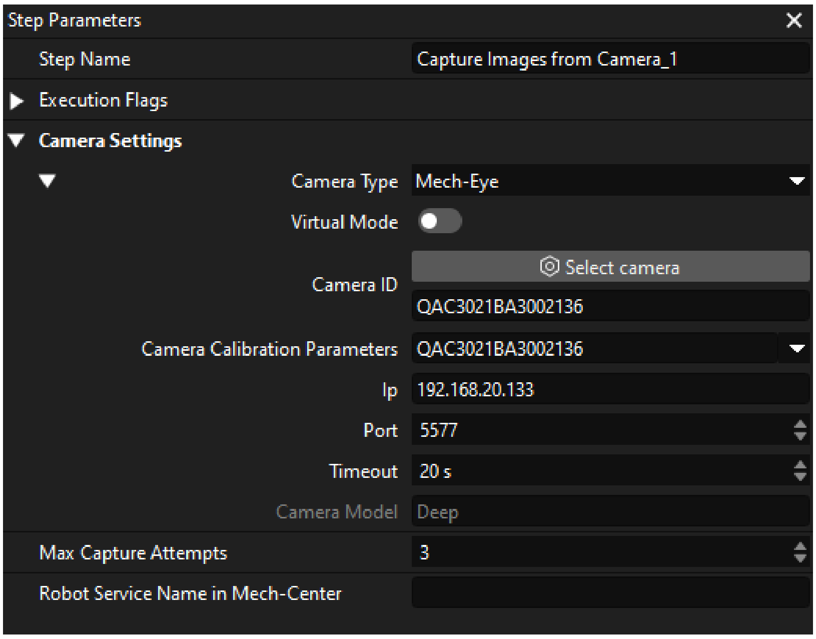


Import parameters

Detected cameras

Camera ID	Parameter Group
Mech-Eye Deep	
QAC3021BA3002136	QAC3021BA3002136
Mech-Eye LSR_L	
WAA30227A403YP32	Not Connected
Mech-Eye LSR_S	
VAA10228A403YD05	Not Connected
Mech-Eye Laser	
WAM3021BA3100734	Not Connected
Mech-Eye LaserEnhanced	
WUM30224A3000007	Not Connected
Mech-Eye LogM	
NAC20215A300502	Not Connected
NAC20217A3020716	Not Connected

3. After the camera is connected and the parameter group is selected, the calibration parameter group, IP address, and ports of the camera will be obtained automatically. Just keep the default settings of the other parameters.



Step Parameters

Step Name	Capture Images from Camera_1
Execution Flags	
Camera Settings	
Camera Type	Mech-Eye
Virtual Mode	○
Camera ID	Select camera QAC3021BA3002136
Camera Calibration Parameters	QAC3021BA3002136
Ip	192.168.20.133
Port	5577
Timeout	20 s
Camera Model	Deep
Max Capture Attempts	3
Robot Service Name in Mech-Center	

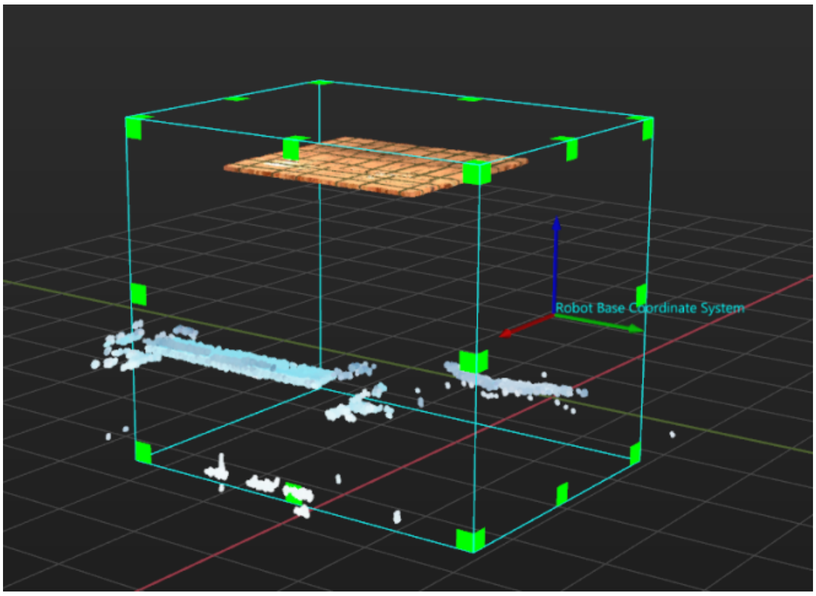
Now, you have connected the software to the camera.

Point Cloud Preprocessing & Get the Mask of the Highest Layer

To prevent the robot from colliding with other cartons while picking items from the non-highest layer, it is necessary to use this Procedure to obtain the mask of the cartons on the highest layer. By giving priority to picking these cartons, you can minimize the risk of collisions during the picking process.

In this Procedure, yo need to adjust the parameter **3D ROI** and **Layer Height**.

1. On the Step parameters tab, click the **[Set 3D ROI]** button to set the 3D ROI.



The 3D ROI frame should generally include the highest and lowest regions of the carton stack, and contain unwanted points as less as possible.

2. To avoid obtaining the cartons not on the highest layer, set the **Layer Height** parameter. The parameter value should be less than the height of a single carton in the stack, for example, half of the carton height. Usually, you just need to keep the recommended value.

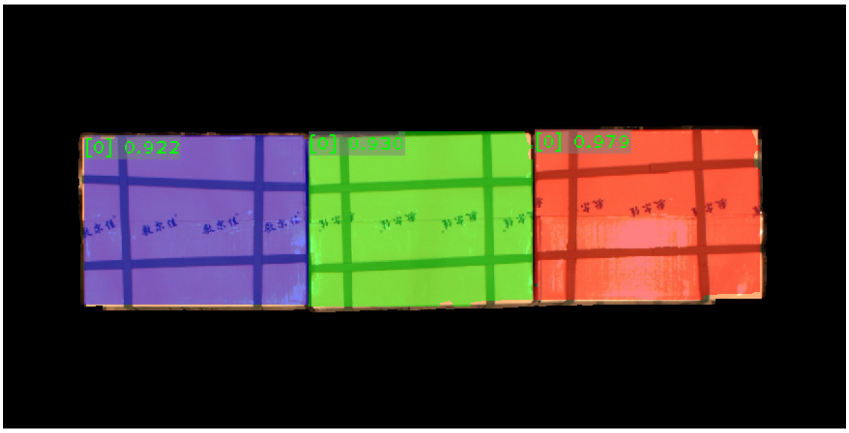
   If the dimensions of single cartons in different stacks are different, you should set the **Layer Height** parameter according to the height of the lowest cartons.

If the setting of the **Layer Height** parameter is improper, the project will obtain the cartons not on the highest layer, which causes the collision between the robot and other cartons during the picking process.

Segment Masks of Individual Cartons Using Deep Learning

After obtaining the mask of the cartons on the highest layer, you need to use deep learning to segment the masks of individual cartons.

The current case project has a built-in instance segmentation model package suitable for cartons. After running this Procedure, you will get the masks of individual cartons, as shown below.



If the segmentation results are not satisfactory, you can adjust the size of the 3D ROI accordingly.

