User Manual

MIRAI Training

VERSION 26.0.2

micropsi
industries

Contents

Introduction

Micropsi Industries' robot control system, MIRAI, allows users to automate complex motion tasks in dynamic process environments. MIRAI is a machine learning-based system that generates robot movements based on real-time image data and sensory input.

Today, most automation solutions are programmed using a scripting language or taught with a teach pendant. In contrast, MIRAI trains robots to solve complex hand-eye coordination tasks by "watching" a human operator perform these tasks and imitating the observed actions.

Key advantages of MIRAI over traditional automation approaches include:

> **Ease of Use:** MIRAI allows users to solve and automate complex motion tasks without requiring prior expert knowledge in automation, machine vision, or programming.
>
> **Adaptability:** MIRAI's underlying machine learning technology can handle a wide range of process variations and dynamic changes within both the task and its environment. This capability allows the MIRAI controller to address new classes of tasks and problems in automation and robotics that previously required complex and expensive solutions or were impossible to automate.

To train the robot, the user guides the robot arm between the target position and the intended starting position(s). These recorded demonstrations of motion paths are called *episodes*. For more complex tasks, the user performs and records multiple episodes by guiding the robot along a precise trajectory from different starting positions to the target position.

These episodes are then transformed into a vision-based robot motion program by our cloud-based machine learning service, creating a *MIRAI skill*. Skills allow the MIRAI controller to replicate the motions and actions previously demonstrated by the user, accurately steering robot movements and managing task variations in real time. Operating in a closed-loop system, the controller continuously receives feedback on the task environment and adjusts the robot's actions to guide the end effector to the target.

1 Safety Precautions

	Hazards during hand-guiding
1	**Squeezing or crushing:** Caused by robot parts approaching fixed structures or rigid objects.
2	**Cutting or puncture:** Caused by workpieces, tools, or structures with sharp edges or points.
3	**Shearing:** Caused by moving robot parts approaching fixed structures or rigid objects.
4	**Burns:** Caused by contact with hot surfaces of parts in the MIRAI setup.
5	**Strain or fatigue:** Caused by long periods of hand guiding.

During the training process, robot movement can be controlled by hand-guiding. When hand-guiding is activated, the robot can be moved in any direction. Follow all safety precautions to prevent injury.

- Be careful not to squeeze your free hand and fingers or shear any body parts.
- Pay extra attention when the robot is handling workpieces with sharp edges or sharp points, which can result in cuts or puncture wounds if the robot and human collide.
- Exercise additional caution if the robot is set to automatic mode to test the demonstrated movements. High-speed robot movements can crush body parts.

Safety precautions

- ☑ When possible, use the automated recording assistant instead of hand-guiding.
- ☑ Wear safety gloves during training.
- ☑ Wear safety glasses during training, especially when workpieces have sharp edges or points.

2 Security: Access control

2.1 Access control

The access control feature is a security feature that protects the MIRAI Training App with password authentication. When access control is activated, you must enter your password in specific situations to continue using the app. You must also enter your password to turn off access control.

When password entry is required

After access control is activated, you will be required to enter your password to restart the application when:

- **Sending the app to the background.** Robot control will stop automatically when this occurs.
- **Closing the app.**
- **Turning off the tablet display.**

2.2 Set up access control

You may activate access control at any time.

1. From the main menu, tap **Access control**.
2. Enter a password of your choice.
   - Any length is acceptable
   - All characters are supported
3. Tap **Set password and turn on** to activate access control.
   - A confirmation message appears: *Password has been set*
   - The button text changes to **Turn off**

PERFORMANCE NOTE

If access control is activated and the tablet is not connected to the MIRAI controller when you open the MIRAI Training App or bring it from the background, you will be locked out of the app. An error message will prompt you to reconnect.

- If you're using a custom IP address, enter it in the field provided and tap **OK**.
- Otherwise, connect the tablet to the controller via Wi-Fi and tap **OK**.

2.3 Turn off access control

1. From the main menu, tap **Access control**.
2. Tap **Turn off**.
3. Enter your current password when prompted to deactivate access control. The screen will now display "OFF" next to the header.

No further password prompts will appear.

2.4 Password recovery

If you forget your password, follow the steps below.

> **WARNING**
>
> This procedure requires system-level access and should only be performed by authorized personnel.

1. Pause the robot program running MIRAI to ensure that no skills or classifiers are executed.
2. **Log in as netadmin:**
   a. Access the controller via SSH or a physical terminal, and then use Linux command-line tools.
   b. Enter the username `netadmin`.
   c. Enter the Wi-Fi password. The password is typically printed on the controller.
3. Run `reset-auth`.
4. Then run `mirai-restart`.
   IMPORTANT: Ensure that no skills or classifiers are being executed while running this command.
5. Verify that the tablet running the MIRAI Training App is connected to the robot via Wi-Fi.
6. Navigate to Access Control, and then enter a new password.

3 The training setup

Every new MIRAI skill starts with the physical setup of the task environment: defining the skill's working area, mounting the cameras to observe this area, and adjusting the lighting conditions.

3.1 Plan the training setup

1. **Define the skill's working area.** This consists of two aspects:
   - Consider how the positions of relevant objects are expected to vary. What extreme positions will the robot move to while executing the task?
   - Consider the plausible robot trajectories. What motion paths will the robot follow to reach a variety of different plausible object positions?

2. **Set the "handover point"** – where the MIRAI skill will take over control from the main robot program. For example, in a table-top task, the handover point is typically a central position directly above the work area.

3. **Optimize the camera viewpoint:**
   - MIRAI supports both wrist-mounted (dynamic) and floor-mounted (fixed) cameras, which can be configured as single or dual setups. A single wrist-mounted camera is effective for most tasks, including those requiring 3-D positioning.
   - Position the camera(s) to capture changes in the position of the target object. The camera(s) should be positioned roughly perpendicular to the plane where precise positioning is needed.
   - Make sure that all task-relevant objects, including the target object(s) and the end effector, stay in view throughout the entire motion path.
   - Choose a viewpoint where visible background features remain constant over the skill's production lifetime. For example, the frame should show the work surface rather than the production flow. While reliable skills can be trained with varying backgrounds, this requires more time and effort to record sufficient episodes.
   - For multi-target positioning skills, the camera(s) should be positioned roughly parallel to the robot tool, looking down along it and ideally showing the object (or part of it) closest to the tool center point (TCP).

4. **Choose the lighting scheme:**
   - Consistent lighting conditions make it easier to train reliable skills. Training with varying lighting conditions requires more time and effort to record sufficient episodes.
   - Consider how lighting in the work area might change over the skill's production lifetime, including different times of day and seasonal variations.
   - Achieve consistent lighting with bright lights that neutralize shifts in ambient light.
   - For wrist-mounted cameras, use a bright ring light that moves with the camera.
   - SETUP (WORK AREA) keep distracting objects out of view (task-unrelated objects). While it is possible to train reliable skills even in the presence of distractors, it comes at the cost of increased demonstration effort.
     Make sure **there is no obstruction in the direct path** from the intended starting point to the target.

3.2 Choose the MIRAI skill type

After planning your training setup, choose the type of MIRAI skill that will best execute your task. MIRAI can learn to move along arbitrary paths, reacting in real time to changes in the task environment.

There are three types of MIRAI skills:

1. Positioning skills
2. Multi-target positioning skills
3. Motion skills

Note

- **Positioning skills** can be trained with or without a force/torque sensor.
- **Multi-target skills** and **motion skills** must be trained with a force/torque sensor.

For details, refer to 4 Creating skills in the MIRAI Training App.

3.2.1 Positioning skills

Positioning skills allow you to position the robot's tool center point in real time relative to a visible target, such as an object for picking or a workpiece for insertion. The robot follows the most direct trajectory to the target position, typically a straight path.

To use positioning skills, two conditions must be met:

1. The direct path from the handover point where MIRAI takes control to the target position must be free of obstructions.
2. The target object(s) must remain stationary during the MIRAI-controlled motion.

Typical use cases for positioning skills include:

- Placing a gripper in a pre-grip pose above an object that may vary in color, size, shape, or position
- Positioning a workpiece precisely for insertion in a subsequent step
- Placing a sensor tool relative to a workpiece for quality inspection

Positioning skills generally require less intensive training compared to motion skills, as they are highly robust with fewer recorded episodes. Training can be fully automated using the recording assistant, eliminating the need for a force/torque sensor.

3.2.2 Multi-target positioning skills

Multi-target positioning skills allow the robot to handle tasks involving multiple targets or objects. The robot moves in a straight, direct path towards the closest target. The robot then returns to survey the scene and moves to the next closest target, until targets have been reached.

Typical use cases for multi-target positioning skills include:

- Picking objects from a flat surface (3 or 4 degrees of freedom)
- Picking an object from multiple positions

3.2.3 Motion skills

Motion skills allow the robot to perform more complex movements in real time. This is useful for a wide range of applications, such as insertion, contour following, picking objects from a moving conveyor, or tool positioning in crowded environments where a direct path is not feasible.

Typical use cases for motion skills are:

- Various insertion tasks, such as cable plugging
- Path- or contour-following tasks
- Picking or placing objects from/on a moving conveyor
- Quality inspection tasks (positioning sensors or cameras)
- Finishing tasks like polishing or gluing
- Tool positioning where obstacles prevent a direct path from the starting point to the goal

Achieving robust motion skills involves thorough training, with repeated demonstration of precise movements across varied paths.

3.3 Set up the first training

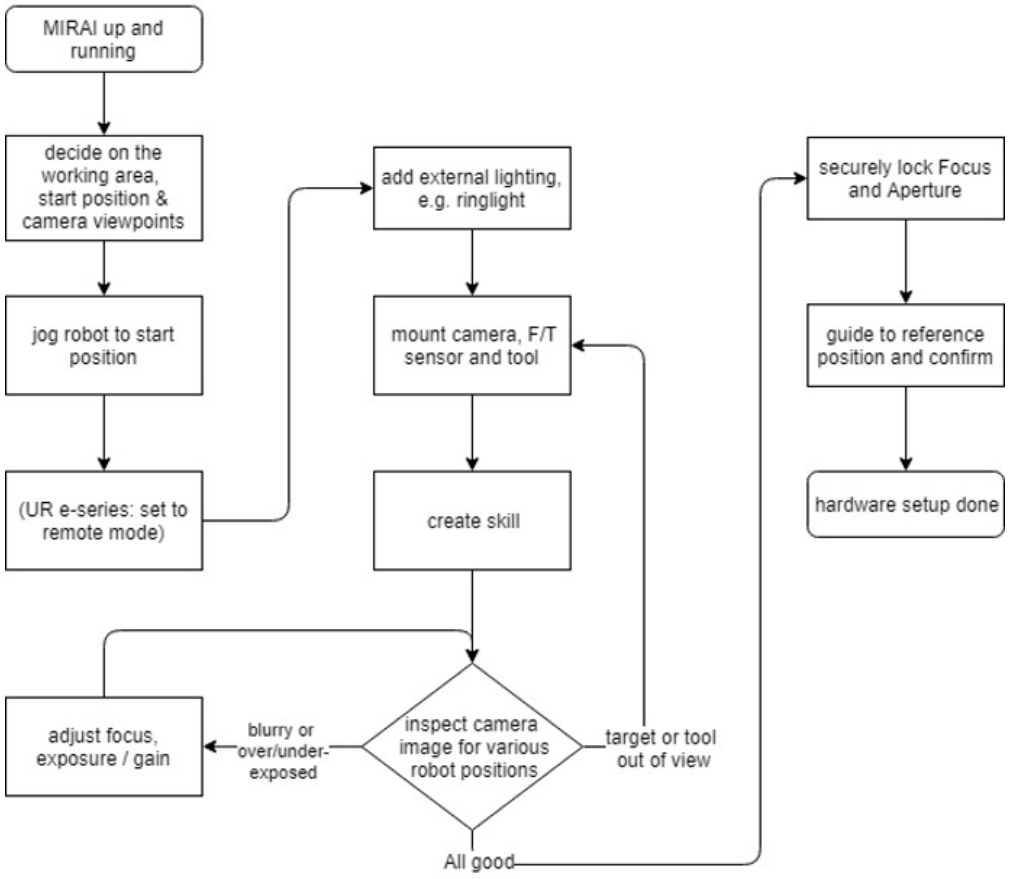


Figure 1: Skill setup flowchart

Set up the robot and workspace:

1. Mount the camera(s), force/torque sensor (if using), and end-of-arm tool on the robot, as described in the Micropsi Industries Robot Integration Guide for your robot platform.
2. Jog or hand-guide the robot to the handover position.
3. Add adequate lighting, as chosen during planning.

Optimize the camera settings:

1. Open the MIRAI Training App and create a new skill (see chapter 4 Creating skills in the MIRAI Training App).
2. Inspect the live camera image displayed in the app:
    - Assess the field of view and ensure that the relevant objects remain in view throughout the intended motion.
    - Guide the robot to various points along the intended motion paths, while watching the image preview.
    - The target object(s) should remain entirely in view throughout the motion path, for every expected position of the target objects.
    - The robot's end effector should remain in view throughout the motion.
3. Determine the camera settings:
    - Try different camera holders and lenses until the field of view fulfills the above criteria.
    - Adjust the focus slightly beyond the end of the tool/manipulator, so the sharpest image occurs when the tool is about to touch the target object.
    - Adjust the aperture.
        - The middle value is generally adequate for most applications.
        - We recommend using even numbers for the aperture so that you can use the markings on the camera as a guide.
        - Closing the aperture reduces the lens opening, increasing the depth of field. This keeps more of the scene in focus, which is useful for keeping the target object sharp across the entire movement path. Conversely, opening the aperture blurs the background, which can also be beneficial during skill training. Additionally, closing the aperture reduces the light entering the camera, so you may need to adjust exposure settings to maintain proper brightness. Balancing these settings may require some experimentation.
    - Adjust settings in the MIRAI Training App:
        - Adjust the exposure time and gain to minimize over/under-exposure.
        - Guide the robot through some sample motions from start to target positions, to assess the exposure along the entire intended motion path.
4. Save the settings:
    - *On the camera:* Secure the focus and aperture settings using small screws on the lens. When the setup is complete, fix the screws with glue.
    - *In the MIRAI Training App:* Save the camera settings.

3.4 Quantify the Target Position

The goal of this step is to ensure you can recreate your ideal tool positioning above the target for future episode training sessions.

Note

The ability to **recreate the ideal position above the target** is the key to obtaining good training data.

- Move the robot tool into the target position where the main robot program should take over from the trained MIRAI skill after the skill's execution.
- MIRAI's target position should not be directly on the workpiece – the target position should be offset from the workpiece in such a way that the robot will have a clear path to the target object.
  - **1. Example:** If inserting a wire into a hole, the target position should offset the wire tip several millimeters perpendicular to the plane of the hole.
  - **2. Example:** If grabbing a part, the target position should place the gripper fingers at an offset of several millimeters above the part.

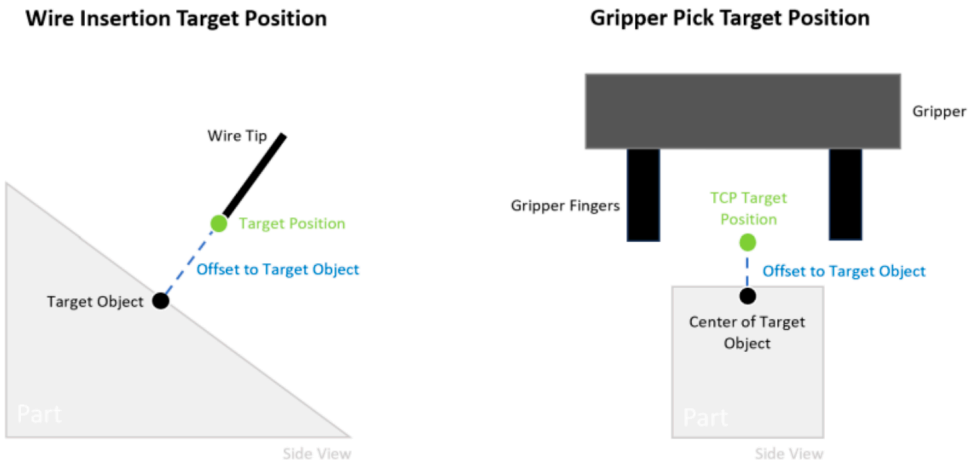


Figure 2: The ideal target position

- Quantify the robot tools target position – where exactly is it in relation to the part? Measure distances and angles between the tool and the part.
- Take note of the movements that brought the robot into this position.
- Record any measurements or notes taken above.

3.5 Switch control between MIRAI and the robot

While training MIRAI, you may need to use the teach pendant to move the robot. For example, you might jog the robot to a specific position or make small adjustments to achieve the desired camera view.

If the robot's controller is used to move the robot, MIRAI will disconnect, triggering a pop-up message in the Training App. To return control to MIRAI, activate remote control or hold the deadman switch, depending on your robot platform. Then press **Retry** to continue in the app.

This message appears in the following MIRAI Training App screens:

- Camera settings
- Set reference position
- Recording episodes
- Recording assistant setup
- Testing skills

4 Creating skills in the MIRAI Training App

In this step, you will enter details about your skills in the MIRAI Training App. This section guides you through each screen and provides additional information on specific settings.

Before you begin:

- Confirm your setup meets the requirements for the skill you intend to train (see 4.1).
- Install, connect, and power on the robot, camera(s), and force/torque sensor, if using.
- Be able to measure the distance between the tool center point (TCP) and the flange, as well as any rotations around the TCP.
- Connect to the Micropsi cloud.

4.1 Setup requirements for MIRAI skills

MIRAI has three types of skills:

- **Single-target positioning skills:** Use for accurate real-time positioning of a tool where the path is a straight, direct motion from the starting point to the target position.
- **Multi-target positioning skills:** Use for tasks that involve a robot moving in a straight, direct path towards multiple objects or positions.
- **Motion skills:** Use for complex non-linear motion tasks, or where the direct path is not possible, such as avoiding around an obstacle or following a path.

For further details on each skill type, refer to 3.2 Choose the MIRAI skill type.

Check the table below to confirm your setup meets the requirements for your intended skill type.

Setup requirements and options for each skill type

Setup	Single-target positioning skill,	Multi-target positioning skill	Motion skill
F/T sensor	Optional	Required	Required
Camera mount	Wrist	Wrist	Wrist or static
Rotations	None, 1 axis, or 3 axis	None, 1 axis, or 3 axis	None or 1 axis
Translations	Full axis translations *(cannot restrict)*	Full axis translations *(cannot restrict)*	Axis restrictions *(select up to 2 axes)*

4.2 Create a new skill

This section includes screenshots that show example inputs and camera images to illustrate the process of creating a skill. The example skill demonstrated here is a single-target positioning skill for inserting a DIMM (a small circuit board) into a computer. Example inputs are highlighted with blue arrows.

For some fields in the app, different skill types have distinct instructions. These are indicated with the following notices:

POSITIONING SKILLS
MULTI-TARGET POSITIONING SKILLS
MOTION SKILLS

To begin: Access the main menu, go to **Ability Overview**, and tap **Add new ability +**. Then follow the steps below. The numbered steps match the numbered sections in each screenshot for quick reference.

! IMPORTANT: You must connect to the Micropsi cloud to add a new skill in the MIRAI Training App.

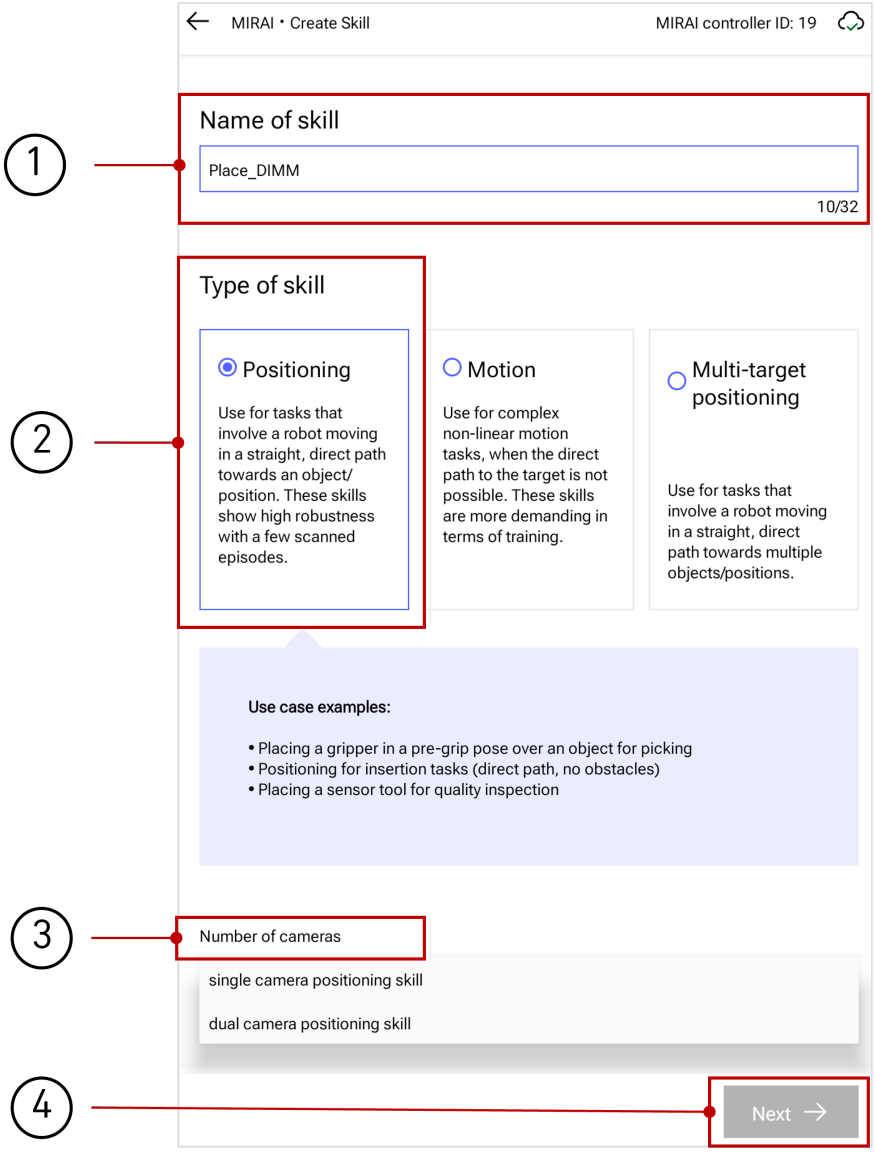
1. Under **Name of skill**, enter a name that clearly describes what the skill does. The skill name cannot be changed later.

   The screenshot shows an example skill name, "Place_DIMM."

2. Under **Type of skill**, select the skill type by tapping on the appropriate box.

3. Select the number of cameras for your skill.
   NOTE: The drop-down menu options will change according to skill type.

4. Tap **Next** to proceed to the next screen, "Skill Configuration."



5. Under **Robot Selection**, select the robot for this skill from the drop-down menu.

6. Under **Device Configuration,** select options for force/torque sensor and camera(s):

   a. Under **Force/Torque Sensor:**

   POSITIONING SKILLS: Choose one of two options:

   - *If **not** using a force/torque sensor*, select **None** from the drop-down menu.
   - *If using a force/torque sensor*, select it from the drop-down menu if not preselected.

   MULTI-TARGET POSITIONING SKILLS: Select the device from the drop-down menu, if not preselected.

   MOTION SKILLS: Select the device from the drop-down menu, if not preselected.

   b. Under **Camera 1/Camera 2**, select the camera from the drop-down menu.

TIP: If a device is missing from the drop-down menus, ensure it is installed, connected, and powered on. After connecting a new device, restart the MIRAI controller to detect the change. For cameras, check the following:

- *USB 3.0 cameras:* Ensure the camera(s) are connected to the the MIRAI controller's USB port(s).
- *GigE cameras:* Ensure that the IP address of the camera(s) is set to the correct subnet and that each camera is selected in the MIRAI Training App's network settings. For instructions, see the relevant Robot Integration Guide.

7. Under **How are cameras mounted?:**

   POSITIONING SKILLS: The **Wrist mounted** option will be preselected.

   MULTI-TARGET POSITIONING SKILLS: The **Wrist mounted** option will be preselected.

   MOTION SKILLS: Choose **Wrist mounted** or **Static mounted**, where the camera is in the workspace and separate from the robot arm.

   TIP: Use a stationary camera only when necessary, as it greatly impacts rotation behavior. Rotations and actions will use the robot's fixed base frame rather than the TCP's relative action frame, as in wrist-mounted setups.

8. Under **Axis Configuration:** Select options for translations and rotations according to skill type:

   a. Under **Translations:**

   POSITIONING SKILLS: Deactivated (axes cannot be restricted).

   MULTI-TARGET POSITIONING SKILLS: Deactivated (axes cannot be restricted).

   MOTION SKILLS: Select up to 2 axes to restrict the robot's range of motion. You can train robot movements along the X, Y, and Z axes. Deselecting an axis will movement along that axis.

   In the screenshot, the Translations option is grayed out because the example shows a single-target positioning skill. For motion skills, this option will be available.

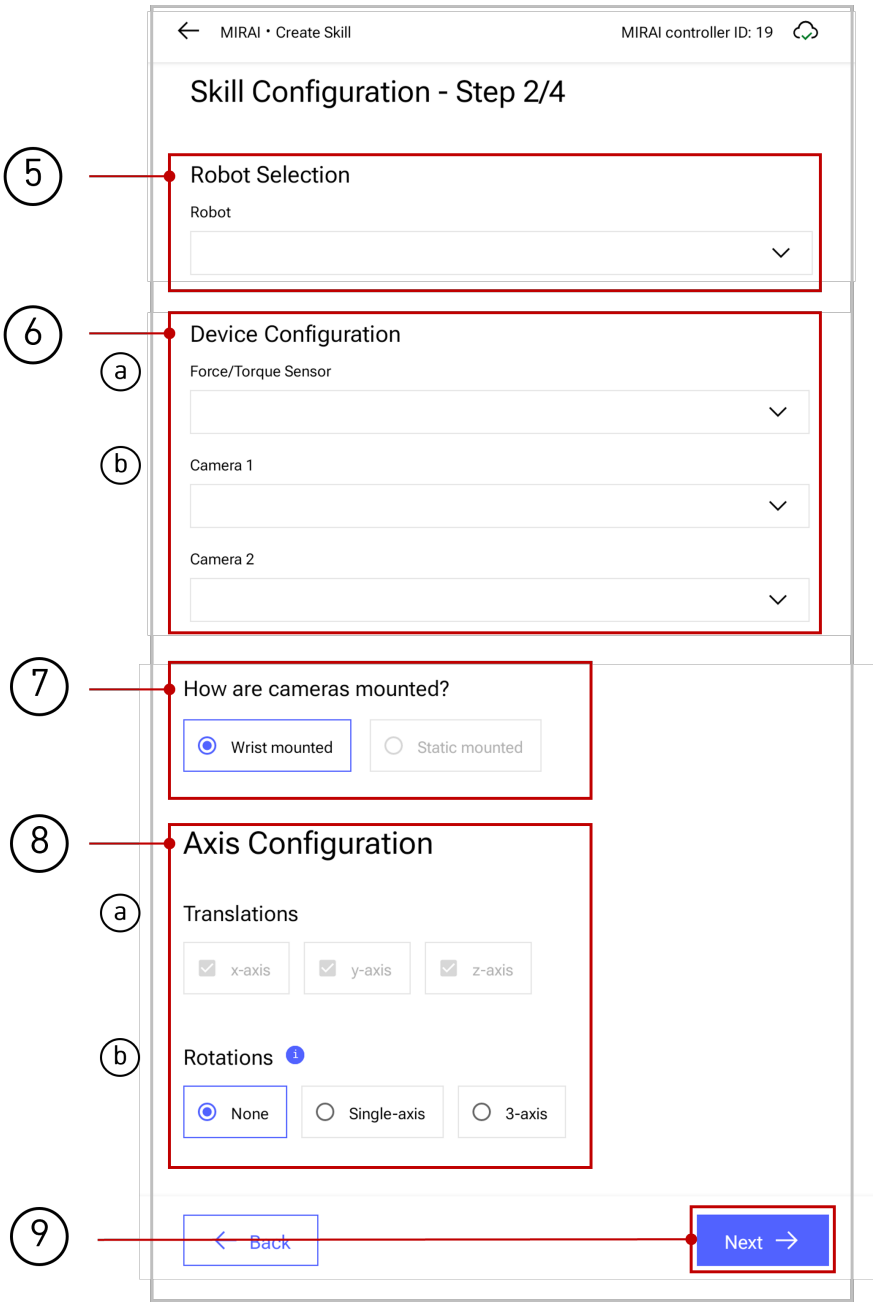
   b. Under **Rotations:**

   POSITIONING SKILLS: All 3 rotational axes can be enabled.

   MULTI-TARGET POSITIONING SKILLS: All 3 rotational axes can be enabled.

   MOTION SKILLS: Only 1 rotational axis can be enabled.

   TIP: For stable and efficient trajectories, enable all axes only when necessary.



If rotations are enabled, additional fields will appear to specify the tool and rotations around the tool center point (TCP):

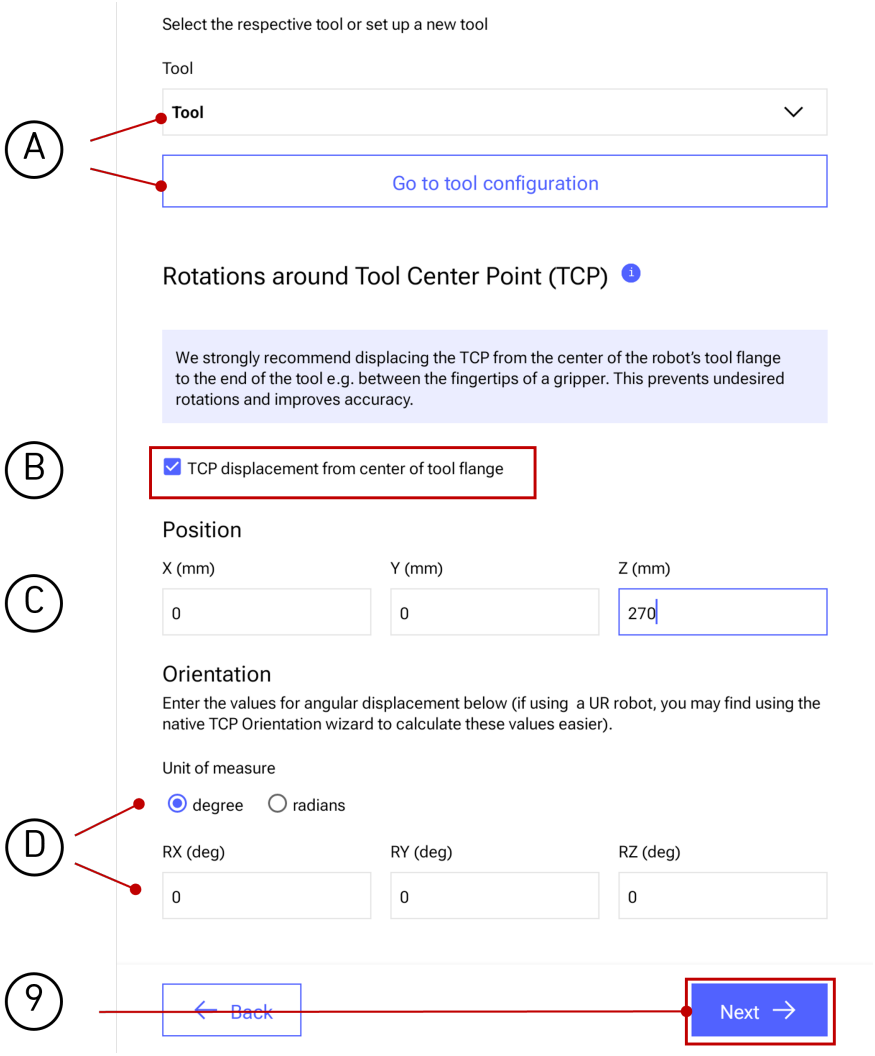
A. To specify the tool in use, select an existing tool from the "Tool" drop-down menu, or tap **Go to tool configuration** to set up a new tool (see 4.3 Configure a tool for instructions).

B. To prevent undesired rotations and improve accuracy, we strongly recommend displacing the TCP from the tool flange center to the tool tip. Tick the check box **TCP displacement from center of tool flange**.

C. Under **Position**, define the position of the TCP in space: Enter the **X**, **Y**, and/or **Z** displacement values in mm.

In the screenshot showing the example DIMM insertion skill, the TCP is displaced 270 mm along the Z-axis to the gripper tip.

D. Under **Orientation**, define the orientation of the TCP in space: Select **degrees** or **radians**, then enter **RX**, **RY**, and/or **RZ** to specify the TCP displacement relative to the tool flange center.



9 Tap **Next** to proceed to the next screen, "Camera Settings."

10 **For the first skill on this setup:** Adjust the focus and aperture **on the camera lens:**

- The aperture ranges from 1.4 to 16. We recommended setting aperture to 8.
- Adjust the focus of the lens so that the tool tip is in focus in the camera feed. If using two cameras, both feeds will be visible. Camera 2's feed appears as a small image. Tap the small image to expand it and adjust the settings.

In the screenshot, the black gripper holding the DIMM is centered at the bottom of the image, just above the yellow DIMM slot.

WARNING

Camera lens settings are set **once** for all skills on a single setup. Changing them after training will cause the trained skills to stop functioning.

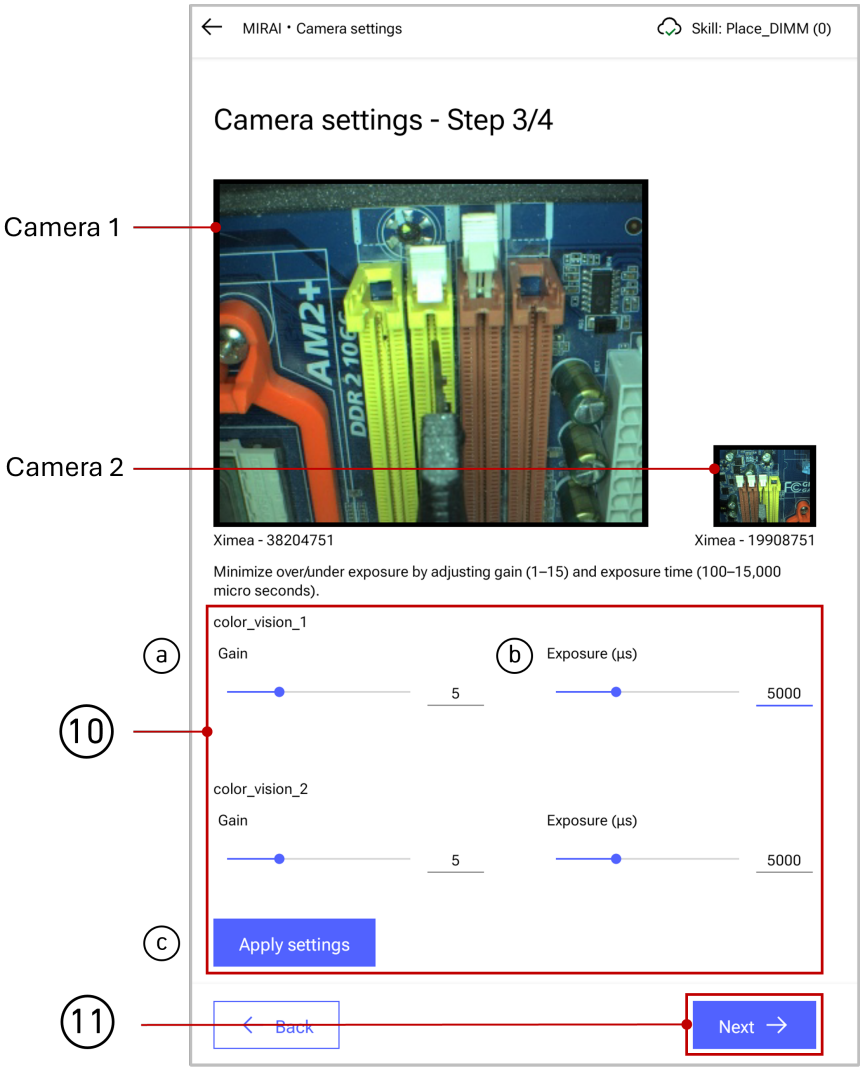
For all skills: Adjust the gain and exposure time in the MIRAI Training App. Move the robot through its expected trajectories to find settings that provide a good lighting range and sufficient contrast.

a. Adjust the camera gain using the slider.

b. Adjust exposure time using the slider.

If using two cameras, adjust the gain and exposure for Camera 2. Camera 2's feed appears as a small image. Tap the small image to expand it, if needed.

c. Tap **Apply settings** to update the camera images.



⑪ Tap **Next** to proceed to the last screen, "Reference Position."

⑫ Move the robot to the *reference position* and tap **Save position**. You can modify the reference position as needed during the recording workflow.

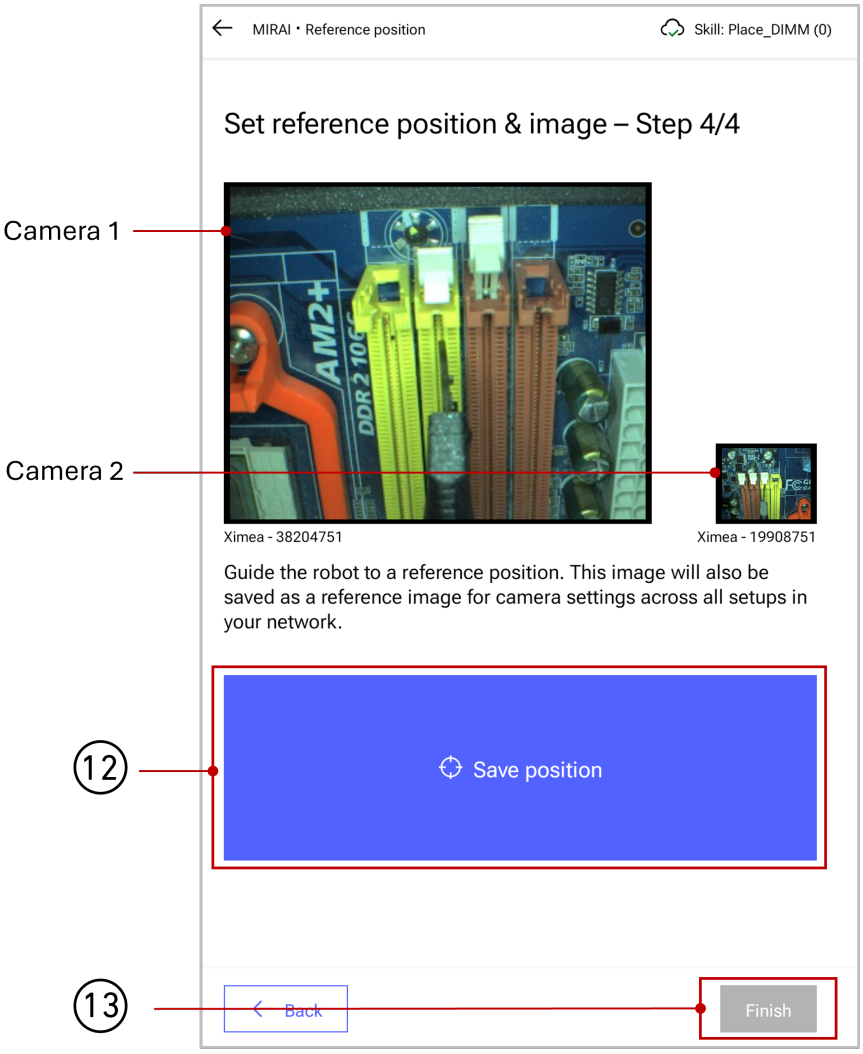
Further information: Reference position

The reference position is a user-defined point that allows the robot to return to a consistent location when recording episodes or testing skills. It is a flexible tool for consistency, not a strict requirement. Use it to enhance the reliability and comparability of your recordings or tests, for example as a reset point or starting marker.

When selecting a reference position, consider:

- **Workspace safety:** Ensure minimal interference with surrounding equipment
- **TCP and range of motion:** Maximize the robot's flexibility and reach for planned tasks
- **Your use case:** Define priorities, such as unobstructed initial movements or efficient setup between recordings or tests.