Calculate Carton Poses

After obtaining the point clouds of individual cartons, you can calculate carton poses. In addition, you can enter the dimensions of the carton to verify the correctness of the recognition results.

In the “Calculate Carton Poses” Procedure, set the parameter **Length on X-axis/Y-axis/Z-axis** and **Box Dimension Error Tolerance**:

- **Length on X-axis/Y-axis/Z-axis**: set these parameters according to the actual dimensions of cartons.
- **Box Dimension Error Tolerance**: Keep the default value 30 mm. If the input carton dimensions and the recognized ones are significantly different, you can try to adjust this parameter.

Adjust poses

To facilitate robot picking, you need to adjust the cartons’ poses to transform them from the camera reference frame to the robot reference frame after obtaining the cartons’ poses.

In this Procedure, you can also sort the cartons’ poses by rows and columns to help the robot to pick items in a certain sequence.

- **Ascending (by Carton Pose’s X Value in Robot Base Reference Frame)**: Usually, keep the default setting (selected). When this option is selected, cartons in rows will be sorted in the ascending order of carton poses' X-coordinate values in the robot base reference frame; otherwise, cartons in rows will be sorted in the descending order.
- **Ascending (by Carton Pose’s Y Value in Robot Base Reference Frame)**: Usually, keep the default setting (selected). When this option is selected, cartons in rows will be sorted in the ascending order of carton poses' X-coordinate values in the robot base reference frame; otherwise, cartons in rows will be sorted in the descending order.

Procedure Out

After obtaining the proper carton poses, the “Procedure Out” Step sends the results of the current project to the backend service.

So far, you have deployed the project for carton locating.

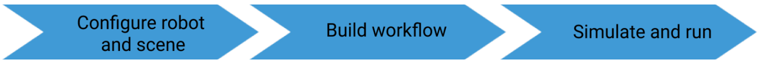
4.5. Pick and Place

After having obtained the cartons' poses using a Mech-Vision case project, you can then build a workflow for the Mech-Viz project to guide the robot to pick and place cartons repeatedly.

▶ https://www.youtube.com/watch?v=pL4ZbeoDF8A/PLVcMd7cW2rXWbHhTQX8m2R1r6PhX6vrCI *(YouTube video)*

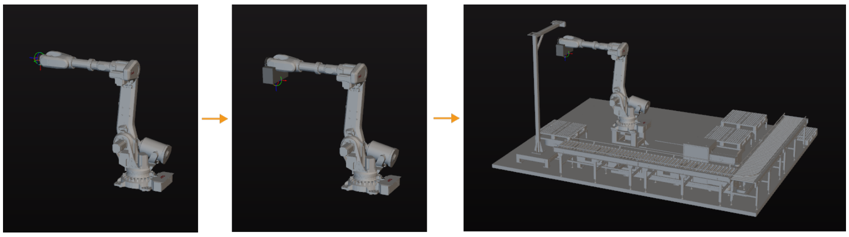
Video Tutorial: Pick and Place

The following figure shows the process of building a project.



Configure the Robot and Scene

To avoid collision with surrounding objects during robot picking and placing, you need to add the tool model, and scene model to the project for collision detection. The process is shown in the figure below.



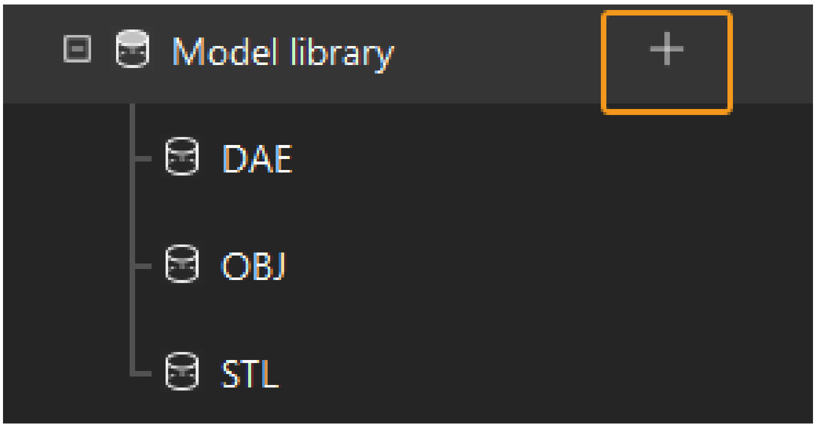
Import and Configure the Tool Model

- Tool refers to the device mounted on the robot end that performs processing/picking jobs.
- In this tutorial, the tool is a gripper containing an array of suction cups.

The end tool should be imported and configured so that its model can be displayed in the 3D simulation space and used for collision detection.

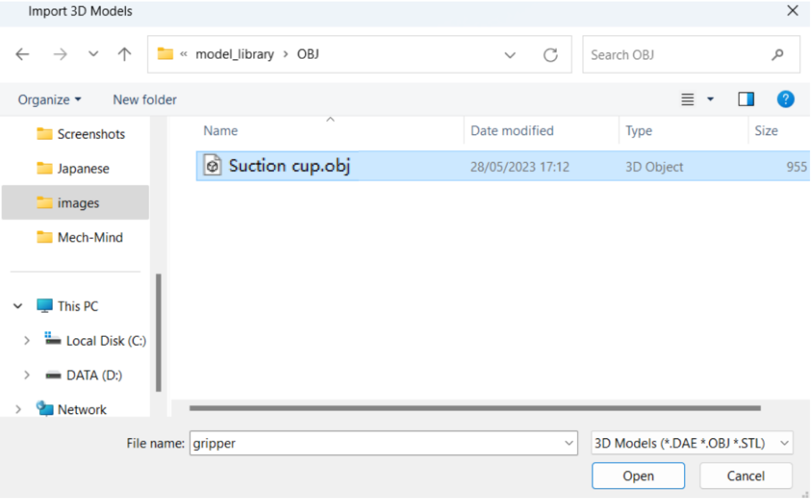
Import the Tool Model

1. Click [+] in Resources › Model Library.



Project resources refer to various fundamental resources used in the project, including the robot, tools, workobjects, and scene objects.