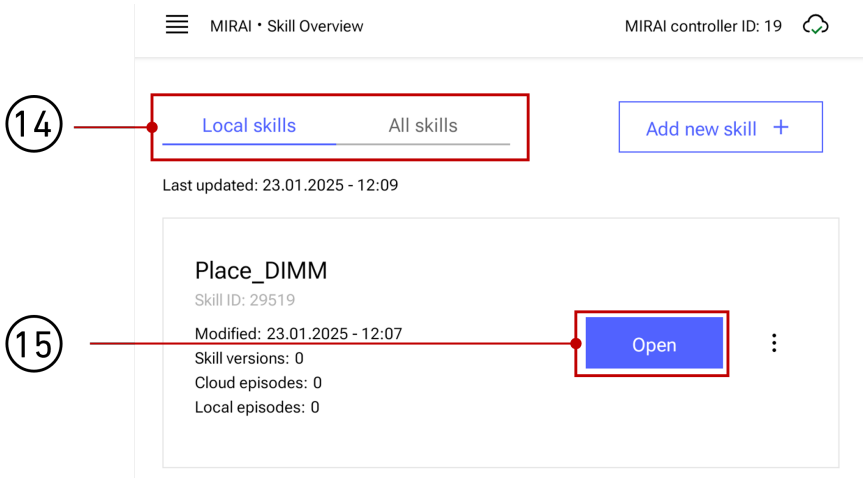
  MULTI-TARGET POSITIONING SKILLS: Select a reference position that shows the entire scene and all potential targets, allowing MIRAI to identify the nearest one.



13 Tap **Finish** to create the skill.

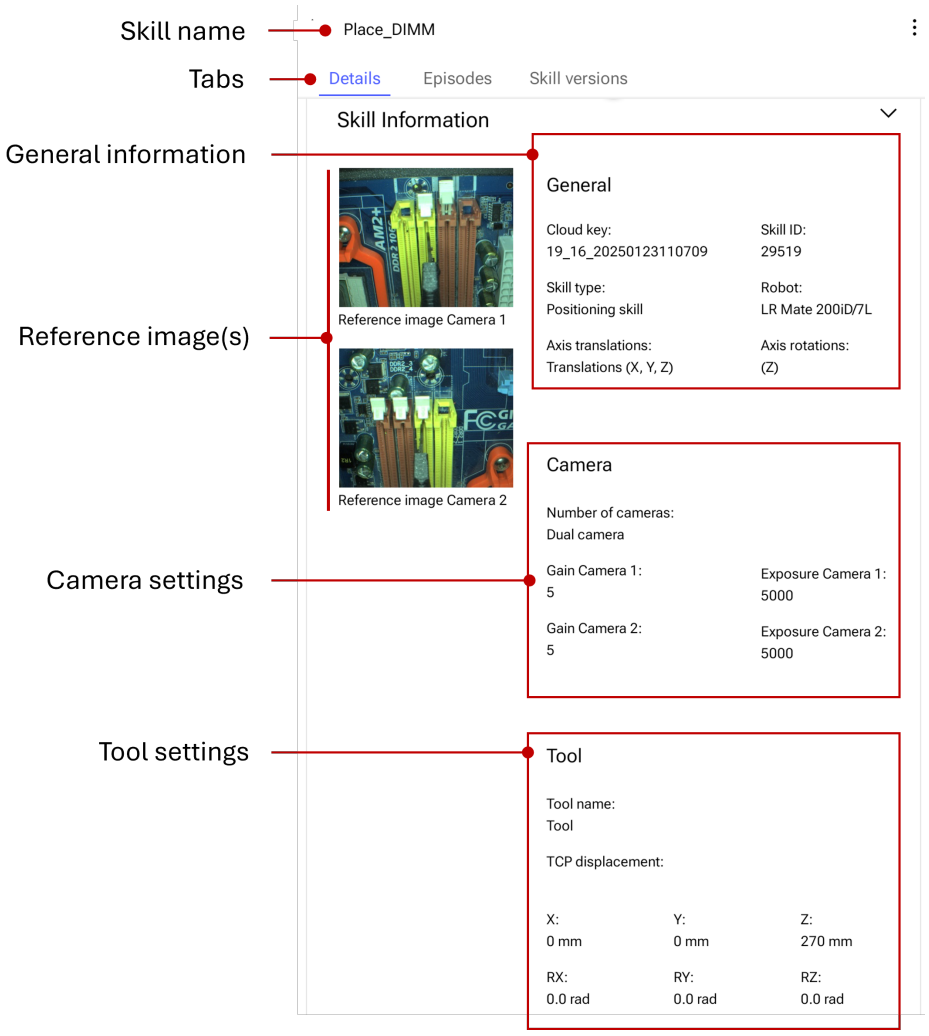
⑭ **Confirm skill creation:** From the main menu, tap **Skill Overview**. Confirm that the skill you created appears under both the **Local skills** and **All skills** tabs.

- **Local skills** are all skills created on the MIRAI Controller you are currently using and skills you have previously synchronized from other controllers in your network.
- **All skills** includes all skills across all MIRAI controllers in your network. These include skills that are available for synchronization and all local skills.

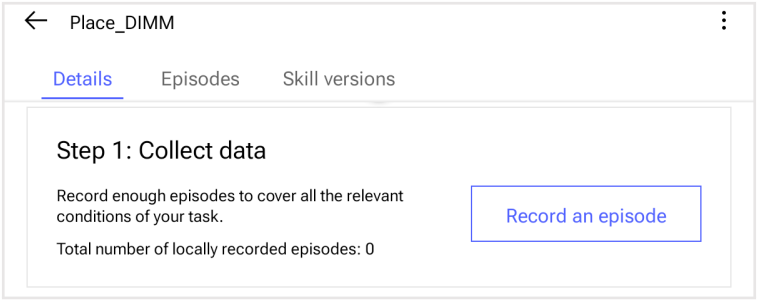


⑮ **Check skill details:** Tap **Open** to access the **Details** screen for the new specific skill. Verify that the following information is correct:

- Under **General**, check the skill type, robot model, and axis rotations and translations.
- Under **Camera**, check the number of cameras and the gain and exposure settings.
- Under **Tool**, check the TCP displacement values.
- Inspect the **reference image** previews:
  - Check that the tool tip is fully visible and ideally aligned parallel to the camera's view.
  - Check that the target object(s) are in view.
  - Check that unrelated objects, such as hands or miscellaneous parts, are excluded from the camera's field of view.
    NOTE: You can modify the reference position as needed during the recording workflow.



You can now start collecting data. Scroll down to the first section below the skill details, and tap **Record an episode**. For guidance on recording data, refer to the section for your skill type.



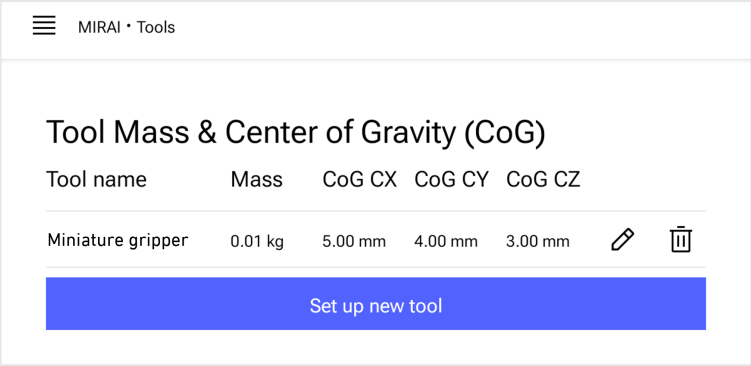
4.3 Configure a tool

Tool configuration is only required when using a force/torque sensor. Follow the steps below to manually enter tool information or use the measurement wizard to estimate the center of gravity (CoG).

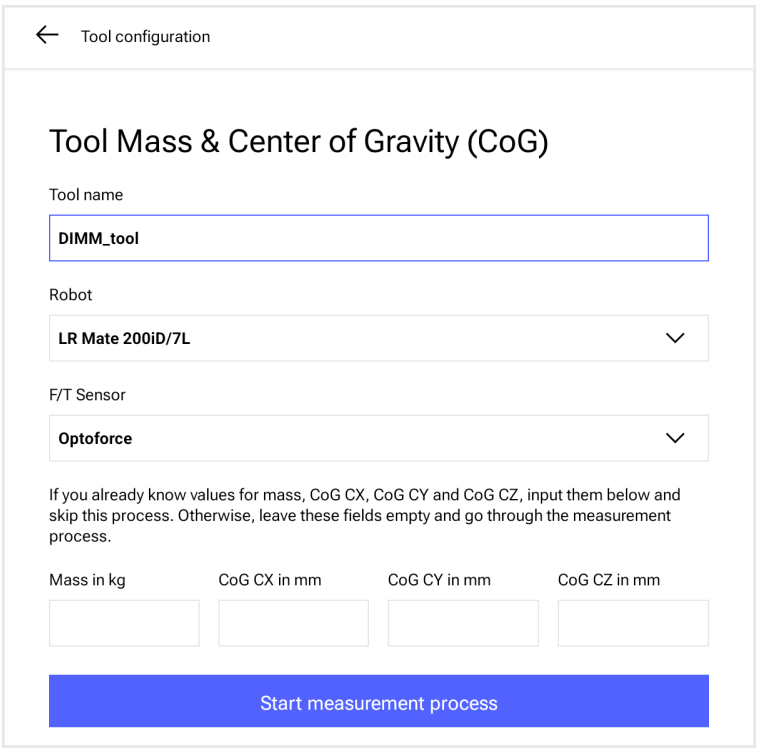
Before configuring a tool, activate tool rotations and ensure force/torque sensor is configured, connected, and powered on.

1. **Confirm prerequisites:**
   - Activate tool rotations for the skill
   - Ensure a force/torque sensor is configured, connected, and powered on.
2. **Open the Tool Configuration screen** (two options):
   - Expand the main menu and tap **Tool Configuration**
   - Access the screen from the "Create Skill" section.
3. **Add or modify a tool**
   - To add a new tool, tap **Set up new tool**.
   - To modify an existing tool, tap ✎. A pop-up message will ask for confirmation. Tap **Yes, edit**.

> In the screenshot, one tool – "Miniature gripper" – has already been configured. If no tools are configured, the table will be empty.



4. **Enter tool name and select robot:**
   a. In the **Tool name** field, enter a descriptive name for the tool. Choose a name that clearly describes the tool's function for all users.
   b. From the **Robot name** drop-down menu, select the robot in use.
   c. From the **F/T sensor menu**, select the force/torque sensor in use.



5 **Enter tool information** (two options):

Manually:

a. Enter **Mass in kg** and **CoG CX**, **CoG CY**, and **CoG CZ** in mm.

b. Tap **Save** to proceed.

Using the measurement wizard:

> ! **IMPORTANT:** MIRAI hand-guiding is deactivated. Move the robot based on your platform:
> **FANUC:** Jog the robot in T1 mode while holding the deadman switch.
> **KUKA:** Run MiraiAppControl on the smartPAD with the enabling switch and start key pressed. Ensure [S][I][R] indicators are green. See the MIRAI KUKA Integration Manual for details.
> **Universal Robots:** Use freedrive, holding the arm **above** the force/torque sensor.

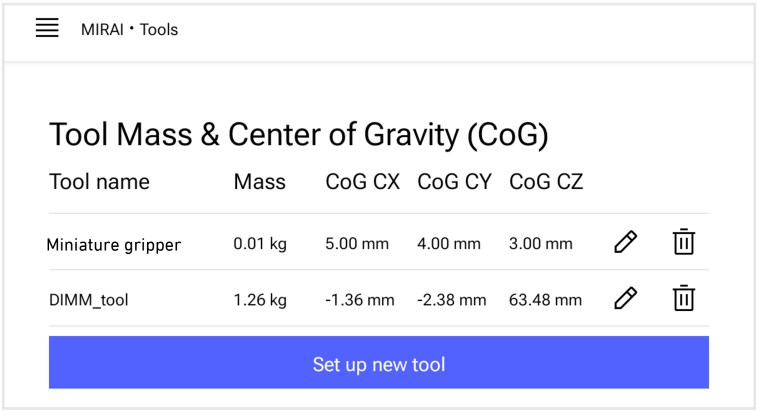
a. Tap **Start measurement process**. The app will guide you through positioning the arm in 6 different poses to obtain values for tool mass and CoG.

b. Read the instructions on the next screen, then tap **Proceed**.

c. **Measure Pose 1:**

- Position the robot's tool as shown on the screen, ensuring the joints' orientation matches the illustration.
- Wait 5–10 seconds for the force/torque sensor to attain a static state.
- Tap **Measure**. When the measurement is done, a green "Measurement successful" message will appear.
- Tap **Next pose** to proceed.

d. Proceed with the remaining poses in the same way, then tap **Finish**. The CoG estimates for your newly created tool will appear on the main screen.



To delete a tool:

- Tap 🗑. A pop-up warning will be displayed.
- Tap **I have read the warning and want to delete the tool**.
  NOTE: You can only delete tools not used in a skill.

5 Classifiers

Classifiers categorize images into user-defined groups or "classes." Based on the outcome of classification, specific robot actions can then be performed. This makes it possible to make a decision in real time about what action to perform in the robot's workspace.

Read Section 5.1 to verify that your setup meets all requirements.
Read Section 5.2 to learn what MIRAI classifiers can do and how to use them with skills.
See the remaining sections for instructions on how to create, train, and duplicate classifiers.

> Look for **TIPS** for practical advice to help you work efficiently.
>
> Look for **IMPORTANT** for must-read reminders to avoid pitfalls.
>
> Look for **info boxes** for technical insights and best practices to improve your results.

5.1 System requirements and constraints

The following hardware and connectivity requirements must be met to train and deploy a classifier:

Camera requirements

- Only **one camera** is used for a classifier, though your setup may include multiple cameras.
- Only **USB cameras** are currently supported (GigE camera support is planned for future updates).
- **Camera replacement is not supported**. If a camera fails, you must create a new classifier. Alternatively, you may use another controller to duplicate the classifier – first to the new setup, then back to the original setup.

Network connectivity

- **During creation:** MIRAI controller requires internet connectivity via LAN, and the camera must be connected to the MIRAI controller.
- **During training:** Robot LAN connection to the MIRAI controller is not required.
- **During deployment in the robot program:** Robot LAN must maintain continuous connection to the MIRAI controller.

Status code reporting using MQTT and/or OPC UA

- **Classification status codes are not available**. When MQTT/OPC UA reporting is activated, the status remains fixed at 255 ("Normal, ready for requests") during classifier deployment.

5.2 Overview of classifiers and their integration with skills

What classifiers do

Classifiers sort items into predefined classes by recognizing key features and assigning specific labels. A common example is email spam filtering, where messages are labeled as either "spam" or "not spam."

MIRAI classifiers are trained to identify two classes. When paired with skills, classifiers enable a robot to make intelligent, real-time decisions about which actions to take. Instead of following a fixed sequence of commands, the robot can observe its environment, classify what it sees, and then perform the appropriate action. Some use cases are shown below.

Example use cases in factory automation:

How classifiers are implemented with skills

The classifier output acts as a decision point in the robot program. Classifiers return one of three outputs:

- **Class 1** (User-trained category)
- **Class 2** (User-trained category)

If MIRAI sees...	Take this action
Part orientation: Right-side up vs. upside down	Execute gripping vs. turning skill
Workpiece color: Red cable vs. yellow cable	Execute specific assembly skill
Workflow stage: Item inserted vs. button pressed	Execute next action in the workflow
Quality control result: Conforming vs. defective	Move to next item vs. execute action

- **Neither class** when the camera view doesn't match either of the trained classes. The Neither output acts as a safeguard, allowing you to define protective actions to safely handle unexpected situations.

Based on the classifier output, the robot follows a corresponding branch of logic, similar to an extended IF–THEN structure. For each outcome, you can define the robot's response, such as:

- Executing a specific skill
- Moving to a designated position
- Waiting or pausing
- Requesting human assistance

For detailed instructions on implementing classifiers and skills in robot programs, see the MIRAI Integration Manual for your robot platform.

5.3 Prepare a recording plan for a classifier

Preparing a recording plan can help make training more efficient and ensure your classifier reliably handles variations during production. MIRAI learns by recognizing patterns across diverse video examples ("episodes"). This means your classifier will only perform well in scenarios similar to what it sees during training. Your role is to show MIRAI the range of expected operating scenarios. For your recording plan, consider aspects like:

- Variance
- Where the workpiece appears
- The lighting conditions
- The skill(s) which precede or follow (for example, the gain and exposure set for the skill(s) should be the same as those set for the classifier)

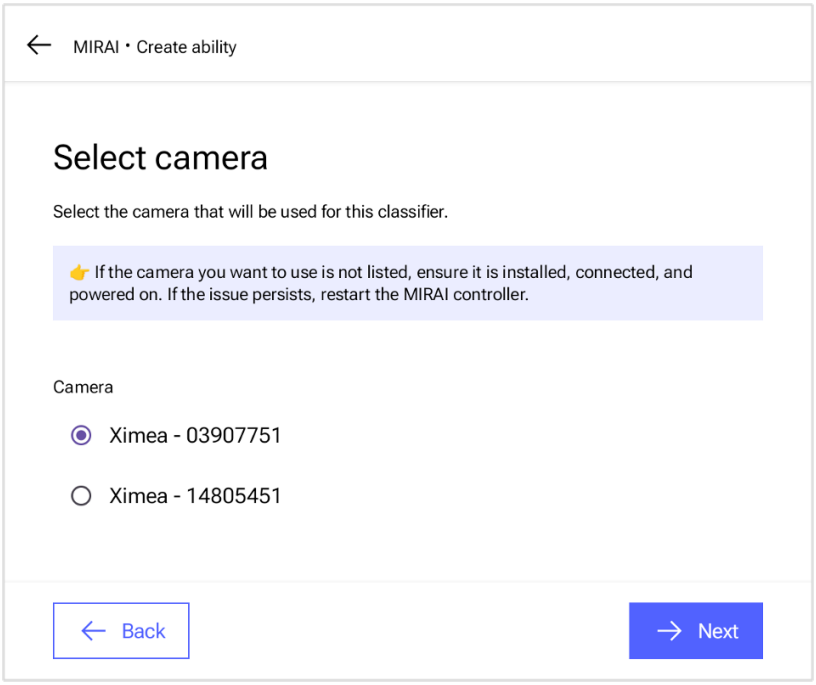
5.4 Create a classifier in the MIRAI Training App

1. **Verify cloud and camera connection:**
   a. Ensure that MIRAI is connected to the Micropsi cloud via the LAN.
   b. Ensure that the camera is connected to the MIRAI controller.
2. **Open Ability creation:**
   a. Open the main menu (☰), then tap **Ability overview**.
   b. In the upper right corner, tap **+ Add new ability**.
3. **Enter an Ability name and select "Classification":**
   a. Choose a name that clearly describes what the classifier does.
   IMPORTANT: Classifier names cannot be changed after creation. To use a different name, delete the existing classifier and create a new one.

b. Under **Type of ability**, tap **Classification** to set the ability type, and then tap **Next**.

④ **Select the desired camera using the radio buttons.** Only one USB camera can be used to train the classifier. Tap **Next** to proceed.

- If you have two cameras, select one.
- If you have only one camera, it will be preselected.



QUICK FIX

If your camera does not appear in the list, follow these steps:

1. Leave the Ability creation process by tapping the ← arrow at the top left corner. On the pop-up *"Leave ability creation?"*, select **Yes, leave**.
2. Check that the camera is connected and powered on.
3. Restart the MIRAI controller.
4. Start the classifier creation process and select the desired camera.