2. In the prompted window, select the collision model file in .obj format, and then click **[Open]**.

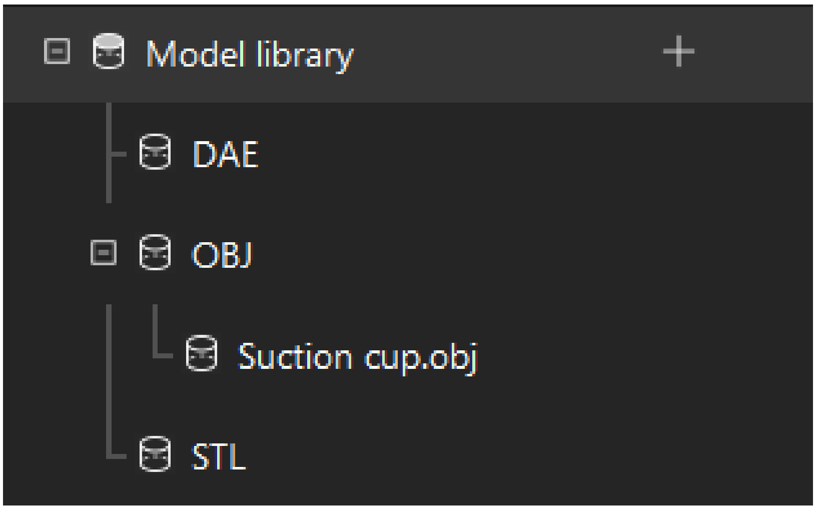


Suction cup.obj 28/05/2023 17:12 3D Object 955

File name: gripper

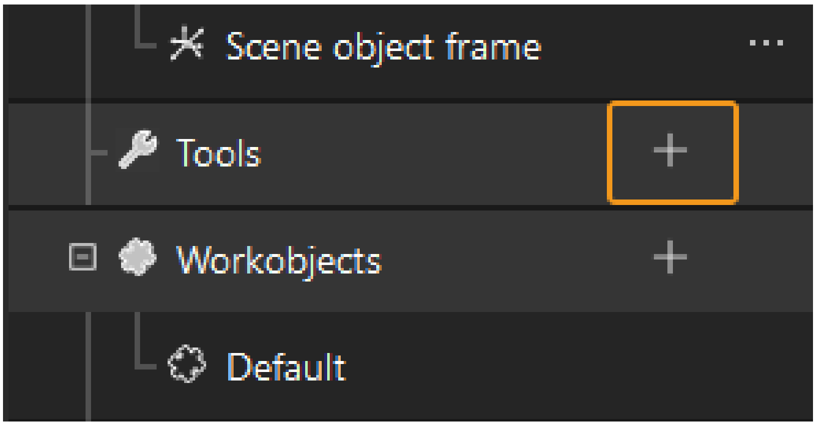
3D Models (*.DAE *.OBJ *.STL)

3. After the model is imported, you can see the imported model in the model library.

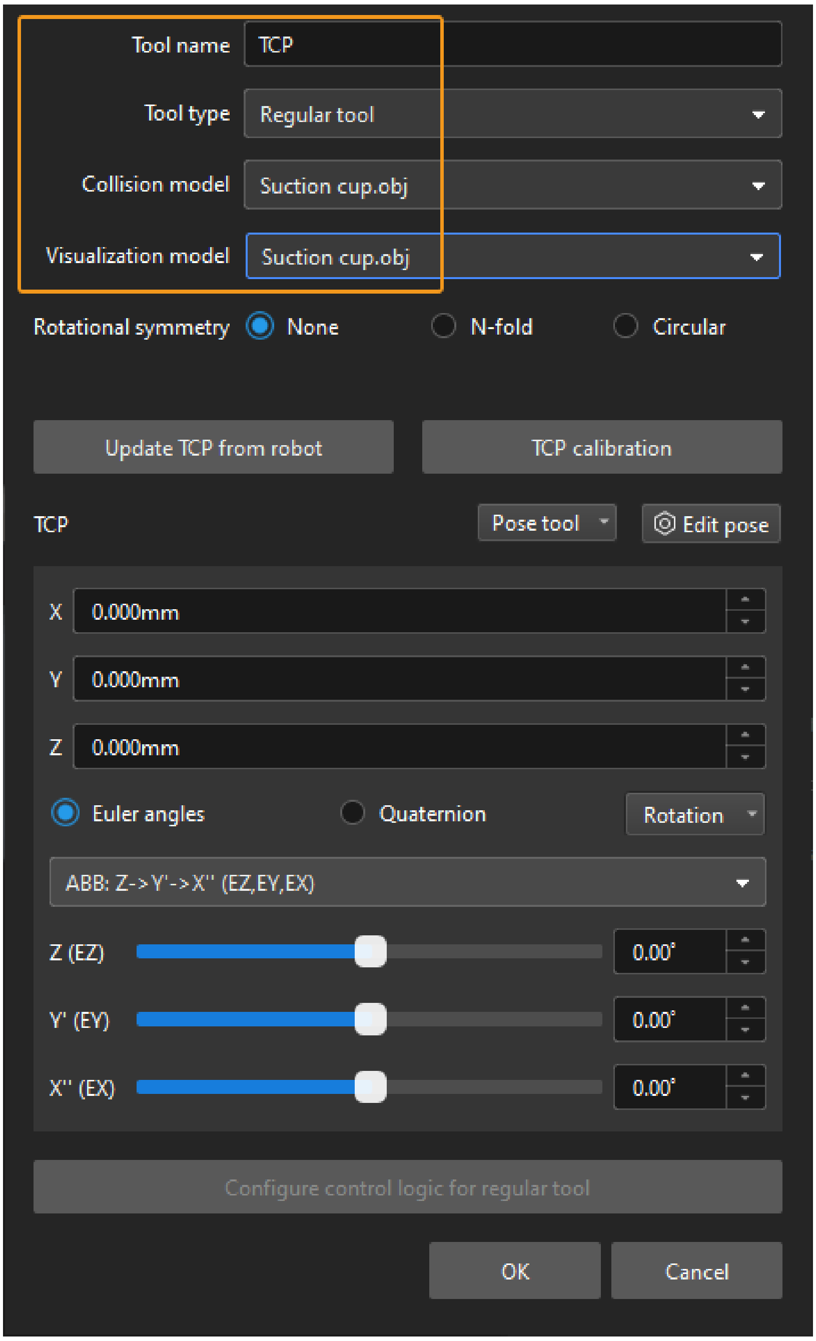


Configure the Tool

1. Click **[+]** in Resources › Tools.



2. In the prompted window, enter the tool name, select the imported tool model as the collision model and visualization model, and then click [OK].



Tool name: TCP

Tool type: Regular tool

Collision model: Suction cup.obj

Visualization model: Suction cup.obj

Rotational symmetry: ● None ○ N-fold ○ Circular

TCP

X: 0.000mm

Y: 0.000mm

Z: 0.000mm

● Euler angles ○ Quaternion

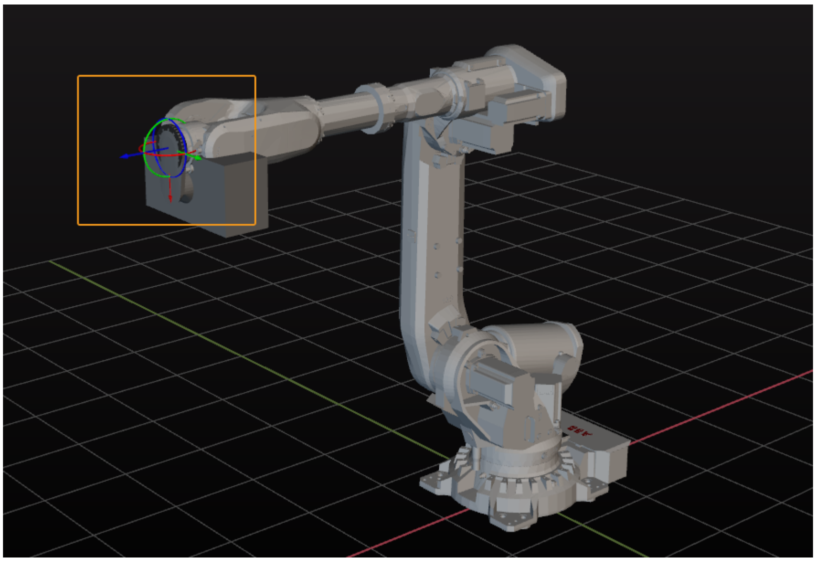
ABB: Z->Y'->X'' (EZ,EY,EX)