⑤ **Adjust the camera settings:**

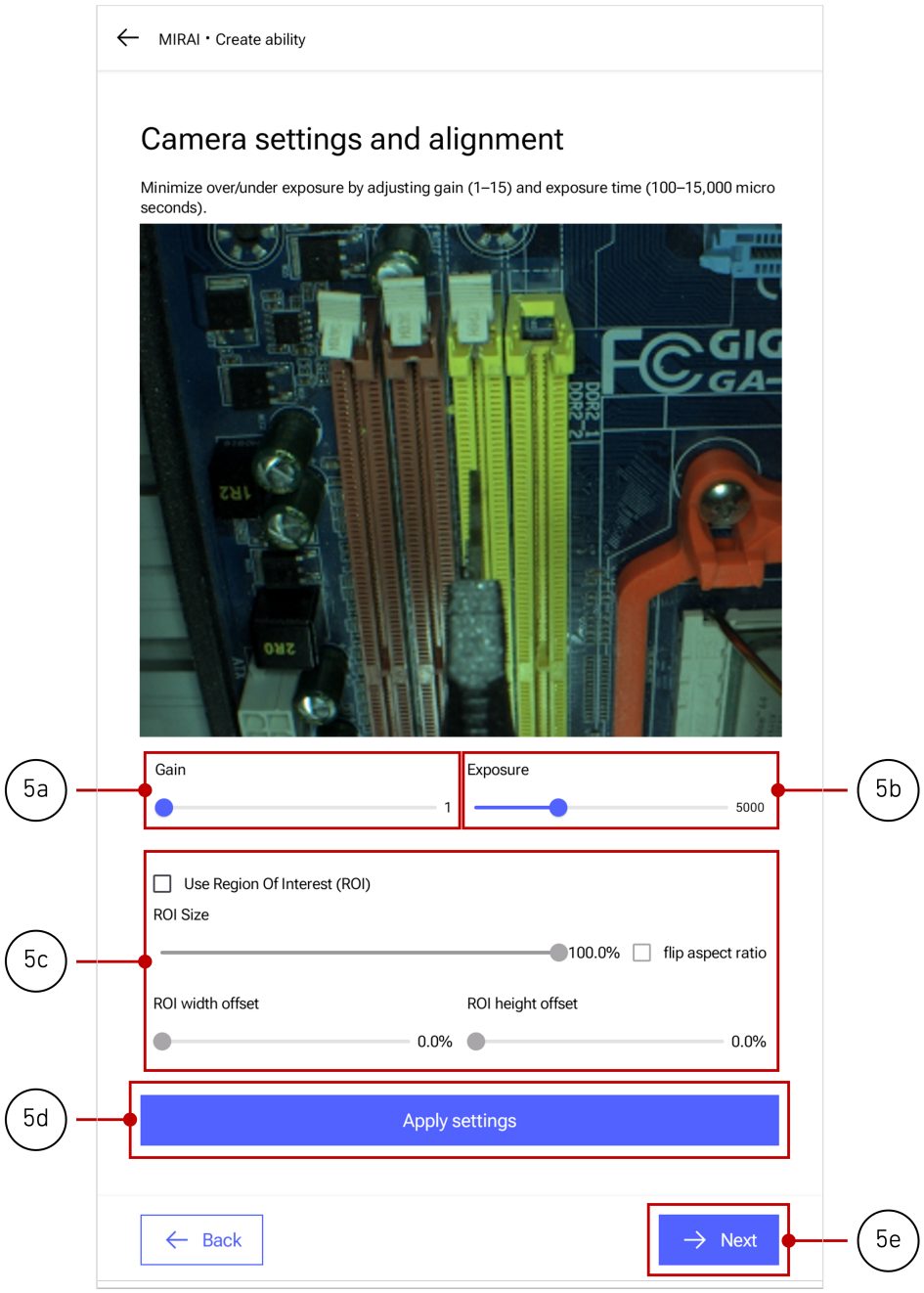
a. Adjust the gain using the **Gain** slider.

b. Adjust the exposure using the **Exposure** slider.

c. OPTIONAL: **Use Region of Interest (ROI)** to define a specific portion of the camera's field of view that MIRAI will use for classification. See Section 5.4.1 for more information on how to define your ROI.

d. Tap **Apply settings** to see the changes. You must tap this to both see and save the adjusted Gain, Exposure, and ROI.

e. Tap **Next** when the camera image shows a clear image.



6 **Label your classes:**

a. Assign a clear, descriptive name to each of the two classes you are distinguishing. Choose names that describe what each class represents. For guidance on creating meaningful class labels, see Section 5.4.2.

IMPORTANT: Class labels cannot be changed after creation. To use different labels, delete the existing classifier and create a new one.

b. Tap **Finish** to create the classifier. You can now record episodes to train the classifier.

5.4.1 Using and defining the Region of Interest (ROI)

The **Region of Interest (ROI)** is a user-defined portion of the camera's field of view that MIRAI uses for classification. Instead of analyzing the entire camera image, the ROI allows the classifier to focus on a specific region of the scene.

Why use a ROI

Using a ROI helps exclude irrelevant areas, objects, or background elements that should not influence the classification result. By focusing on a smaller and more relevant area, the classifier can learn finer visual details, and often achieve a more robust and reliable performance.

NOTE: Using and defining a ROI affects only the image region used for the specific classifier and does not change the physical camera setup.

How to define a ROI

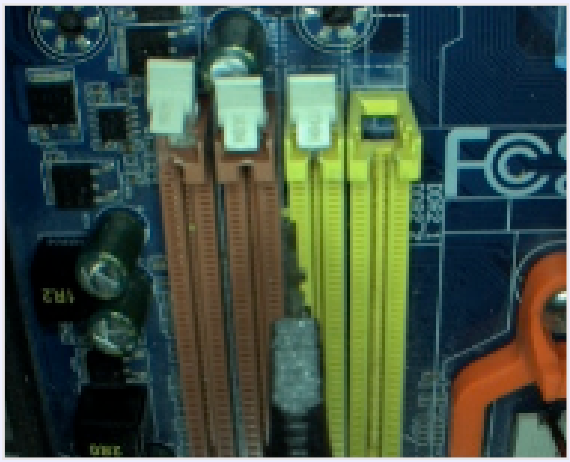
- ☑ **Use Region of Interest**: Check this to define the camera view that MIRAI should focus on. If left unchecked, the full camera view is used.
- ⚌ **ROI Size**: Use this slider to set how much of the camera's full view should be used for classification. Setting a smaller value zooms in on a centered region. (When "Use Region of Interest" is checked, the maximum ROI Size is 95%.)
- ⚌ **ROI Width Offset**: Use this slider to move the ROI to the left or right. Increasing the width offset shifts the ROI toward the right side of the camera view. On the camera image, this looks like objects move to the left.
- ⚌ **ROI Height Offset**: Use this slider to move the ROI up or down. Increasing the height offset shifts the ROI downwards. On the camera image, this looks like objects move up.

  NOTE: Depending on the camera mounting on the robot, the directions of the width and height offset can appear switched. We recommend experimenting with different offset combinations to determine the optimal ROI for your classifier. Remember to tap **Apply settings** after each adjustment to preview its effect on the camera view.

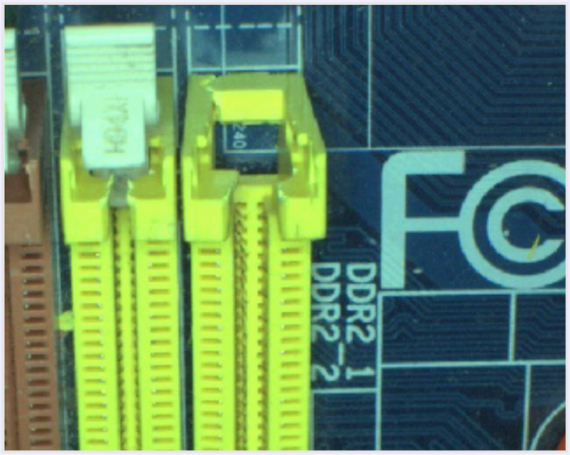
- ☑ **Flip Aspect Ratio**: Check this to rotate the camera view by 90° counter-clockwise (as displayed on the app screen). This only changes the viewing orientation and does not change the ROI itself; however, it does affect how the width and height offsets are perceived on screen.

Example: Using ROI to focus on relevant elements (DIMM Insertion)

In this example, the goal is to train a classifier that determines whether the DIMM tab on the yellow slot (second from the right) is *open* (Class 1) or *closed* (Class 2). The classifier is paired with a skill that inserts a DIMM into the yellow slot when the tab is open



Camera view, *no* ROI: The camera sees all four DIMM slots and the surrounding environment. Changes in the red slots, which are irrelevant to the task (for example, a tab is closed in one recording and open in another), can influence classification results during training and execution, and may require additional training to compensate.



Camera view, *with* ROI: The ROI Size is set to 43.78% and the Width Offset set to 79.85% in this view. The classifier can focus on the relevant yellow DIMM slot's tab, while excluding as much of the irrelevant areas as possible. As a result, changes in the red slots' tabs no longer affect classification results.

5.4.2 Creating descriptive class labels

Choose class labels that clearly communicate their meaning to all users. Your labels can reflect different aspects of the classification:

1. **Part characteristics:** Distinguish parts using their key features
   ✔ Recommended: *brown cable* and *blue cable*
   ✘ Not recommended: *cable1* and *cable2*
2. **Decision/Action:** Highlight the specific choice or next step in the task
   ✔ Recommended: *choose-position_grip* and *choose-position_turn*
   ✘ Not recommended: *position1* and *position2*
3. **Outcome/Result:** Emphasize the final state or evaluation
   ✔ Recommended: *DIMM_open* and *DIMM_closed*
   ✘ Not recommended: *yes* and *no*

Best practices:

- Make names easy to distinguish
- Use clear and specific words
- Align names with the task's purpose

5.5 Train a classifier

You will record multiple videos – called *episodes* – to show the classifier examples of each class. The recording process has two iterative steps:

1. Record episodes.
2. Evaluate classifier performance using two metrics.

Repeat this process until you are satisfied with how accurately the classifier identifies each class.

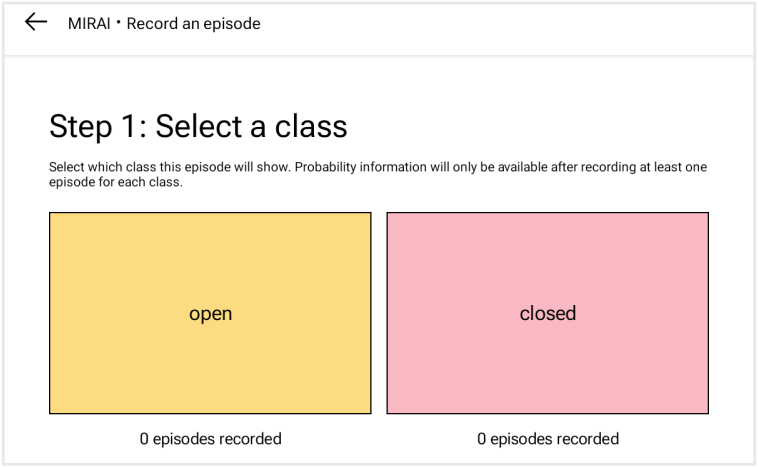
5.5.1 Record one episode for each class

① **Open the recording screen:**

a. Tap **Ability Overview** in the main menu and select the classifier you want to train.

b. Tap **Record an episode**.
IMPORTANT: MIRAI must be connected to the LAN to save episodes. Verify the connection before recording.

② **Select the class you will record** by tapping that class.
If you select the wrong class, tap the **Back** button and reselect the class.



③ **Record the episode:**

a. Tap **Start recording**.

b. Show different views of the scene or workpiece using one of two methods (see Quick Guide):

- Jog the robot with the teach pendant.
- Move the object of the class by hand.

Quick guide: Recording data for classifiers

Training a classifier is different from training a skill. Instead of using precise movements around the target, use loose free-form movements to quickly show different views of the object or scene.

There are two methods to record data:

Method 1: Move the *camera* around the scene
Use the teach pendant to move the robot's wrist-mounted camera around the object or scene. Hand-guiding is unavailable because MIRAI does not control the robot during this process.
→ Choose this method for scenes or large objects.

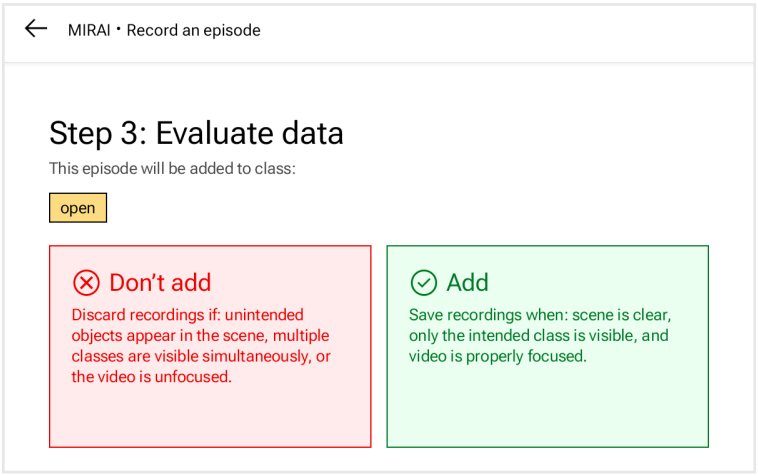
Method 2: Move the *object* in front of the camera
Grab the object and move it around the scene to show different perspectives. Your hands can appear in the frame.
→ Choose this method for objects that are easy to move.

c. When you are finished, tap **Stop recording**.

d. Save or discard the episode:

- *To save the episode*, tap the green box that says "Add." A confirmation pop-up will appear: "Episode added successfully." Tap **Finish**.
- *To discard the episode*, tap the red box that says "Don't add." A confirmation pop-up will appear: "Don't add episode to classifier?" Tap **Confirm** to proceed.

TIP: Discard episodes that show both classes together.



④ Record an episode for the other class by following steps ② and ③.

⑤ **Proceed to** Section 5.5.2 to use the Probability and Accruracy metrics to guide further training.

5.5.2 Before you train: Understanding Probability and Accuracy metrics

When training your classifier, you will use two metrics to assess its performance: Probability and Accuracy. Both metrics are expressed as percentages, but each reflects a different aspect of the classifier's behavior:

1. **Probability (*per-frame confidence*)**

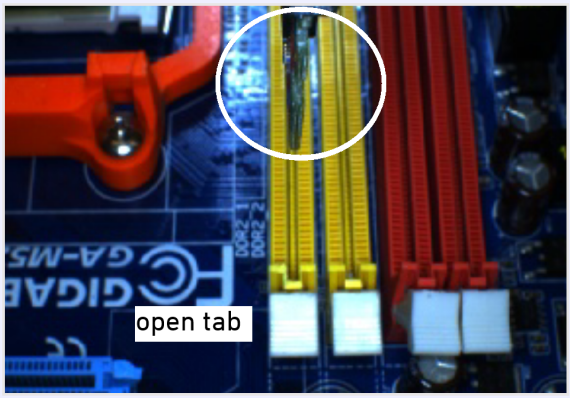
- *Probability* reflects the classifier's confidence that the current frame belongs to a specific class. Probability is displayed on the **Recording Screen**.
- Because it is based on a single frame, the probability value updates continuously as the camera view changes.
- Use this metric to guide your recording strategy. A low value (<50%) for the target class indicates that the classifier requires additional episodes for that class.

2. **Accuracy (*overall performance*)**
   - *Accuracy* is calculated using all recorded episodes and estimates how well the classifier will perform with new, unseen data. Accuracy is displayed on the **Details Screen**.
   - Accuracy is updated each time you record a new episode.
   - Use this metric to decide when the classifier is ready for testing in the robot program. A high value (roughly >80%) suggests the classifier can be tested.

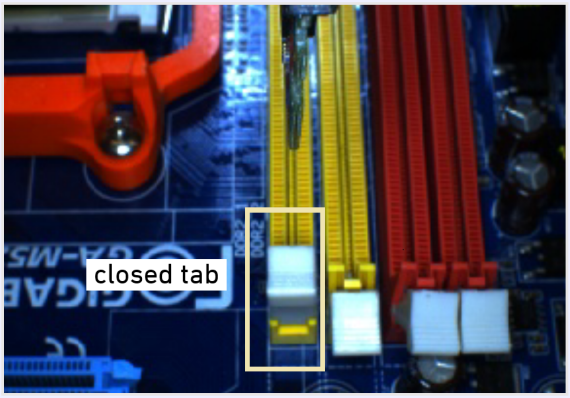
5.5.3 Use the Probability metric to guide training

After you record **one episode of each class**, use the Probability metric to guide training. This section provides step-by-step instructions and illustrates typical scenarios you may encounter during training.

This section uses screenshots from a classifier that identifies whether a DIMM tab is *open* (class 1) or *closed* (class 2). The classifier is paired with a skill that inserts a DIMM when the tab is open.



The tab is in the **open** position. The gripper holding a DIMM is visible in the white circle above the leftmost yellow slot.



The tab is in the **closed** position. The rectangle highlights the key visual difference between the open and closed classes in the two camera views.

Understanding the Probability display

The recording screen shows real-time confidence levels for each category (Neither, Class 1, Class 2) both as percentages and colored bars (gray, yellow, red respectively). High probability indicates greater classifier certainty about the category; low probability indicates less certainty.

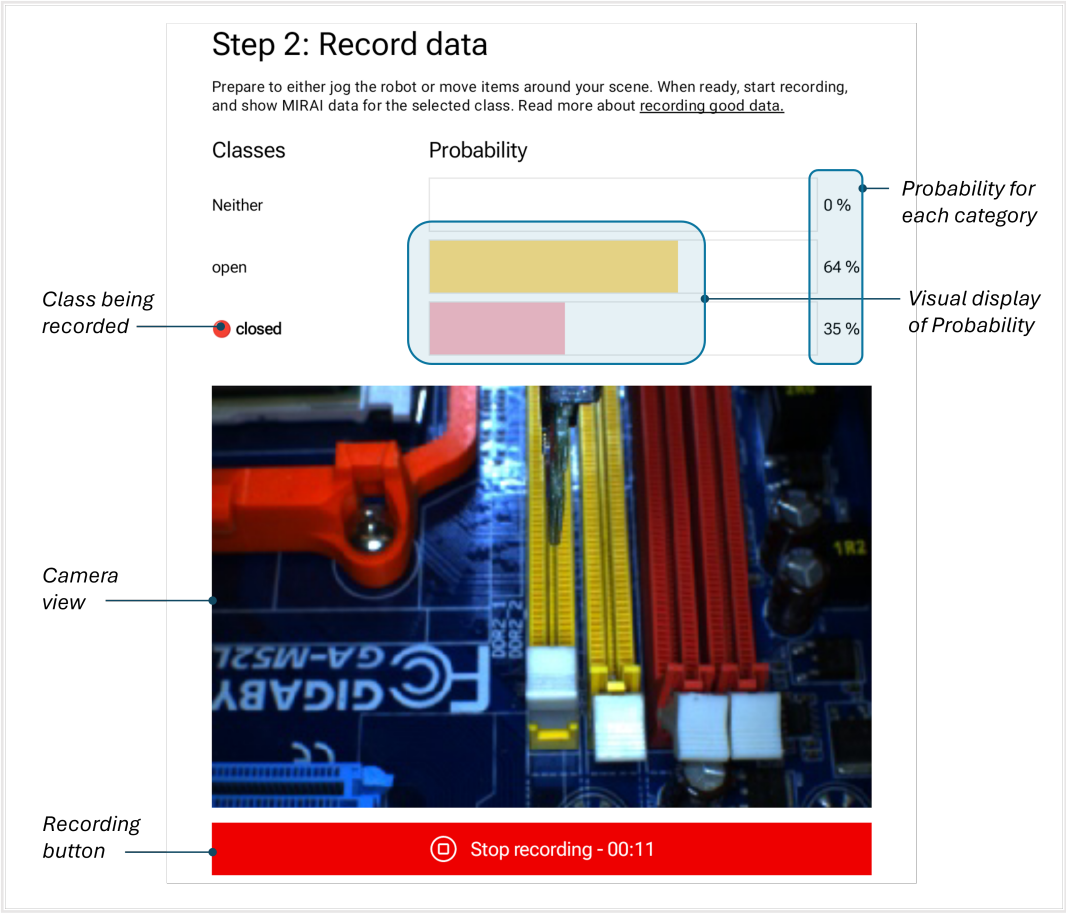
What to expect during early training

Do not be concerned if you see changing probabilities or high confidence for incorrect classes – this is normal and just indicates the classifier needs more data.

Overview of the Recording Screen

The Recording screen displays the Probability metric, which shows confidence levels for each category. These Probability values change continuously as the camera view shifts – this is normal behavior, particularly during your first few recording sessions.

NOTE: While multiple Probability values are visible on screen, data is only being recorded for one specific class. You can identify which class is actively being recorded by looking for the red dot indicator next to it.