Z (EZ): 0.00°

Y' (EY): 0.00°

X'' (EX): 0.00°

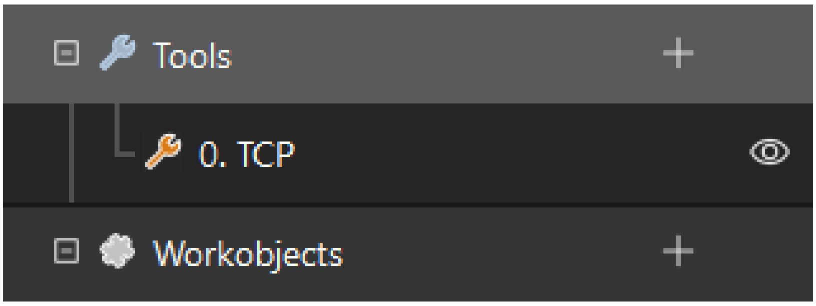
3. After the tool is configured, you can view the configured tool in the 3D simulation space, as shown below.



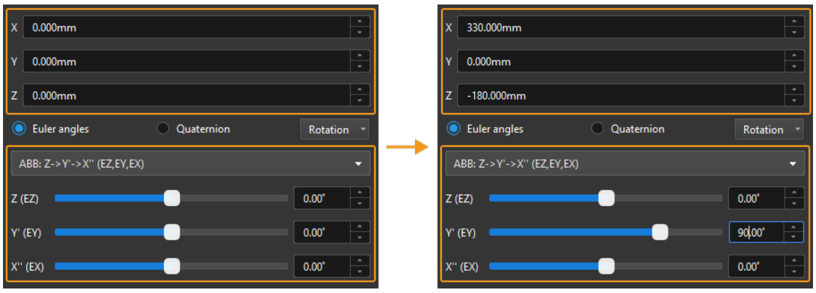
Adjust the Tool

As shown above, the position of the Tool Center Point (TCP) in relation to the robot is obviously wrong. So, you should adjust the position of the TCP so that it is at the tip of the tool.

1. Double-click the tool model in the model library.



2. In the prompted window, adjust the TCP (Tool Center Point).

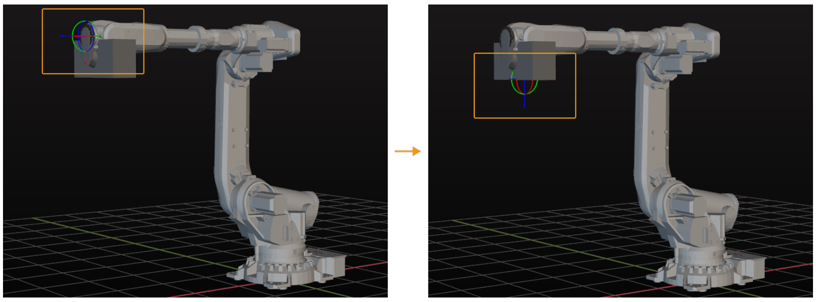


X 0.000mm
Y 0.000mm
Z 0.000mm
● Euler angles ○ Quaternion Rotation
ABB: Z->Y'->X'' (EZ,EY,EX)
Z (EZ) 0.00°
Y' (EY) 0.00°
X'' (EX) 0.00°

X 330.000mm
Y 0.000mm
Z -180.000mm
● Euler angles ○ Quaternion Rotation
ABB: Z->Y'->X'' (EZ,EY,EX)
Z (EZ) 0.00°
Y' (EY) 90.00°
X'' (EX) 0.00°

> The TCP is usually on the tip of the tool. In this tutorial, the TCP in the center points of the suction cups.

3. The tool’s poses before and after the adjustment are shown below.

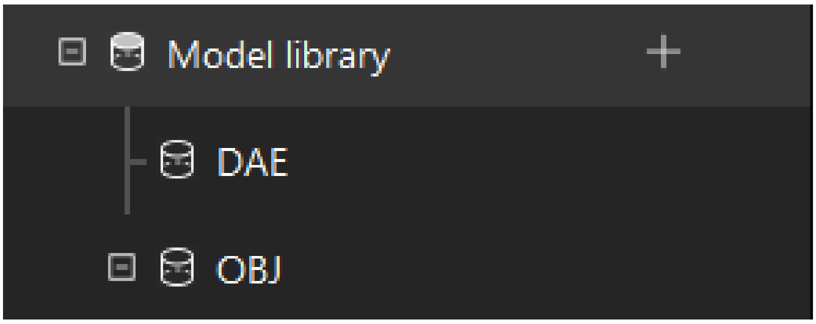


Import and Configure the Scene Model

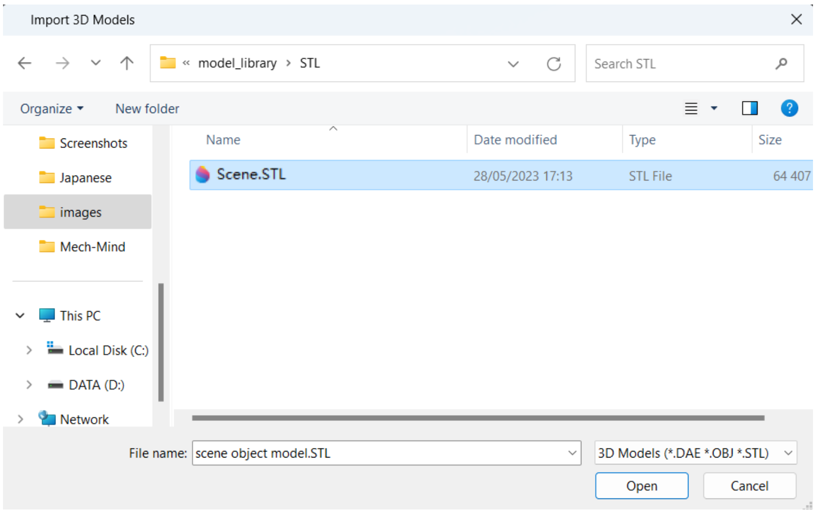
Scene objects are imported and configured to make the scene in the software closer to the real scenario, which facilitates the planning of the collision-free robot motion path.