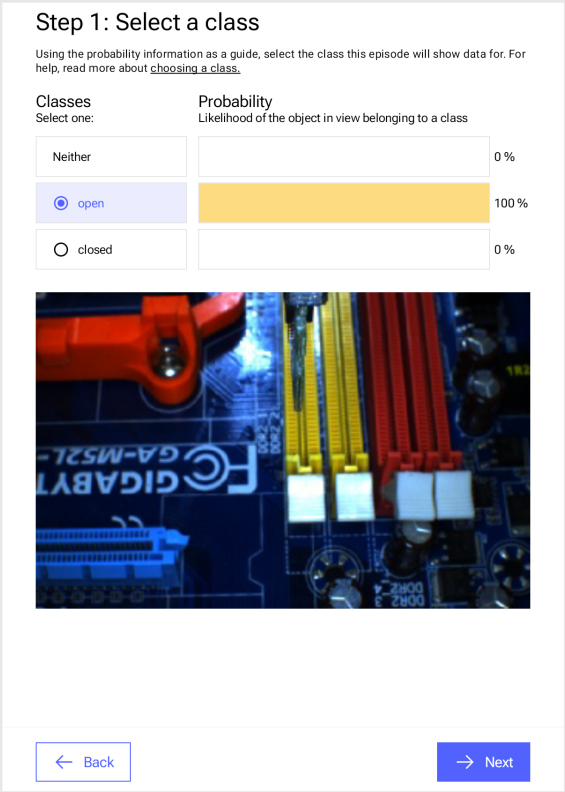


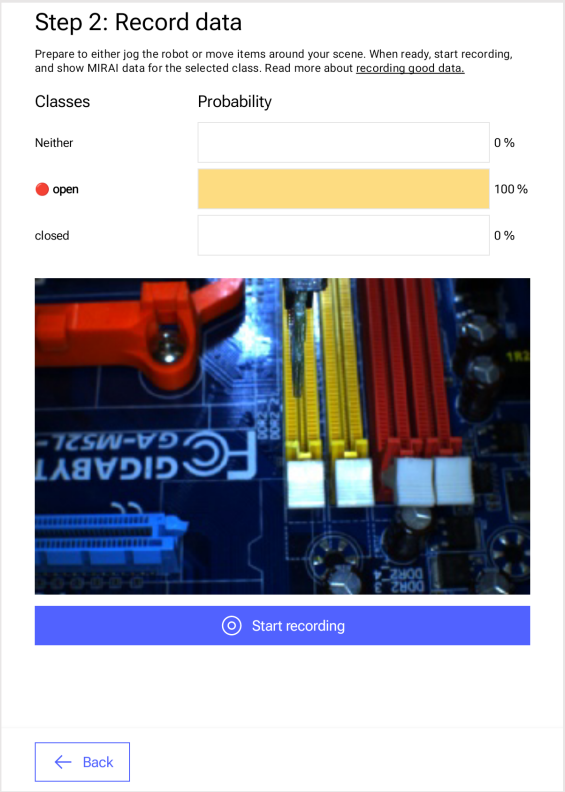
Step-by-step training process

The following steps are intentionally detailed to help you understand how the Probability metric behaves during training. This guided example is meant for first-time use. Later, you can apply the same principles more flexibly.

1. **On the workspace**, set up one class to record data for:
   - Position the object, or
   - Arrange the scene
2. **In the MIRAI Training App**, select that class by tapping the circular button next to its name.
3. Tap **Next** to proceed to the recording screen.

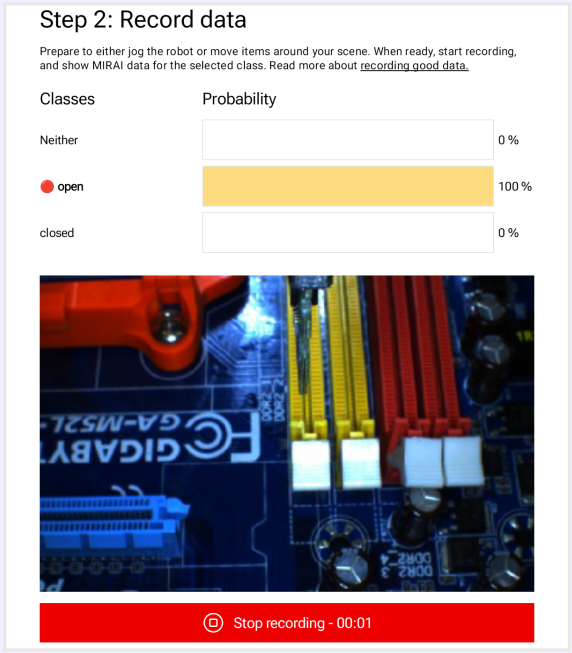
   → A red dot indicates the class selected.
4. Tap **Start recording**.



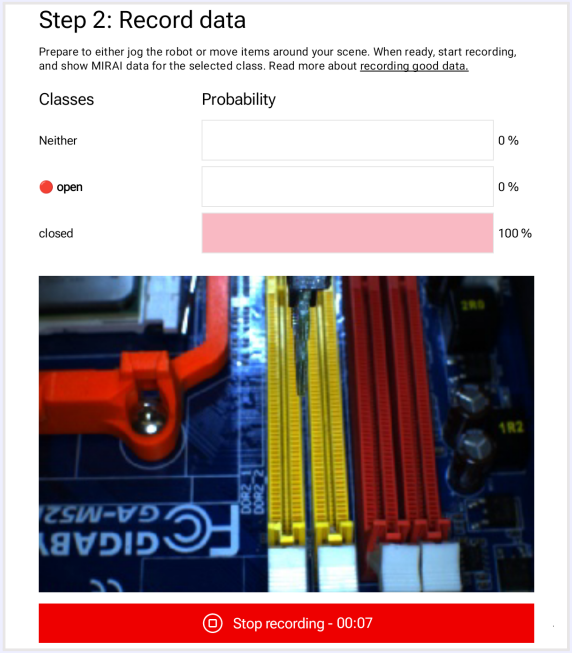


5. **Show different views of the target class:**
   - Watch the camera view while moving the robot or object.
   - Keep the distinguishing visual features of your class visible as much as possible. Don't worry if hands occasionally appear in the frame or if the object briefly moves out of view.
   - Aim for variety in your recording rather than length. Multiple short episodes (around 30 seconds) showing different perspectives and situations will train the classifier more effectively than one long recording of the same view. Track your recording time on the **Stop recording** button.
   - Remember that Probability values will fluctuate as the camera view changes.
6. Tap **Stop recording** to end the episode.
7. Save or discard the episode, to return to the Recording Screen.

Typical training situation: *Probability changes in real time as the camera view shifts.*
This is normal behavior. All images are being saved and labeled as a single class – indicated by the red dot – so the classifier learns to recognize what that class looks like.

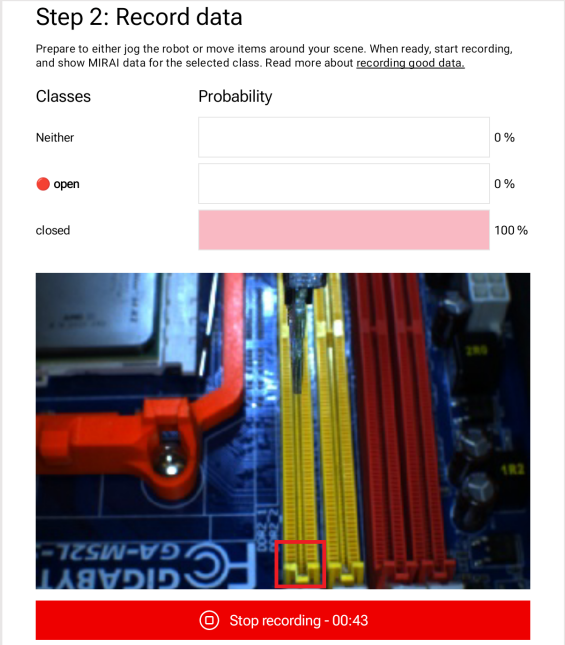


Time: 00:01

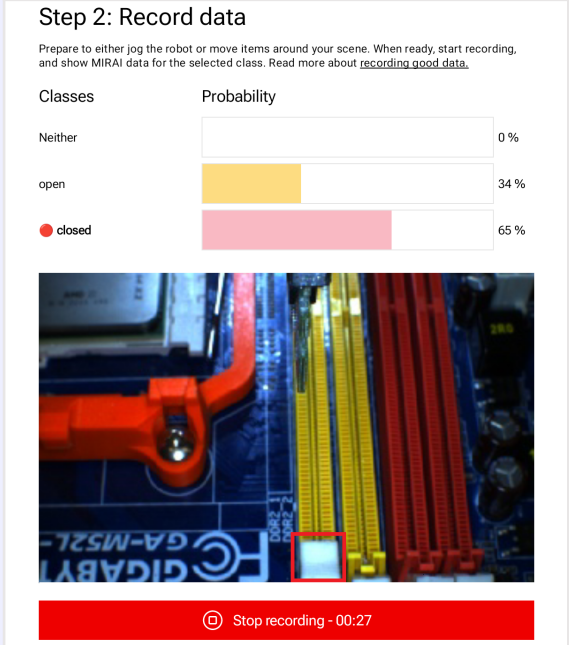


Time: 00:07

AVOID: *Key features out of view.*
When key class features disappear from view, the classifier relies on familiar background elements to make decisions. Here, the distinguishing feature (red box) occupies only a small portion of the frame, with most visual content identical across both classes. This produces unreliable results – either mixed probabilities or confident but incorrect classifications. Reliable detection of such subtle differences requires additional training data.



Open position – tabs are out of view: Confident but incorrect decision.



Closed position – only one tab in view: Mixed probabilites.

8 **Test classifier performance to identify what additional data is needed:**

a. On the **Select a class** screen, show different camera views of one target class and observe the Probability values.

b. Look for consistent errors – when Probability is highest for the wrong class or "Neither." Focus on testing scenarios that may not appear in your recorded episodes:

- Edges of the workspace
- Different viewing angles
- Various lighting conditions

Note any situations where the classifier makes mistakes, then record additional episodes focusing on these problematic scenarios. These errors reveal exactly what data your classifier needs to improve its overall performance.

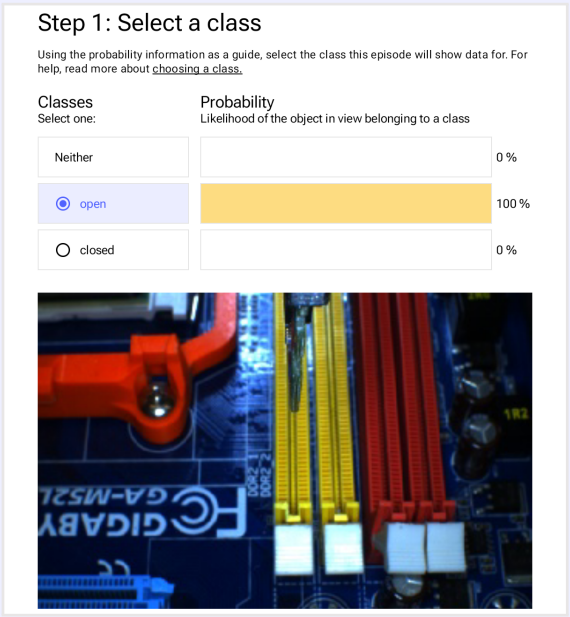
9 **Record more episodes and test again.**

- After adding several new episodes, episodes, check the Accuracy metric to determine if the classifier is ready for testing in the robot program. For details, see 5.5.4.

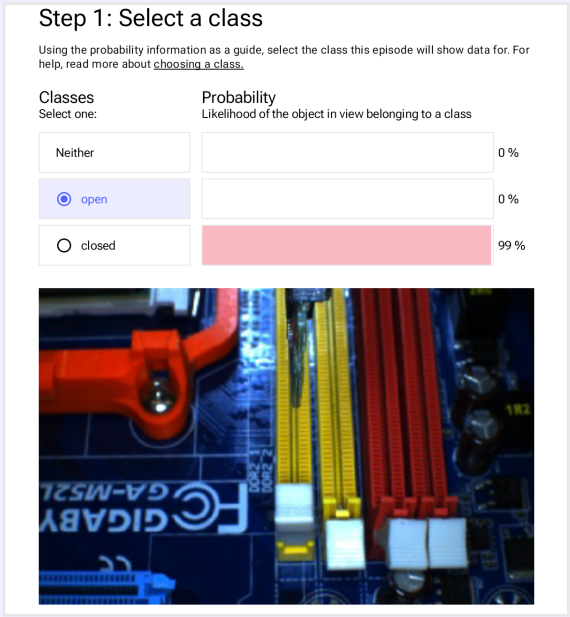
Testing situation: *Expected behavior*

The classifier correctly identifies the camera view. No additional recording is needed for these scenarios.

NOTE: The preselected class (shown in blue) does not affect testing. To record additional data for a class, tap that class and then **Next**.



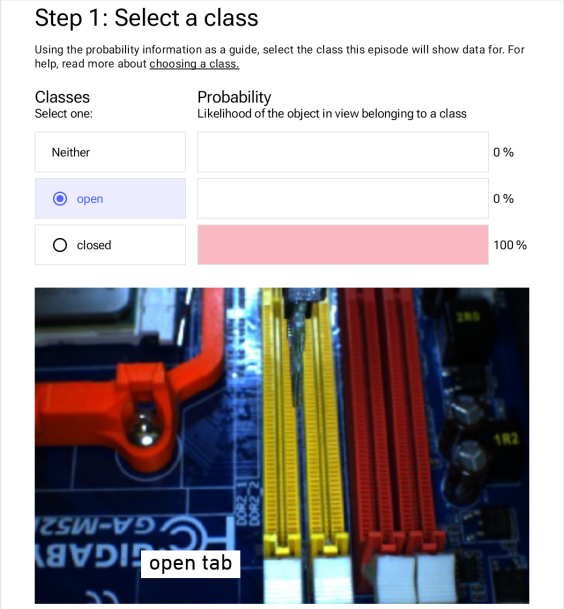
Open DIMM slot shown, classifier identifies open class.



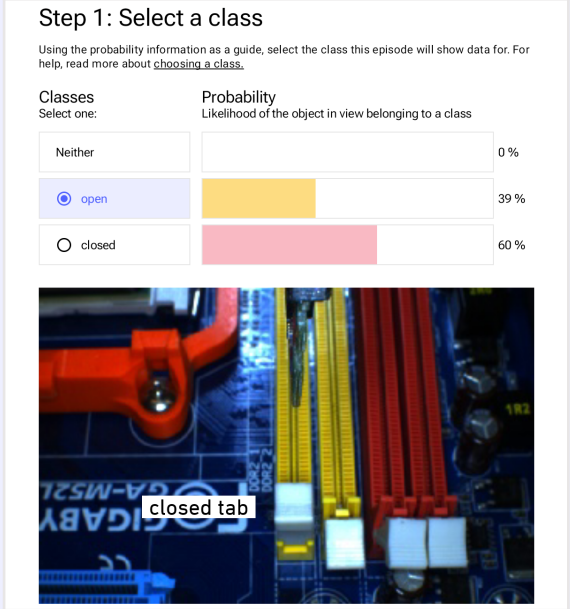
Closed DIMM slot shown, classifier identifies closed class.

Testing situation: *More data needed*

The classifier identifies the wrong class or shows mixed probabilities, indicating it needs additional training data for the current camera view.

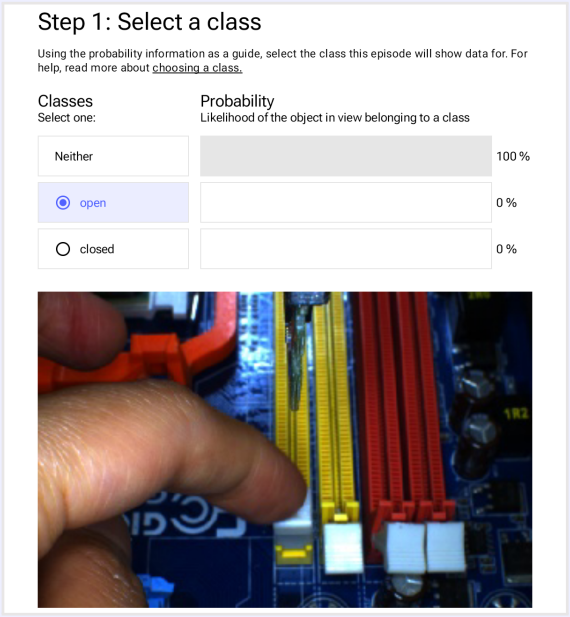


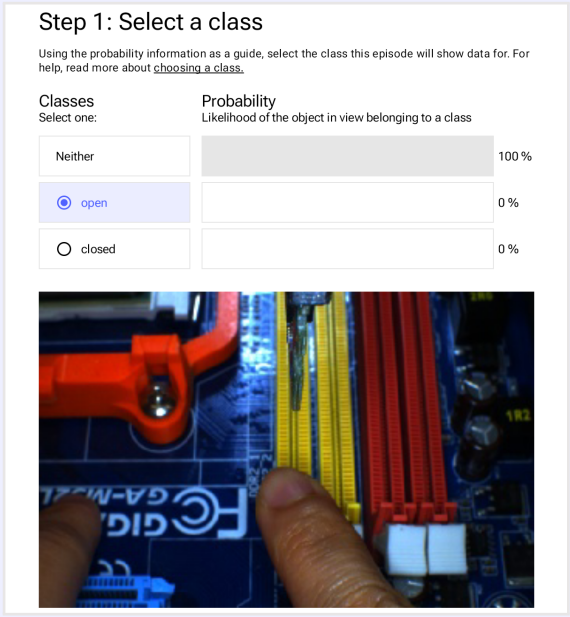
Incorrect identification at workspace edge: Record more open class views from this angle.



Mixed probabilities: Record more closed class views from this perspective.

Testing situation: *Highest Probability for Neither*
This occurs when the classifier encounters something unfamiliar. In these images, fingers prominently cover the key class features either partially (*left*) or entirely (*right*). A well-trained classifier should rarely output "Neither" – this category serves as a safeguard for unexpected situations.





5.5.4 Use the Accuracy metric to assess testing readiness

The Accuracy metric becomes available after you record two episodes for each class. It estimates how well the classifier will perform on new data and indicates when to begin testing in the robot program.

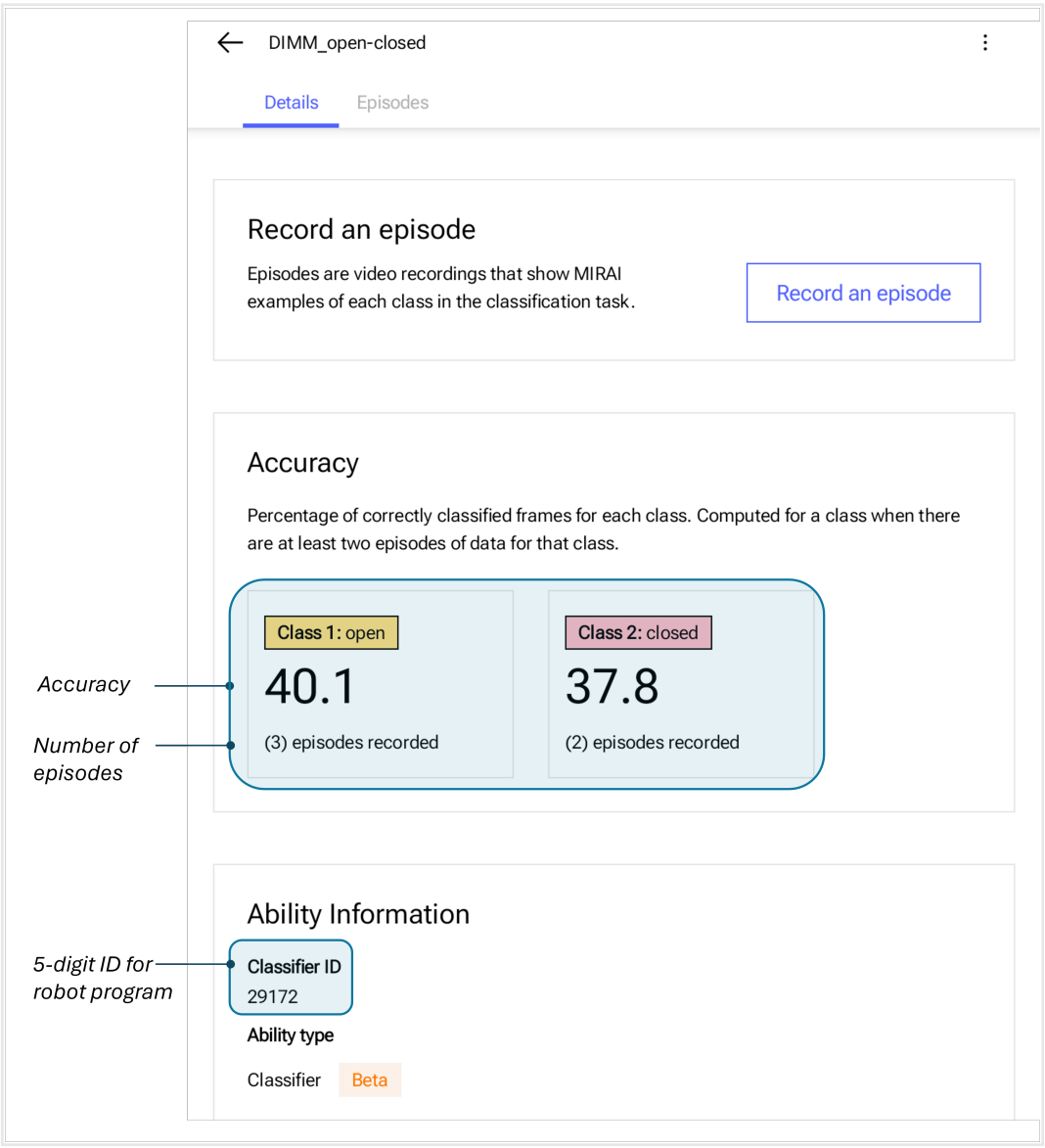
How to check for testing readiness
Use these flexible guidelines to assess your classifier:

- **Focus on Probability during training:** Use real-time Probability feedback while recording episodes to gauge immediate performance
- **Check Accuracy periodically:** After adding several episodes, review Accuracy as a general performance check (aim for >80% when possible)
- **No strict rules:** Your classifier may work well in your robot program even with lower Accuracy values
- **Trust your testing experience:** If Probability values consistently show correct classification during hands-on testing, the classifier is likely ready to be tested in the robot program – regardless of the exact numerical values.