Import the Scene Model

1. Click [+] in Resources › Model Library.



2. In the prompted window, select the scene object model file, and then click [Open].

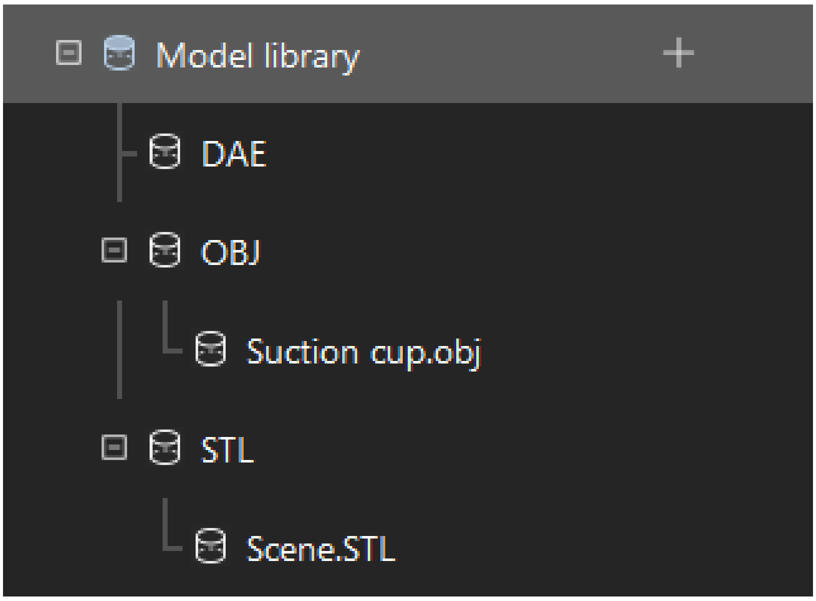


Name	Date modified	Type	Size
Scene.STL	28/05/2023 17:13	STL File	64 407

File name: scene object model.STL

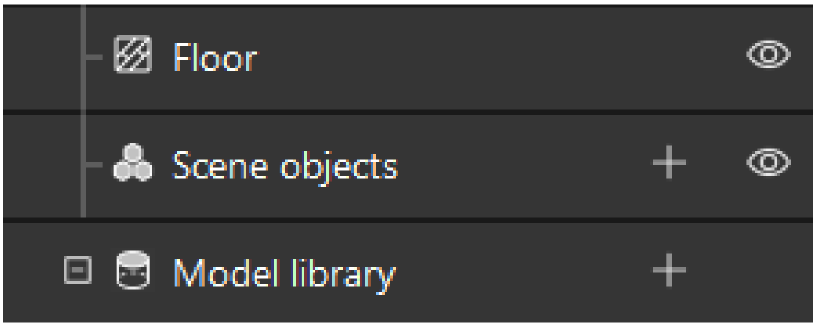
3D Models (*.DAE *.OBJ *.STL)

3. After the model is imported, you can see the imported model in the model library.

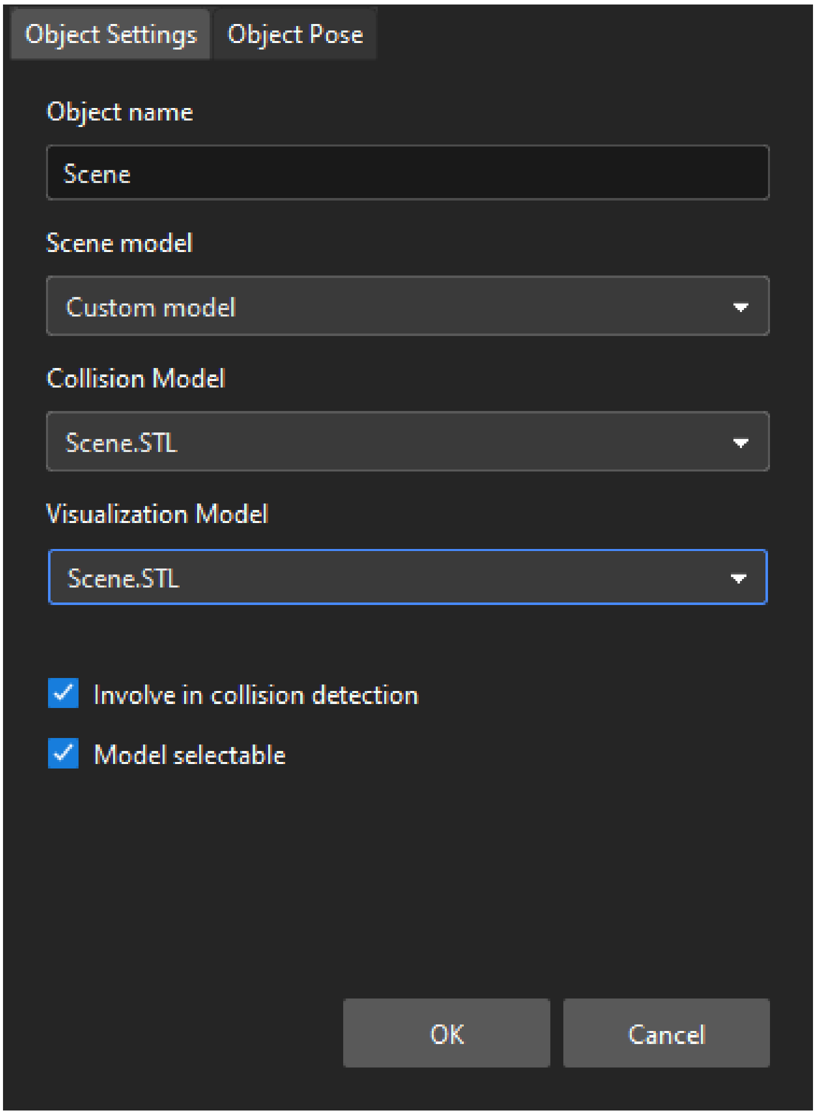


Configure the Scene Model

1. Click [+] in Resources › Scene Objects.



2. In the prompted window, enter the object name, select “Custom model” as the “Scene model”, select the imported scene object model as the collision model and visualization model, and then click [OK].



Object Settings | Object Pose

Object name: Scene

Scene model: Custom model

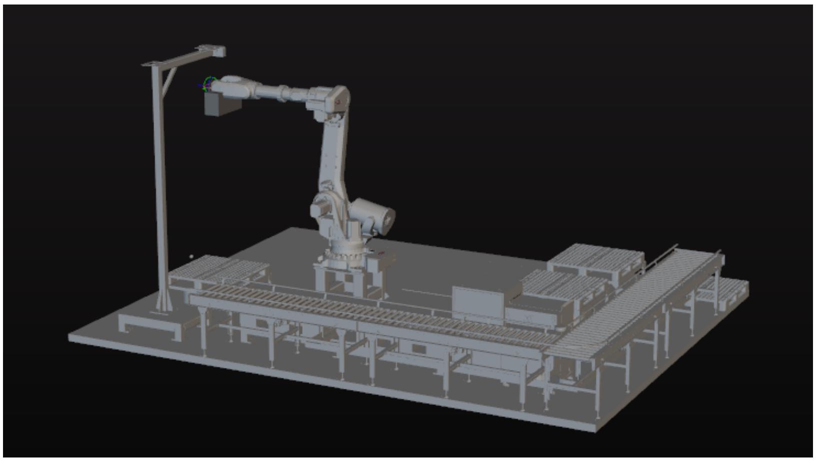
Collision Model: Scene.STL

Visualization Model: Scene.STL

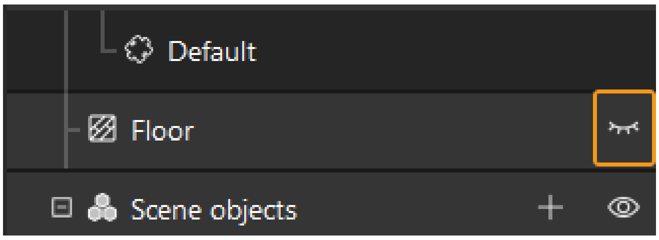
☑ Involve in collision detection

☑ Model selectable

3. After configuration, the scene object is displayed in the 3D simulation space.



To view the scene model better, you can hide the floor.