Overview of the Accuracy metric

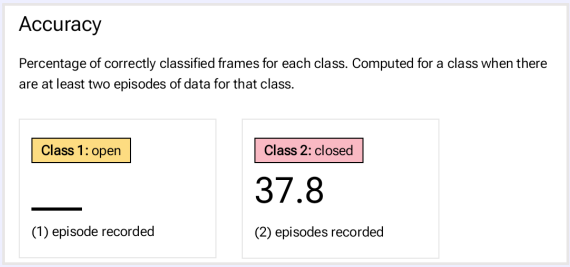
The **Details** screen displays the Accuracy metric for each class. The number of episodes recorded is shown in parentheses. Accuracy will update every time you add or delete episodes.

The 5-digit Classifier ID used to implement the classifier in the robot program is also found on the **Details** screen.

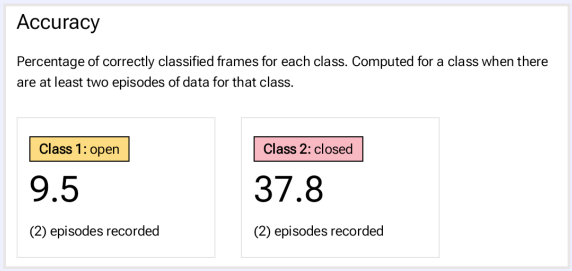


Typical scenario: *Accuracy improves with more episodes*

Accuracy typically increases as you add more episodes, and it is normal for different classes to show different values as your classifier learns.



Accuracy unavailable for Class 1 – more episodes needed.



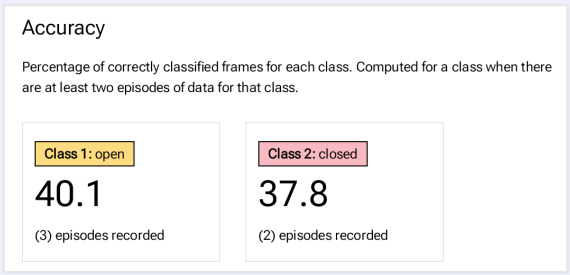
Accuracy available after 2 episodes per class.

Typical scenario: *Accuracy changes with more episodes*

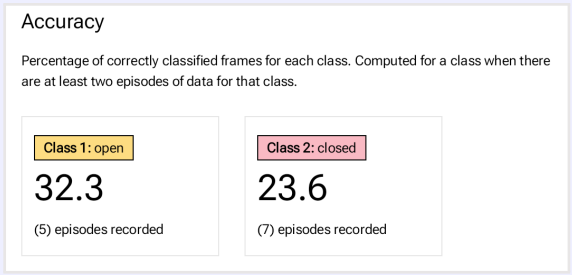
Accuracy typically improves as you record more episodes, but may temporarily drop when adding very different examples. This happens for two reasons:

Normal learning process: Accuracy drops temporarily when the classifier learns to handle more varied data (such as different lighting conditions). Continue recording episodes – accuracy will improve as the classifier adapts.

Poor training data: New episodes contain incorrect labels or poor-quality data (e.g., unexpected objects, both classes). Review and delete problematic episodes following the process in 5.5.5.



Class 1 Accuracy improves with an additional episode.



Temporary decrease for both classes after further episodes – this is normal when introducing divese data but can also occur when episodes are labeled with the wrong class or contain poor-quality data.

5.5.5 Review episodes to diagnose training issues

If the classifier's performance becomes unpredictable or its confidence drops, the issue may be caused by either mislabeled episodes or poor-quality training data.

- **Episodes labeled with the wrong class:** These are recordings where the label does not match what is shown in the camera view. These issues are usually easy to spot and should be deleted immediately.

- **Poor-quality data:** These episodes may include unexpected objects, both classes appearing in the same frame, or key visual differences being blocked. Identifying these issues requires watching the full episode carefully.

Since labels cannot be changed after recording, any problematic episodes must be deleted and re-recorded. This section explains how to review your training data to find and remove these issues.

① **Open the list of episodes:**

a. Go to the classifier **Details screen** by tapping **Ability Overview** and selecting your classifier.

b. Tap the **Episodes** tab at the top to see all episodes.

② **Locate the episode(s) you want to check:**

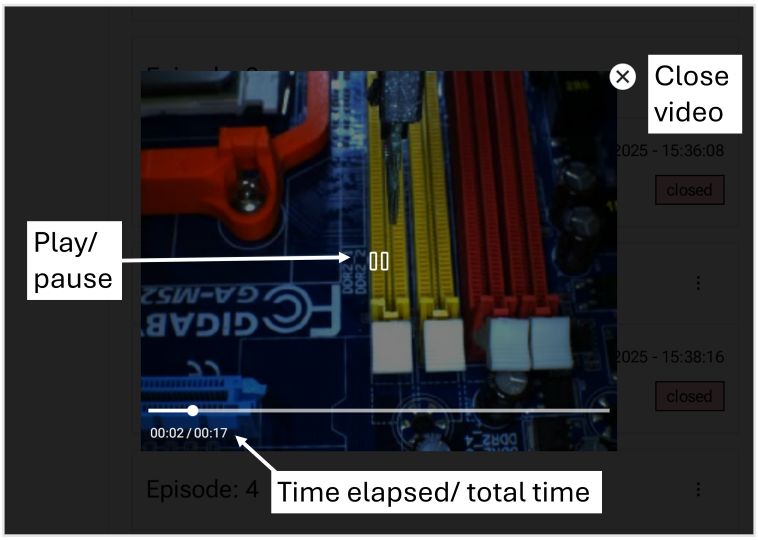
- Episodes are numbered and listed in the order they were recorded.
- Dates are formatted as day.month.year, and times use the 24-hour format (hour:minute:second).
- The class label is shown in yellow (class 1) or red (class 2).

③ **View an episode:**

a. Tap the **three-dot menu** next to the episode, then select **View episode**.

b. The episode viewer opens. Use the ▶ **play** and ⏸ **pause** buttons to control playback. Tap the ✖ button in the upper right to close the viewer.

NOTE: The viewer is set to 3× playback speed.



④ **Check the content of the episode, ensuring the following:**

- The class label matches the class shown.
- Only one class is shown in the episode.
- The target object or scene is visible for most of the episode.

⑤ **Delete episodes** that show the wrong class, include multiple classes, or do not show the target:

a. Tap the **three-dot menu**, then select **Delete**.

b. When prompted, tap **Delete** to confirm.

⑥ **Check classifier performance after deleting episodes:**

a. Go to the **Details screen** > **Record an episode**.

b. Show each class in turn and use the probability scores to evaluate the updated classifier.

c. Record additional episodes as needed.

5.6 Forced classification option

The classifier provides two operational modes:

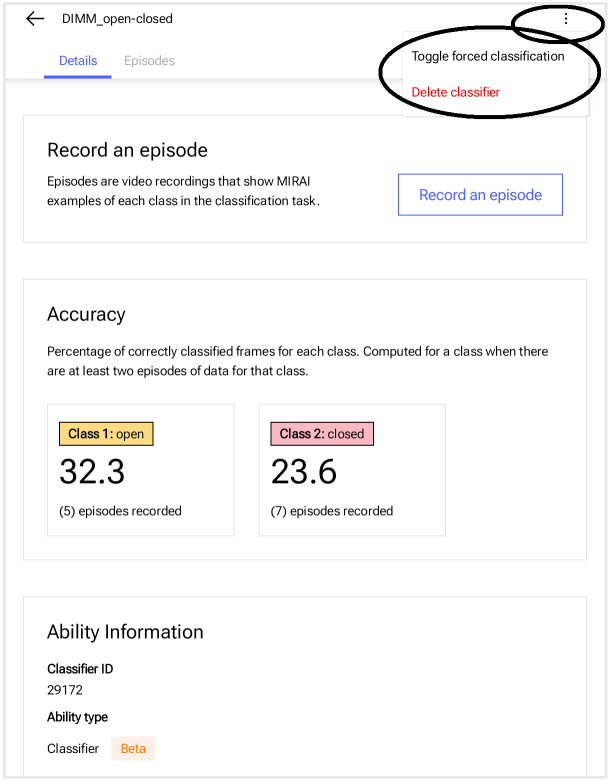
- **Default three-category output** (Class 1, Class 2, or "neither")
- **Optional forced classification** (Class 1 or Class 2 only)

The default "neither" output acts as a safeguard for uncertain classifications, while forced mode eliminates this safeguard. Use default mode in most production settings, and forced mode only when binary decisions are essential and classifier performance is fully validated. For details on each mode, see the ℹ box below.

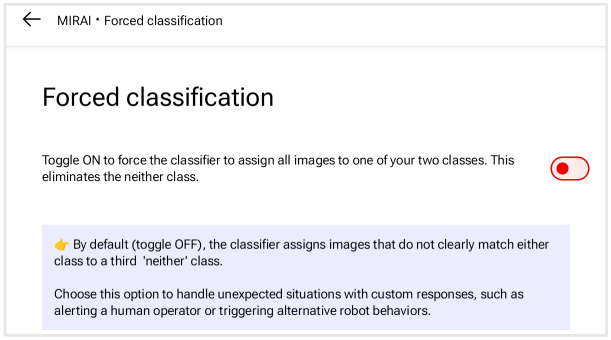
To turn on forced classification:

1. Navigate to Ability Overview and select your classifier from the list.
2. Tap the three-dot menu and in the upper right corner. Select Toggle forced classification.
3. Toggle on (green) or off (red) as needed

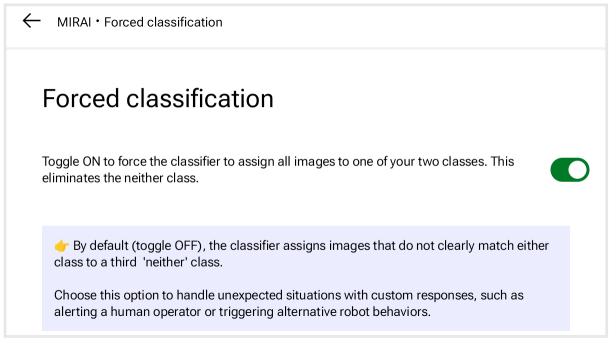
This setting can be modified at any time during operation.



Details Screen: Tap the three-dot menu in the upper-right corner.



Default: A **red toggle** means the classifier may return "Neither" if uncertain.



Forced classification: A **green toggle** forces the classifier to choose class 1 or class 2, even with low confidence.

When to choose each mode

Default three-category mode (*recommended*): Choose this mode to implement predefined safety measures in your robot program based on classification outcomes:

- **Class 1 or Class 2 results:** Robot executes the designated skill or action for that specific class
- **"Neither" result:** Robot executes predefined safety protocols (moves to safe position, stops operation, alerts human operators)

This mode provides critical production safety by handling unexpected scenarios such as foreign objects in the workspace through the "neither" classification.

Forced binary classification mode: Choose this mode only when:

- Your robot program requires guaranteed binary decisions without safety fallbacks
- You have thoroughly verified classifier performance across all expected production inputs
- Eliminating the safety "neither" option is acceptable for your specific application

Warning: Forced mode removes the built-in safety mechanism that handles uncertain classifications, requiring careful validation before deployment.

5.7 Implement a classifier in a robot program

Classifiers are called in robot programs using platform-specific implementation methods. Refer to your platform's Integration Manual for template robot programs and detailed instructions.

IMPORTANT: Close the recording screen in the MIRAI Training App before using your classifier in a robot program. Otherwise, the classifier may fail to deploy.

5.8 Duplicate classifiers across robots

Classifiers can be duplicated across multiple robots in your MIRAI controller group, allowing you to reuse classifiers on different setups. Each duplicate creates an independent copy that can be customized for its new environment while maintaining separate training data from the original.

Prerequisites

Before duplicating a classifier:

- Ensure both controllers are in the same controller group (contact Micropsi Industries to add controllers).
- Sync any skills that will be used with the classifiers on the new setup.
- Verify the new controller has LAN/internet connection to access cloud-based duplication.
- Prepare camera hardware that matches the original setup (same lens type and connection).

5.8.1 How classifier duplication works

Classifier duplication creates independent copies that can be customized for different setups. All versions function separately, allowing multiple specialized versions across different controllers.

Independent copies

When you duplicate a classifier, you create a completely separate instance:

- Each duplicate has the same name as original, but **a different 5-digit Classifier ID** for unique identification.
- Duplicates operate with **independent training data**, meaning changes to any duplicate will not affect the original classifier or other copies.

- Each copy can be **customized and fine-tuned** for its specific setup requirements without impacting other versions.

Typical workflow:

- **Train the base classifier** on Controller 1 to establish the required accuracy.
- **Duplicate the trained classifier** to Controller 2, creating an identical version with its own unique ID.
- **Add local training data** to Controller 2's copy if fine-tuning is needed for the specific setup.

> **Classifiers vs. Skills:** Unlike skills (which are "synced" with a single ID across controllers), classifiers are "duplicated" to create independent copies with unique IDs. This means that new training data on one setup will not affect classifiers on other setups.

5.8.2 Duplicate a classifier

Follow these steps on **the MIRAI controller for the new setup**:

1. Select **Ability Overview** from the main menu. Under **All abilities**, find the classifier you want to duplicate and tap **Duplicate**.

2. Tap **Class information** and review the two classes on the next screen. Tap **Confirm** to proceed.

3. Tap **Camera settings and alignment**. Select the camera you intend to use. Tap **Next**.

   TIP: If the camera doesn't appear, check installation, connection, and power, then restart the MIRAI controller.

   IMPORTANT: Replicate the original camera setup exactly, including lens and connection type (USB 3.0).

4. Set the **Gain** and **Exposure** sliders (and optionally, use and define a **Region of Interest**) to match the original setup. Tap **Apply settings**, then **Confirm**.

5. Verify the "Duplicate classifier" screen shows green "Completed" status for both sections. Tap **Finish**.

6. **Test the duplicate's performance on the new setup:**

   a. Toggle on forced classification if using.

   b. Open the recording screen and show examples of each class.

   c. Verify Probability values are high for the correct class.

   d. Record additional episodes if performance is insufficient.

   NOTE: Duplicate classifiers have the following limitations:

   - **Accuracy metric is unavailable:** Duplicates display "_____" under Accuracy instead of numerical values, even after recording new episodes. Use Probability values to evaluate performance instead.
   - **Episode viewing is unavailable:** While training episodes from the original classifier appear in the Episodes list, they cannot be viewed or accessed on duplicate controllers.

5.8.3 Managing multiple duplicates

Use these guidelines when working with multiple classifier duplicates across your system:

Scaling considerations

- You can create an **unlimited number of duplicates** from any classifier.
- Each duplicate can be **independently trained and customized** to its setup.
- Changes made to one duplicate **never affect other duplicates** or the original classifier.

Backup and restore

- Original classifiers **retain all episodes** when backed up and restored.
- Episodes cannot be viewed for duplicate classifiers.
- The target controller **must run a MIRAI version that supports classifiers**.
- The original training camera for the classifier must be used, but it can be paired with a different robot.

5.9 Classifier troubleshooting

Issue	Possible causes	Recommended action
MIRAI Training App: Tapping **Add** on the **Evaluate Data** screen does not save the episode. Instead, the app returns to the **Details** screen.	The MIRAI controller is not connected to the LAN.	Verify the network connection and try again.
MIRAI Training App: During training, the classifier's performance suddenly becomes unpredictable or its confidence drops.	One or more episodes may be mislabeled – the camera view shows one class, but the label corresponds to another.	Review the episodes to ensure each one is labeled with the correct class. See5.5.5 for details.
UR teach pendant: **Error message:** ⚠ XMLRPC: Failed with error code 500: DeviceOffline: Ximea camera serial number <*number*> is disconnected.	**A.** The camera is not on. **B.** The camera was replaced.	**A.** Turn on/connect the camera. **B.** ⓘ You must create a new classifier. If the camera is replaced (e.g., after a crash), the classifier will no longer work! **WORKAROUND:** Use another controller to create a duplicate, then use the current controller to create a duplicate of the first duplicate.
UR teach pendant: Classifier is not appearing on the TP under "List of available classifiers."	More time is needed to load the classifier on the teach pendant.	On the UR teach pendant: **1.** Navigate to **Installation** on the top menu bar. **2.** On the right pane of that screen, tap **Load skill and classifier information**. **3.** Tap **Program** on the top menu bar and check if the classifier appears in the list. If not, repeat steps 1 and 2.
Robot program: The robot program is not running properly.	The robot program is running while the classifier recording screen is open in the MIRAI Training App. Depending on the robot platform, the issue will be silent or throw an error.	Close the recording screen in the MIRAI Training App and try again.

6 Positioning skills

Positioning skills allow robots to precisely position end-of-arm tools at target objects even when task conditions vary, such as changes in object position.

Before recording data, review the skill's capabilities, confirm your system meets all requirements, and read the overview of recording options.

Then follow the remaining sections to create a recording plan, record data via hand-guiding or the automated Recording Assistant, and review data for optimal skill performance.

6.1 Key applications

Robots equipped with positioning skills can accurately position a tool at a target object for diverse applications. The robot follows the most direct trajectory to the target position, typically a straight path.

Typical use cases for positioning skills include:

- Placing a gripper in a pre-grip pose above objects that vary in color, size, shape, or position
- Positioning workpieces precisely for insertion in subsequent steps
- Placing sensor tools relative to workpieces for quality inspection
- Specialized tasks such as rack-hanging, cable-plugging, and leak-sniffing

In all cases, the path from the handover point (where MIRAI takes control) to the target object must be free of obstructions. Target objects must remain stationary, except when the conveyor-tracking functionality is activated on supported platforms.

6.2 Hardware and system requirements for positioning skills

Required hardware:

- Wrist-mounted camera(s)

Optional Hardware:

- Force/torque sensor (only needed for hand-guided recording)

Movement configurations

- **Translations:** All axes must remain unrestricted
- **Rotations:** Limited to either none or one axis only

6.3 Options for recording training data

Two options are available to record data to train positioning skills:

- **Hand-guiding:** Manually move the robot to desired positions (*force/torque sensor required*)
- **Recording Assistant:** Automated recording without requiring a force/torque sensor

You can switch from using the Recording Assistant to hand-guiding at any time to record episodes. After configuring the Recording Assistant setup, follow these steps to switch back to hand-guided recording:

a. On the bottom bar, tap **Recording Assistant** to expand options.

b. From the expanded menu, tap ← **Adjust basic setup**.

c. Tap ← **Back** to reach the "Step 2/3: Shape the handover region" screen.

d. Tap **Delete all**.

e. Confirm deletion when prompted.

f. Tap anywhere on the upper portion of the recording screen to exit the Recording Assistant setup. You can now begin hand-guided recording.

6.4 Record data for a positioning skill using hand-guiding

When your robot is equipped with a force/torque sensor, you can hand-guide the arm to record training data. This section leads you through the screens in the MIRAI Training App and provides a recording tutorial for your first training sessions. Proceed through the blue headings below, following the numbered steps within each subsection.

Look for **info boxes** throughout this section for best practices to optimize your training data.

6.4.1 Open a data recording session

1. From the main menu, tap **Skill Overview**.
2. Go to your skill and tap **Open** to go to the Details page.
3. Scroll down to "Step 1: Collect data" and tap **Record an episode**.

6.4.2 Position the robot and calibrate the force/torque sensor

Before recording each episode, you will be prompted to position the robot and calibrate the force/torque sensor.

1. **Position the robot:**

   a. Tap either **Send to reference position** or **Manual positioning**.

   How to choose positioning behavior

   This step is mandatory in the app flow but does not necessarily require physical action on the robot. Each positioning option has specific use cases:

   Send to reference position

   Select this option to automatically move the robot to a predefined position. This is useful when:

   - You need to return the robot to a known position.
   - It is possible to maintain a consistent target position across multiple recordings, such as when varying only lighting or background and not moving the target object.

   Manual positioning

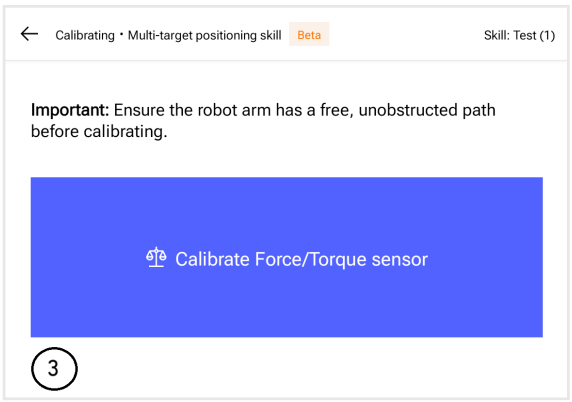
   Select this option when:

   - You prefer to position the robot using hand-guiding.
   - The robot is already in your desired position. No physical movement is needed – tap to acknowledge the current position.

2. **Calibrate the force/torque sensor:**

   a. Let go of the robot arm (any force will affect calibration).

   b. Tap **Calibrate force/torque sensor**.



6.4.3 Recording tutorial

This section includes detailed technique guidance for effective data collection - experienced users may proceed directly to recording episodes, referencing 6.4.4 Record episodes for a positioning skill if needed.

> Look for **TIPS** throughout this section for practical advice.
>
> Look for **info boxes** for more in-depth guidance to improve training.

WARNING

When hand-guiding the robot, stay aware of your surroundings to prevent injury. Keep your hands clear of pinch points, moving parts, and sharp objects.

1. **Check and adjust guiding sensitivity**
   - **a.** First, test the current sensitivity:
     i. Position the gripper in the target pose above a workpiece.
     ii. Move the gripper in small circles around the workpiece, gradually moving upward and outward.
     iii. If rotations are enabled, rotate the wrist.
   - **b.** Evaluate the sensitivity:
     It should allow these movements to be performed easily and consistently across multiple episodes.
   - **c.** If adjustment is needed:
     i. Tap **Guiding Sensitivity** on the bottom bar.
     ii. Adjust the sensitivity and tap **Apply Settings**.
     iii. Test the movement again and readjust if necessary.

Note: If you are using rotations, reducing the sensitivity helps achieve smoother trajectories.

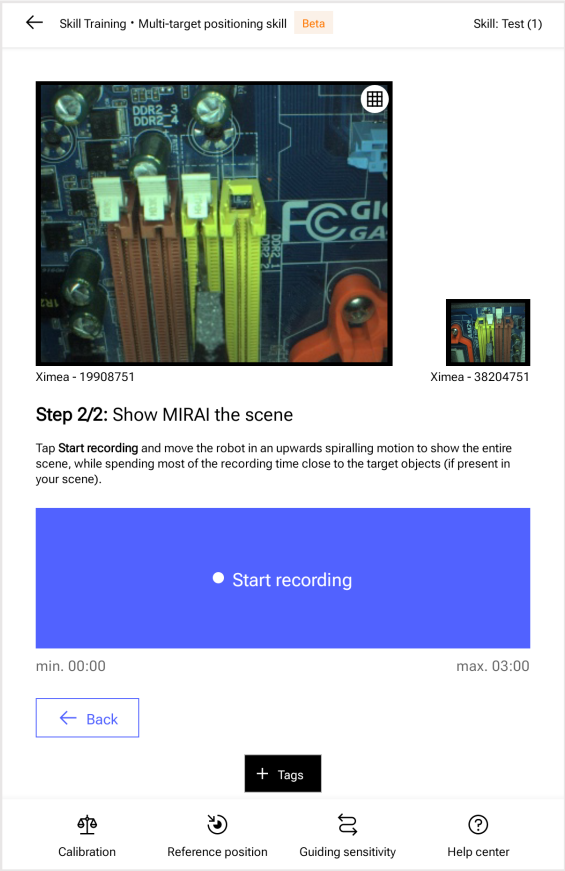
1. **Add tags to describe the episode:**
   - **a.** Tap + Tags.
   - **b.** Enter specific keywords ("tags") that describe the episode.

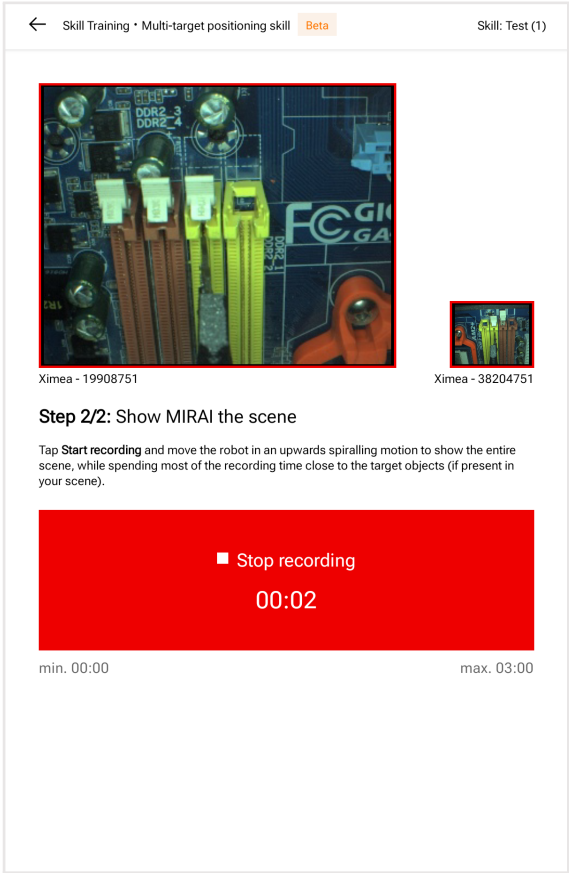
Why use tags?

Tags are keywords attached to episodes. Tags give you a quick overview of the content of your episodes and help identify gaps in your recording plan. By using meaningful names, you can better organize, track, and plan recordings across different conditions and scenarios.

Common tags include:

- Workpiece orientation
- Lighting conditions
- Background scene
- Person recording





2. Position the robot tool in the target pose. Be as consistent as possible across episodes.

   TIP: To maintain a consistent vertical distance between the workpiece and the tool, lower the tool tip until it just touches the workpiece. Then, enter the desired offset on the teach pendant to raise the tool by that amount.

3. Tap **Start recording**.

   - The button will turn red and display a timer to indicate that recording is in progress.

4. **Begin by hand-guiding the robot arm around the scene.**

   This episode will be deleted. Its purpose is to help you get comfortable with the recording technique.

   As you guide the arm, focus on the following:

- Keep the target in the camera frame.
- Stay close to the target for most of the recording time (see **Recording time allocation**).
- Move around the target by spiralling upward and outward to show it from different views (see **Hand-guiding fundamentals** and **Hand-guiding technique**).
- If you have rotations, rotate while spiraling.
- Periodically let go of the robot and let it return to the target, then begin spiralling away again (see **Release and return technique**).

Recording time allocation

Maximum episode duration: 3 minutes

Follow the 40/30/20/10 rule for distance-based time allocation:

- 40% of time up close (< 5mm from target)
- 30% of time near (< 10mm from target)
- 20% of time at medium distance (< 50mm from target)
- 10% of time far away (> 50mm from target)

Note: This guideline is most important when recording your first 20 episodes, but becomes less critical as you record more episodes and build up your dataset.

Hand-guiding fundamentals

Dual awareness: Keep one eye on the workspace to avoid collisions and another eye on the camera view to ensure the target remains in view.

Hand placement: Place your hand just below the force/torque sensor along the robot arm. Holding the wrist near the force/torque sensor ensures that even tiny motions from the tip of the tool/gripper are properly captured in the training. Reaching from higher up the arm will be more difficult to steer and may result in jerky or unpredictable pathing.
IMPORTANT: You must delete episodes that show your hand in the camera view. Be especially careful to avoid this while rotating the arm.

Hand-guiding technique

Goal: Recreate all positions and orientations the robot might move through during skill execution. This shows MIRAI what the target looks like from different perspectives as the tool moves through various positions.

General Motion Strategy: From the target position, slowly move upward in a spiraling motion. If rotations are enabled, constantly rotate the tool to show the target from different angles.

Axis Movement: Move through all available axes of translation and rotation while recording episodes. Examples:

- 6-axis skill: Move the robot tool through all 6 axes of motion
- 4-axis skill: Move through 3 axes of translation while rotating the 4th axis

Release and return technique

2 –3 times per episode, let go of the robot arm. It will move back to the target position via the shortest direct path. This technique:

- Keeps recordings focused on the target position
- Provides a preview of the final trajectory from a given position
- Shows the system direct paths to the target
- Helps introduce more variance around the target per episode

5 When you are finished, tap **Stop recording**.

6. Discard this episode:
   - Tap **delete**.
   - You will then be prompted to move the robot to the reference position and to calibrate the force/torque sensor.
7. **Now try a few practice episodes:** Aim to identify necessary adjustments to your setup or technique.
   a. Tap **Start recording** and walk through an episode from your training plan.
   b. When you are done, tap **stop recording**.
   c. Then tap **Discard** to delete the episode.
   d. Try recording an episode for a different condition to get a feel for the variance you intend to show.
8. Evaluate your experience and make any needed adjustments.
   - Did the recording go as expected?
   - Do you need to change anything, such as your timing or guiding technique?

   After you are satified, move on to recording the episodes to train your skill.

6.4.4 Record episodes for a positioning skill

This section provides step-by-step instructions for recording episodes in the MIRAI Training App.

> Look for **TIPS** throughout this section for practical advice.
>
> Look for **info boxes** for more in-depth guidance to improve training.

WARNING

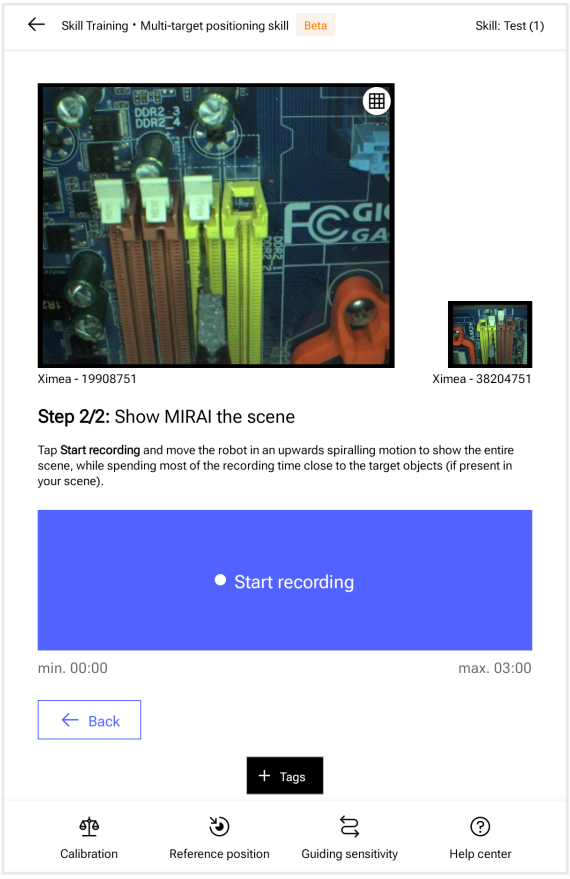
When hand-guiding the robot, stay aware of your surroundings to prevent injury. Keep your hands clear of pinch points, moving parts, and sharp objects.

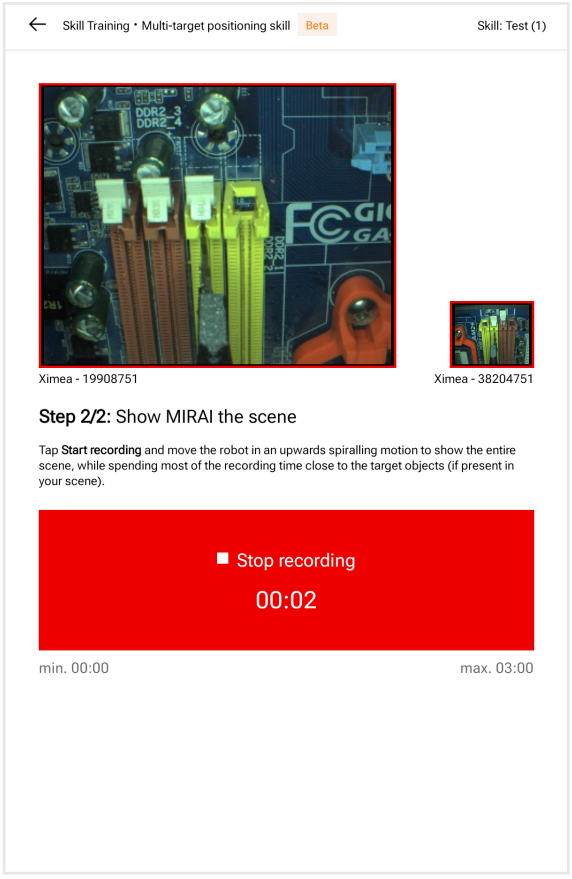
Recommendation

Back Up Your MIRAI Controller regularly, especially after refining a skill. In case of damage or loss, a backup facilitates recovery. See 16 Backup and Restore for instructions.

1. **Add tags to organize episodes:**
   a. Tap `+ Tags`.
   b. Enter specific keywords (”tags”) that describe the episode.
      Common tags include:
      - Workpiece orientation
      - Lighting conditions
      - Background scene
      - Person recording

      Why use tags? Tags give you a quick overview of your episodes and help identify gaps in your recording plan. By using meaningful names, you can better organize, track, and plan recordings across different conditions and scenarios.





2. Position the robot tool in the target pose. Be as consistent as possible across episodes.

   TIP: To maintain a consistent vertical distance between the workpiece and the tool, lower the tool tip until it just touches the workpiece. Then, enter the desired offset on the teach pendant to raise the tool by that amount.

3. Tap **Start recording**.

4. **Guide the arm around the scene.** Pay attention to the following:

   - Gently move the tool around the target by spiralling upward and outward, showing the target object from different perspectives.
   - If you have rotations, rotate while spiraling.
   - Stay close to the target for most of the recording time.
   - Keep the target in the camera frame as you move further away.
   - Periodically let go of the robot and let it return to the target, then begin spiralling

   TIP: You can record an episode for up to 3 minutes. The elapsed time is shown on the screen.

5. When you are finished, tap **Stop recording**.

6. Save or discard the episode:

   - Tap **Save** to keep the episode.
   - Tap **delete** if the episode shows anything that would not be seen during production.

When to delete episodes

To ensure MIRAI learns correct behavior, delete any episode showing actions or visuals that wouldn't occur in production. Remember: MIRAI learns from what it sees, so unintended behavior can lead to poor performance.

Stop recording and delete the episode if any of the following occur:

- A hand or any unintended object enters the camera view.
- The target object is missing from the camera view.
- The target object is moved during recording.
- Lighting changes abruptly during recording.
- The robot reaches joint limits or enters a singularity.
- The tool collides with the target or any fixture.

You will then be prompted to move the robot to the reference position and to calibrate the force/torque sensor.

7. Repeat to record further episodes. Refer to your recording plan to capture the planned variance.

   Record 5-10 episodes, not more. Use the episode counter (top right of the screen) to track your progress.

   When finished, tap the home button to return to the Skill Details page.

   Continue working through your recording plan. After you have recorded your initial episodes, send your data to the Micropsi cloud to create the first version of your skill.

6.5 Record data using the Recording Assistant

The *Recording Assistant* is a tool that automates episode recording by moving the robot autonomously within user-defined areas. It offers two key benefits:

- It allows you to train positioning skills without a force/torque sensor.
- It provides an alternative to hand-guiding when using a force/torque sensor.

This section provides detailed instructions on setting up the Recording Assistant and recording episodes.

6.5.1 Configure the Recording Assistant

The Recording Assistant setup consists of three screens: handover behavior, handover region, and motion settings. The numbered steps in this section guide you through each screen in the MIRAI Training App.

> **Look for information boxes** throughout this section for guidance on how to make appropriate selections and determine values for your specific application.

1. **Access the Recording Assistant setup:**

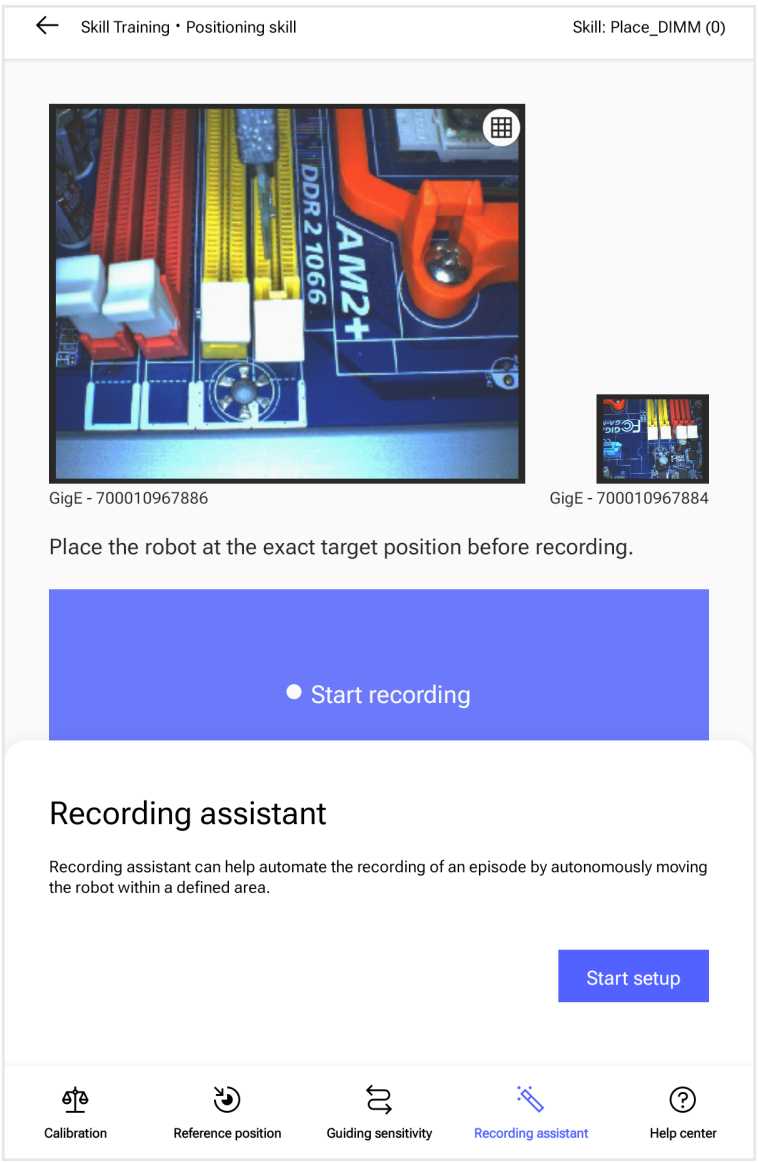
   a. From the main menu, tap **Skill Overview**.

   b. Go to your skill and tap **Open** to go to the Details page.

   c. Scroll down to "Step 1: Collect data" and tap **Record an episode**.

   d. On the bottom bar, tap **Recording assistant** to expand the the bottom sheet.

   e. Then tap **Start setup** on the bottom sheet.

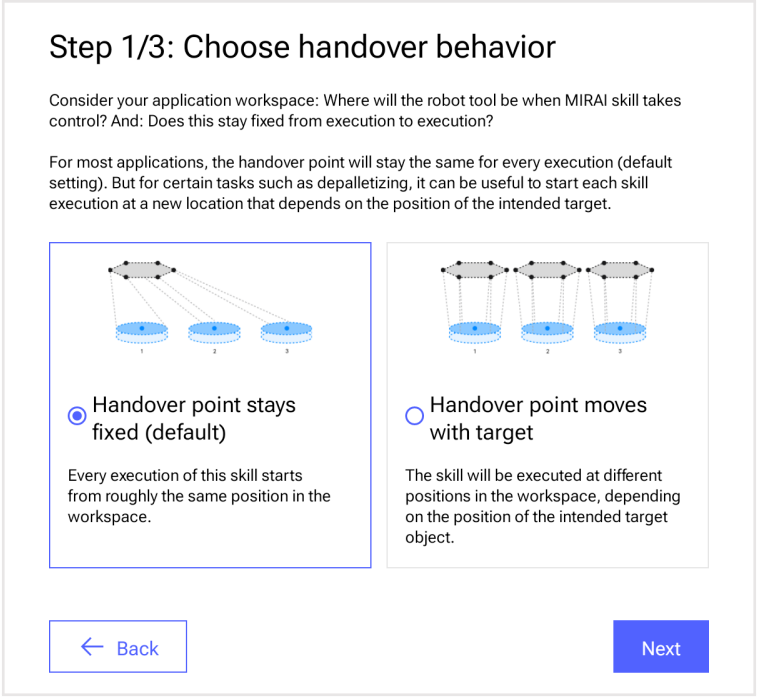


2 **Configure handover behavior:**

a. Select **Handover points stays fixed** or **Handover point moves with target**.

b. Tap **Next** to proceed.