Default

Floor

Scene objects

After configuring the models, click **[Sync Robot]** on the toolbar to keep the simulated robot's pose consistent with that of the real robot.

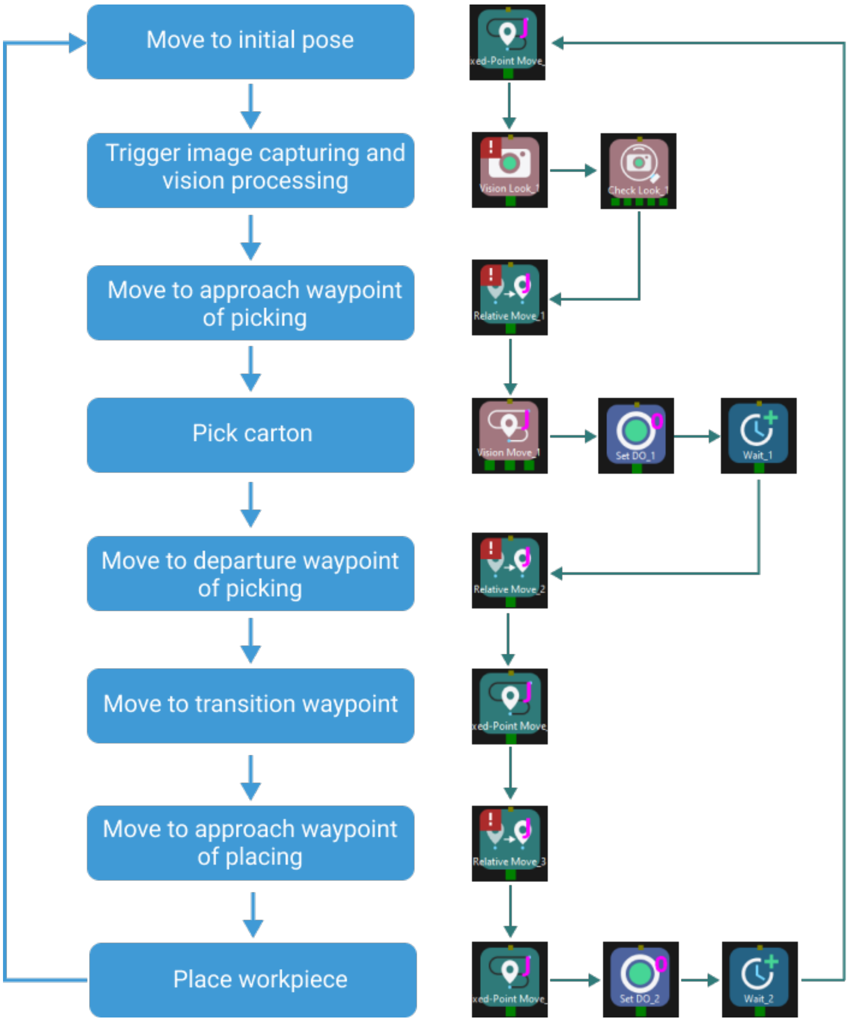
Build a Workflow

Now that you have configured the models, you can start creating a workflow. Drag a step from the step library to the graphical programming workspace, set the parameters of the Steps, and connect Steps to achieve preset program functions.

- The workflow refers to the robot motion control program created in Mech-Viz in the form of a flowchart.
- A Step refers to a function module of robot programming.

Introduction to the Project Workflow

The built workflow is shown below.



Move to initial pose

Trigger image capturing and vision processing

Move to approach waypoint of picking

Pick carton

Move to departure waypoint of picking

Move to transition waypoint

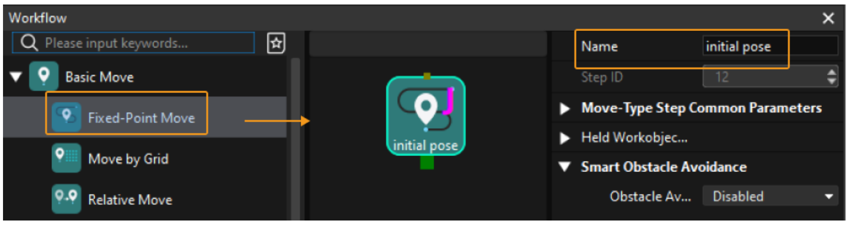
Move to approach waypoint of placing

Place workpiece

Move to “Initial Pose”

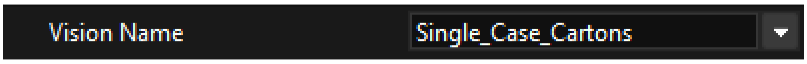
The initial pose is the initial point of the robot motion, and it also should be a safe position. At this pose, the robot should be away from the objects to pick and the surrounding devices, and should not block the camera’s FOV.

After the robot arrives at the self-defined home position, select **Fixed-Point Move** in the Step Library, and drag it to the graphical programming workspace, rename it as “initial pose”. Then click **[Sync Robot]** on the toolbar to record the current pose of the robot for this Step.



Name initial pose
Step ID 12
Obstacle Av... Disabled

Trigger Image Capturing and Vision Processing

Step	Vision Look
Description	Start the Mech-Vision project to obtain the vision recognition result.
Operation	Find the “Vision Look” Step in the Step Library, and drag it to the graphical programming workspace.
Parameter Settings	Select **Single_Case_Cartons** from the **Vision Service Name** drop-down list.
Image	 Vision Name Single_Case_Cartons

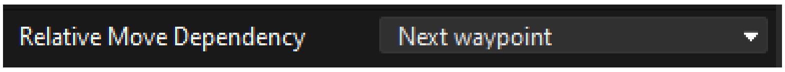
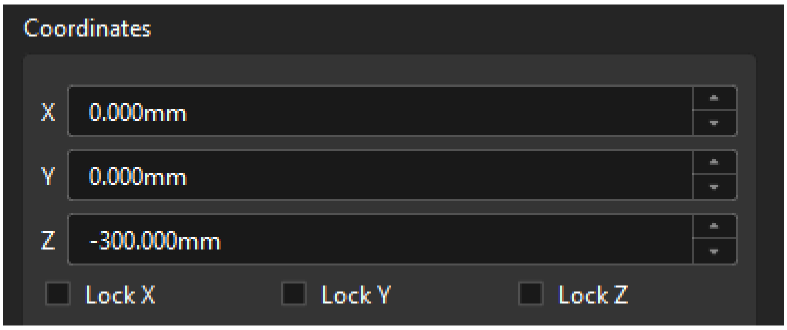
After the “Vision Look” Step, add the **Check Look** Step to check whether there is any vision result.

Step	Check Look
Description	Check whether there is any vision result.
Operation	Find the “Check Look” Step in the Step Library, and drag it to the graphical programming workspace.
Parameter Settings	Keep the default parameter settings.

Move the Robot to the Approach Waypoint of Picking

When picking cartons, if the robot directly moves from the initial pose to the picking waypoint, collision with the cartons may occur. After obtaining the vision recognition result, you can then use the **Relative Move** step to move the robot to the approach waypoint.

Step	Relative Move
Description	The robot moves according to the vision recognition result.
Operation	Find the “Relative Move” Step in the Step Library, drag it to the graphical programming workspace, and rename it as “Approach waypoint 1”.

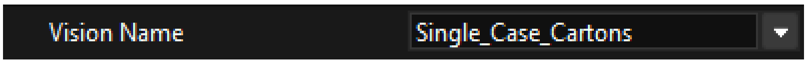
Parameter Settings	Set **Relative Move Dependency** to **Next waypoint**; set **Waypoint type** to **Tool**, and set the **Z** coordinate to a proper value, such as -300mm.
Image	 Relative Move Dependency: Next waypoint  Coordinates X: 0.000mm Y: 0.000mm Z: -300.000mm □ Lock X □ Lock Y □ Lock Z

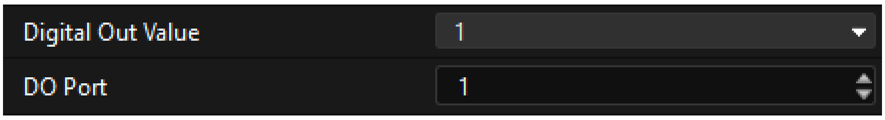
Pick the Carton

After the robot has arrived at the approach waypoint, you can control the robot to pick cartons. The picking process can be divided into two steps.

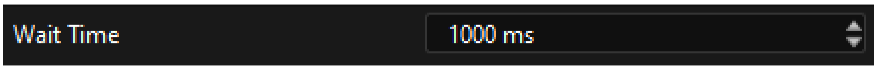
1. Step 1: Use the **Vision Move** Step to control the robot to arrive at the picking waypoint.
2. Step 2: Use the **Set DO** Step to control the robot to pick the carton using the suction cups.

See the following for details.

Step	Vision Move
Description	The robot moves according to the vision recognition result.
Operation	Find the "Vision Move" Step in the Step Library, and drag it to the graphical programming workspace.
Parameter Settings	Select **Single_Case_Cartons** from the **Vision Service Name** drop-down list.
Image	 Vision Name: Single_Case_Cartons

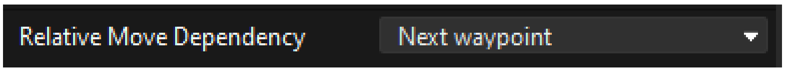
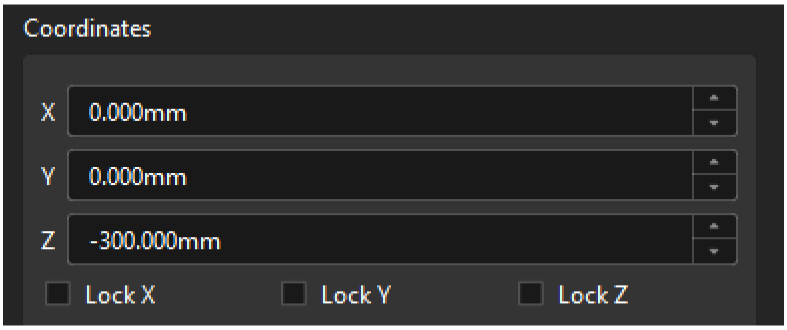
Step	Set DO
Description	Control the suction cups to pick the carton.
Operation	Find the "Set DO" Step in the Step Library, drag it to the graphical programming workspace, and rename it as "Trigger Suction Cup".
Parameter Settings	Set the **Digital Out Value** and **DO Port** parameters to **1**. These parameter values are reference only. Please determine them according to actual on-site requirements.
Image	 Digital Out Value: 1 DO Port: 1

To ensure that the robot can pick the carton securely, add the **Wait** Step.

Step	Wait
Description	To avoid the failures to pick cartons.
Operation	Find the “Wait” Step in the Step Library, drag it to the graphical programming workspace, and rename it as “Wait for firm gripping”.
Parameter Settings	Set the **Wait Time** parameter to **1000ms**.
Image	 Wait Time 1000 ms

Move the Robot to the Departure Waypoint of Picking

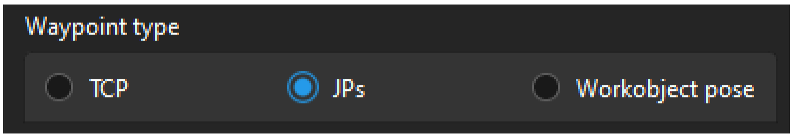
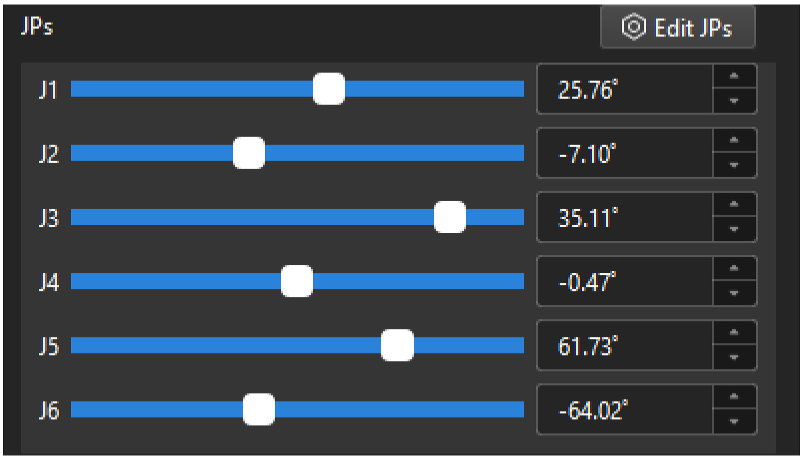
After the robot picks the carton, you need to use the **Relative Move** Step to move the robot to the departure waypoint to avoid the collision between the carton and the scene objects.

Step	Relative Move
Description	The robot moves according to the vision recognition result.
Operation	Find the “Relative Move” Step in the Step Library, drag it to the graphical programming workspace, and rename it as “Departure waypoint”.
Parameter Settings	Set **Relative Move Dependency** to **Next waypoint**; set **Waypoint type** to **Tool**, and set the **Z** coordinate to a value larger than the height of the cartons. For example, if the carton height is 500 mm, set the **Z** coordinate to -600 mm.
Image	 Relative Move Dependency Next waypoint  Coordinates X 0.000mm Y 0.000mm Z -300.000mm □ Lock X □ Lock Y □ Lock Z

Move the Robot to a Transition Waypoint

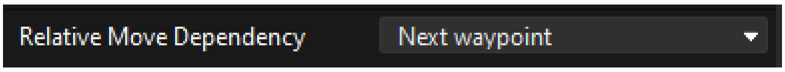
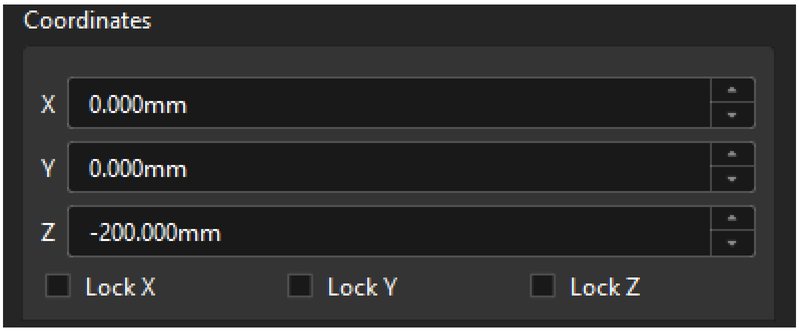
To avoid collision, you can add transition waypoints using the **Fixed-Point Move** Step after the departure waypoint to optimize the picking and placing path.

Step	Fixed-Point Move
Description	Optimize the robot picking and placing path
Operation	Find the “Fixed-Point Move” Step in the Step Library, drag it to the graphical programming workspace, and rename it as “Transition waypoint”.
Parameter Settings	Set **Waypoint type** to **JPs** and set the **JPs** values.

Image	 Waypoint type ○ TCP ● JPs ○ Workobject pose  JPs J1 25.76° J2 -7.10° J3 35.11° J4 -0.47° J5 61.73° J6 -64.02°

Move the Robot to the Approach Waypoint of Placing

Before the robot places the carton, you need to use the **Relative Move** Step to move the robot to the approach waypoint for placing to avoid collision between the carton and the scene object.

Step	Relative Move
Description	The robot moves according to the vision recognition result.
Operation	Find the “Relative Move” Step in the Step Library, drag it to the graphical programming workspace, and rename it as “Approach waypoint 2”.
Parameter Settings	Set **Relative Move Dependency** to **Next waypoint**; set **Waypoint type** to **Tool**, and set the **Z** coordinate to a proper value, such as **-200 mm**.
Image	 Relative Move Dependency: Next waypoint  Coordinates X 0.000mm Y 0.000mm Z -200.000mm □ Lock X □ Lock Y □ Lock Z

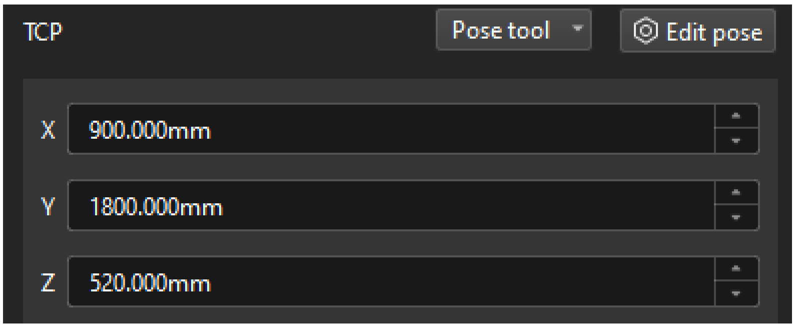
Place the Carton

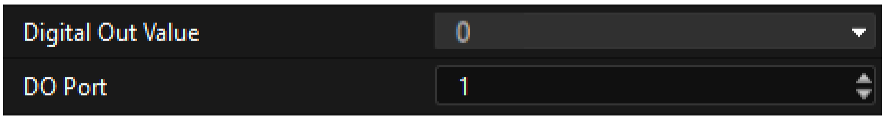
After the robot arrives at the approach waypoint for placing, you can control the robot to place cartons. The picking process can be divided into two steps.

1. Step 1: Use the **Fixed-Point Move** Step to control the robot to arrive at the placing waypoint.

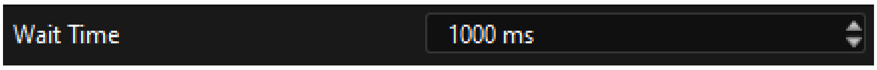
2. Step 2: Use the **Set DO** Step to control the robot to release the suction cups to place the carton.

See the following for details.

Step	Fixed-Point Move
Description	Control the robot to arrive at the placing waypoint.
Operation	Find the “Fixed-Point Move” Step in the Step Library, drag it to the graphical programming workspace, and rename it as “Placing waypoint”.
Parameter Settings	Set a suitable **TCP** as the placing waypoint.
Image	 TCP X 900.000mm Y 1800.000mm Z 520.000mm

Step	Set DO
Description	Release the suction cups to place the carton.
Operation	Find the “Set DO” Step in the Step Library, drag it to the graphical programming workspace, and rename it as “Release suction cup”.
Parameter Settings	Set the **Digital Out Value** parameter to **0** and the **DO Port** parameter to **1**. These parameter values are reference only. Please determine them according to actual on-site requirements.
Image	 Digital Out Value 0 DO Port 1

Since it takes time for the suction cup to release, you need to add the **Wait** Step to avoid the failure of placing the carton by the robot.

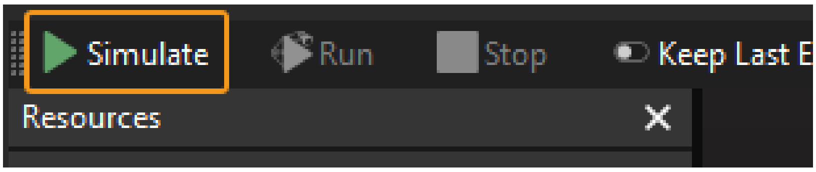
Step	Wait
Description	Avoid the failure of placing the carton by the robot.
Operation	Find the “Wait” Step in the Step Library, drag it to the graphical programming workspace, and rename it as “Wait for release”.
Parameter Settings	Set the **Wait Time** parameter to **1000ms**.
Image	 Wait Time 1000 ms

Connect Steps

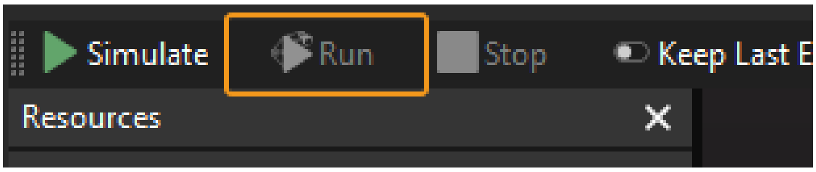
After the preceding Steps are added, connect them one by one. To control the robot to pick and place cartons repeatedly, you need to connect the exit port of "Placing Wait" to the entry port of "Initial Pose".

Simulate and Run the Project

1. After all the Steps are connected properly, click the [Simulate] button on the toolbar to simulate the Mech-Viz project.



2. If the project runs as expected in simulation, click the [Run] button on the Mech-Viz toolbar to run the project with the real robot.



It is recommended that the robot should move at a low speed and that you pay attention to the robot motion trajectory. In an emergency, press the emergency stop key on the teach pendant.

5. Topic: Improving Picking Accuracy

In actual production environments, after the 3D vision–guided applications run for a certain period, issues such as "robot picking inaccuracy" may sometimes occur. This issue indicates a decrease in the application’s picking accuracy, and that the picking errors have exceeded the picking accuracy requirements of the project.

This topic aims to guide you in improving picking accuracy during the application deployment to meet the project’s accuracy requirements, and troubleshooting the "robot picking inaccuracy" issue in the stable production stage.

Learn about Picking Accuracy

Introduces the concepts related to picking accuracy and various sources of picking errors.

```
<div class="card-row two-column-row">
```

5.1.

Overview of Picking Accuracy

Introduces the concept and meaning of picking accuracy, and clarifies misunderstandings about picking accuracy and picking errors.

Introduction to Picking Accuracy

Picking Accuracy Remarks

5.2.

Sources of Picking Errors

Introduces the various sources of errors leading to robot picking inaccuracy, and their impacts, and analyzes the reasons that cause the errors.

Sources of Picking Errors

```
</div>
```

Ensure Qualified Picking Accuracy

During the actual application deployment, you can refer to this section to ensure that the application’s picking accuracy meets the project accuracy requirements.