> **How to choose handover behavior**
>
> *Handover behavior* determines where your robot will transfer control to MIRAI to begin skill execution. Most applications use a fixed handover point.
>
> - Choose **Fixed** when the handover point should remain at the same absolute position in space regardless of target location (e.g., performing an operation at a stationary workstation or picking from a fixed conveyor belt location).
> - Choose **Moving** when the handover point should automatically adjust its position based on the target object's location (e.g., depalletizing items from stacks of varying heights).



3 **Configure the handover region:**

a. Move the robot's end effector to a point in space near the handover point.

b. Tap **Save position**.

c. Repeat to save at least three points, creating a polygon above the target. To change the points, tap **Delete all** and set new points.

d. Tap **Next** to proceed.

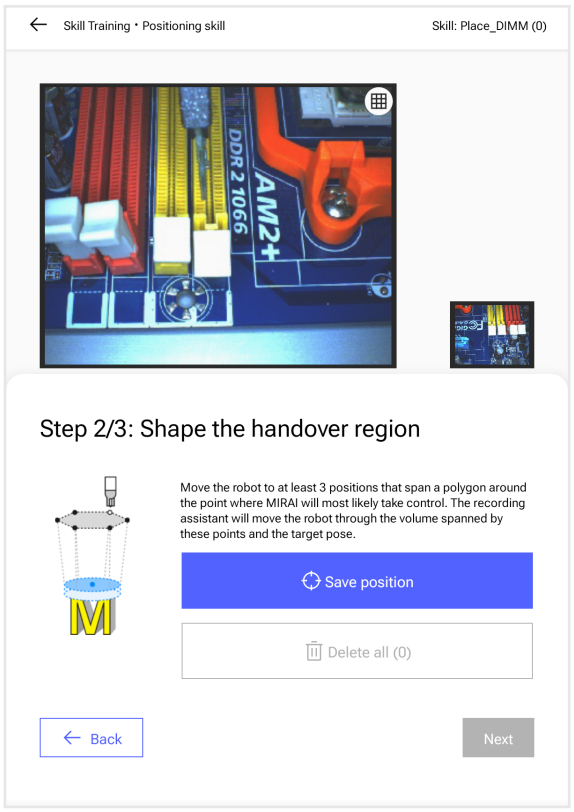
How to shape the handover region

The *handover region* defines the spatial boundaries where the Recording Assistant will move the robot during the recording process. This region serves two critical purposes:

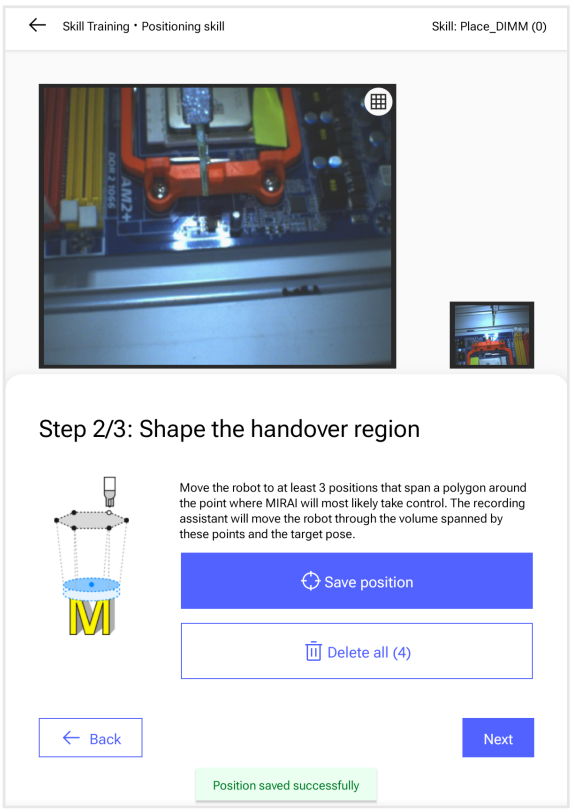
(1) It establishes a safety boundary that restricts the Recording Assistant's movements.
(2) It creates a space to collect comprehensive data about the task, ensuring the model learns from diverse positions.

To define the handover region, consider these application-specific factors:

- **Coverage:** Position the polygon points to cover all possible motion paths between the handover point and the target. The target may be positioned in any direction (below, above, or to any side) relative to the polygon.
- **Position extremes:** Move the robot arm to the extreme positions in all enabled degrees of freedom while keeping the target in view.
- **Rotation:** If rotations are enabled, include rotational extremes in your polygon definition.
- **Efficiency:** Keep the polygon as compact as possible while capturing necessary variation.
- **Workspace geometry:** Shape your polygon in relation to your workspace. Ensure the polygon covers diverse positions and orientations relevant to the task.



Before: No points set for the handover region



After: 4 points set (shown after 🗑 **Delete all**). Note the changed camera views.

④ **Configure motion settings:**

a. *If rotations are enabled*, enter a value between 0° and 45° under **Amplitude of random rotations** to set the maximum amplitude per tool axis.

> **Rotation amplitude during recording**
>
> When rotations enabled, the Recording Assistant introduces minor random adjustments to tool orientation throughout the recording process. While you can limit the range of these movements, be aware that the Recording Assistant will automatically rotate the tool as necessary to transition between target pose orientations and those in the handover region. Significant orientation differences between these positions will result in greater overall rotation, regardless of your specified rotation restriction settings.

b. Define **Obstacle-free space near the target**:

i. Under **Orientation:**
- Select the axis closest to the main direction of motion near the target.

ii. Under **Size:**
- Enter the distance in mm **beyond** the target pose – this space extends from the target pose along a chosen axis.
- Enter the distance in mm **around** the target pose – this space determines the radius of the cylindrical space.

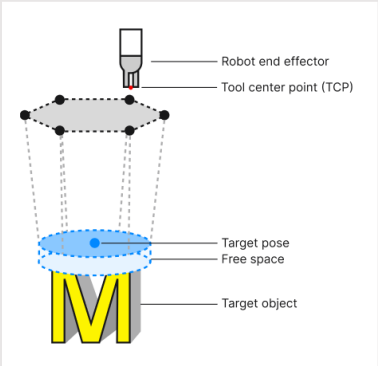
Recording beyond and around targets

To enhance MIRAI's ability to generalize and recover from positioning errors, record data from multiple positions relative to the target pose. During production, your skill will guide the tool to the target pose (slightly offset from the target object by a few millimeters). For training purposes, however, the Recording Assistant must capture data from a wider area.

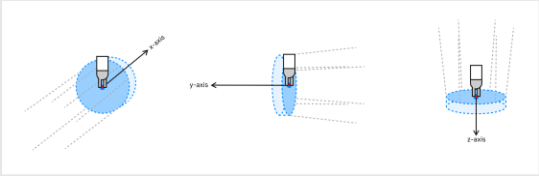
In the figure below, this area near the target is shown as a blue cylinder for illustrative purposes. During recording, it forms a continuous recording area up to the polygon. Define the cylindrical space around the target point with two parameters:

- **Beyond:** The space between the target point and the target object, allowing MIRAI to learn recovery from overshooting.
- **Around:** The radial distance from the target point, enabling recovery from imprecise positioning.

The MIRAI Training App shows default values of 5mm for both parameters, which you can adjust based on your specific application requirements. This comprehensive sampling ensures MIRAI can successfully operate even when positioning is not perfect.

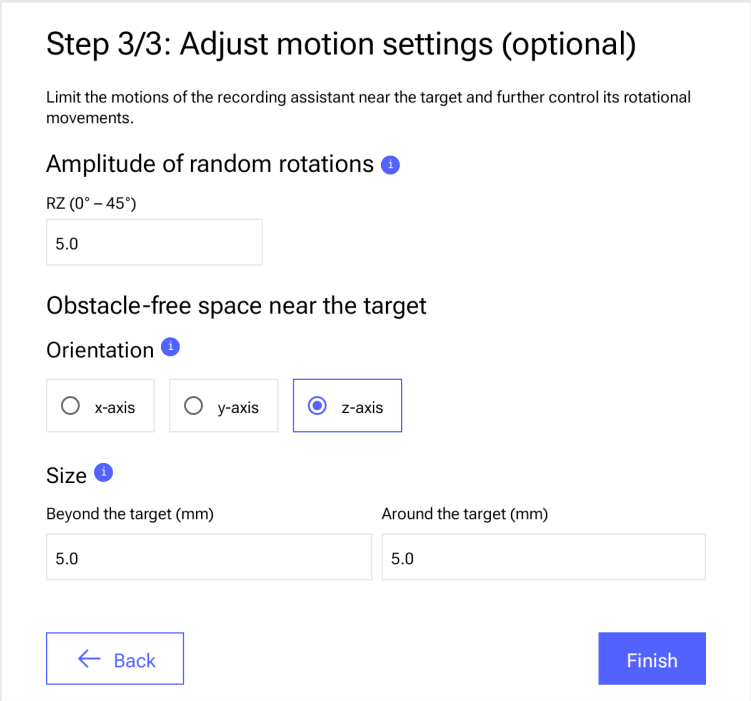


Recording area: Space between handover and target.



Orientation: Select main motion axis.

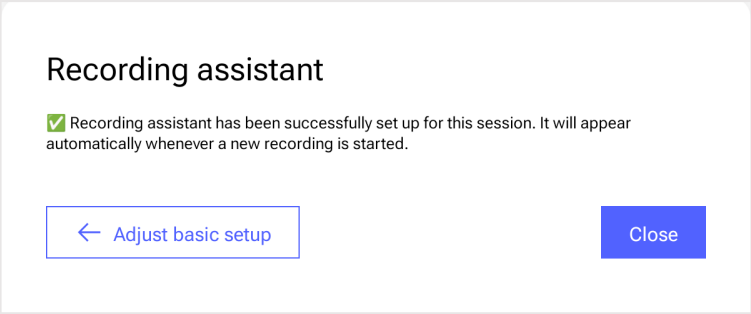
Size: Define space beyond and around target.



c. Tap **Finish** to proceed.

d. Close the setup or modify if needed:

- Tap **Close** to proceed to the recording screen.
- To modify your setup, tap ← **Adjust basic setup**. Then tap ← **Back** to reach the desired screen.



6.5.2 Record episodes with the Recording Assistant

During recording, the robot moves autonomously within the defined area, but you still play an active role. Your responsibilities include:

- Starting and stopping each recording
- Monitoring the robot's movements throughout the process
- Ensuring all safety precautions remain in place

The numbered steps in this section will guide you through the recording screens in the MIRAI Training App.

Safety precautions

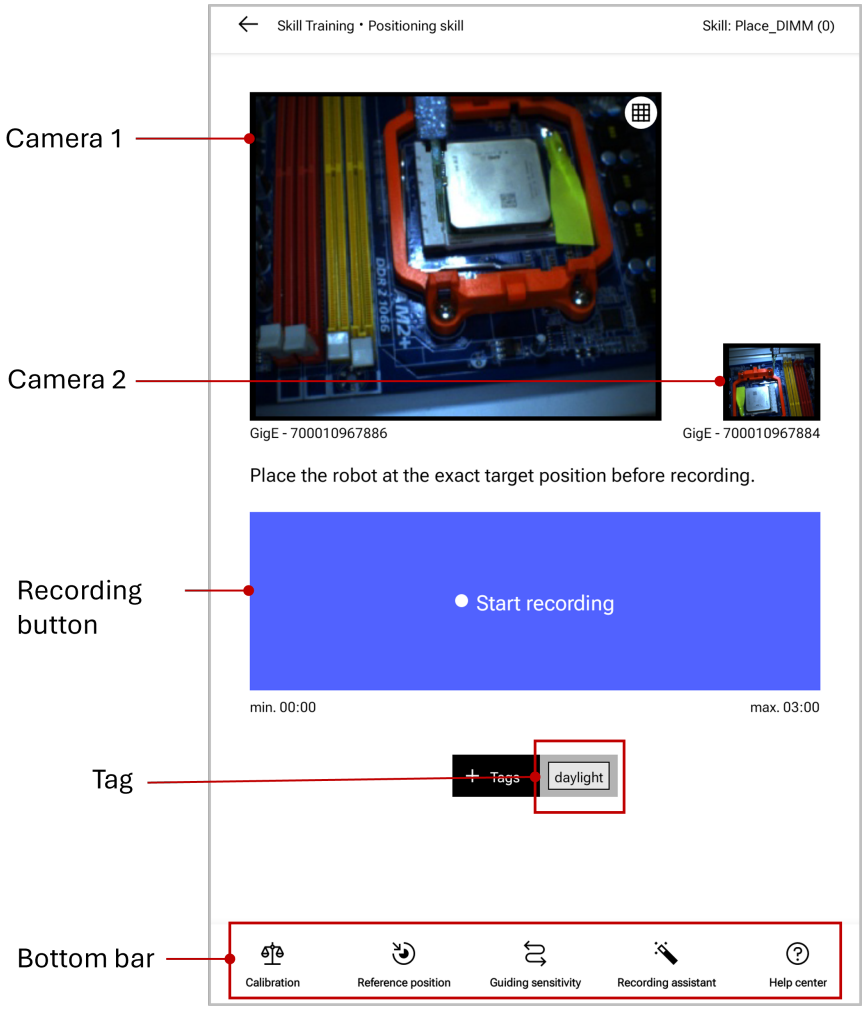
- ☐ **Monitor the robot's movement** closely and pause recording if necessary.
- ☐ Ensure that recording area around the target is **free of obstructions**.
- ☐ Ensure proper **cable management** to accommodate robot rotation during recording.
- ☐ **If the TCP was not displaced**, ensure the robot's movement matches your expectations.
- ☐ **Begin at a slow setting** and increase speed as you become comfortable with the movement.

1. **Select tags (keywords) to identify episodes** (optional):
   a. Tap `+ Tags`.
   b. On the pop-up screen, manage tags by:
      - Entering new tags
      - Selecting existing tags from the list
      - Deselecting unwanted tags
   c. Tap **Done** when finished. All tags will appear next to the `+ Tags` button.

! IMPORTANT: Tags persist between episodes. Any tags you select will automatically be added to subsequent episodes unless manually modified.

You can review and modify tags in the Skill Overview. For instructions, see 8.5 Adding tags (keywords) to episodes.

2. **Initiate recording process:**
   a. Position the robot at the target pose with maximum precision. This critical step directly impacts skill accuracy and requires careful attention – take the time needed to get it right.
   b. Tap **Start recording**.

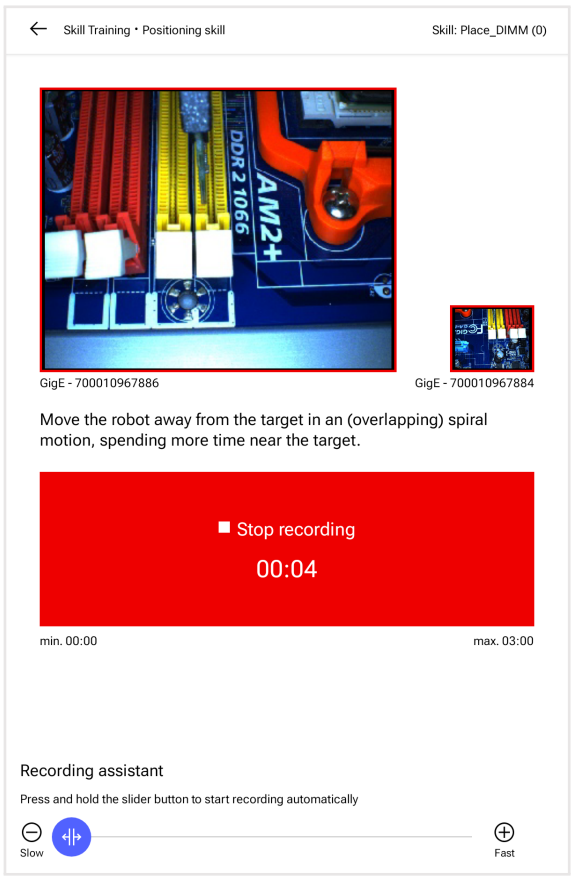


③ **Activate the Recording Assistant and monitor recording:**

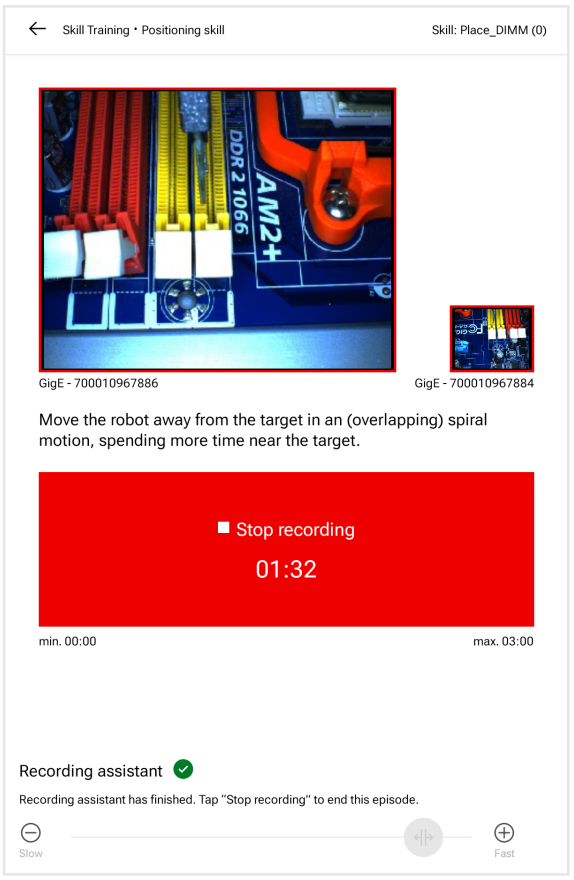
a. Tap and hold the **blue slider** at the bottom of the screen. Start with the slowest setting and gradually increase speed as appropriate. The robot arm will begin exploring the defined area near the target.

b. Monitor the robot's movements, with special attention to cable management. Allow the robot to complete its exploration sequence, after which it will automatically return to the target pose.

! IMPORTANT: If you release the slider, the robot arm will pause while video recording continues. Tap the slider again to resume movement.



To activate the Recording Assistant: Tap and hold the slider for the duration of the recording.



To end the episode: Wait for ✅ to appear, then tap **Stop recording**.

4. **Stop the recording:**
   a. When the Recording Assistant completes its exploration, the arm will return to the target pose and stop. Wait for the green confirmation indicator ✅ to appear at the bottom of the screen.
   b. Tap **Stop recording**.

5. **Save or discard the episode:**
   a. Tap **Save** to store a successful recording or **Discard** to delete it.

   NOTE: After saving, a "Processing the episode..." message will appear for several seconds.

6 **Continue recording episodes:**

a. Tap **Start recording** to proceed with the next episode.

b. Introduce planned variations for each episode (for example, repositioning the target object, changing lighting conditions).

c. Repeat steps 1 to 5 for each new episode.

d. Record 5–10 episodes that capture the essential variations expected during production.

TIP: The number of recorded episodes is displayed in brackets next to the skill name in the top-right corner of the recording screen.

e. Return to the Skill Details screen to review your target poses for consistency:

- Tap the main menu ☰ > **Skill Overview** > *<your skill name>* **Open**.
- See 6.6 Check target pose consistency for instructions.

IMPORTANT: Focus on quality over quantity in your initial recordings. Begin with just enough episodes to capture essential variations, then test your skill. Add targeted recordings only where refinement is needed. This strategic, iterative approach delivers superior results more efficiently than collecting excessive data upfront.

6.6 Check target pose consistency

After recording 5 to 10 episodes to train a positioning skill, it is crucial to ensure that the target positions are consistent across all episodes before sending the training data to the Micropsi cloud. Consistent target positions are the key to achieving accurate MIRAI skill execution. For details, refer to 3.4 Quantify the Target Position). Follow these steps to check your data using Target Frame Review: Following these steps will make your training data more accurate and improve the execution of your skill.

1 **Open Target Frame Review:**

a. On the Skill Details screen, scroll down to **Step 2: Review target frames**, tap **Start review** to display the first frame of an episode. This "target frame" shows the view as the arm is positioned at the target.

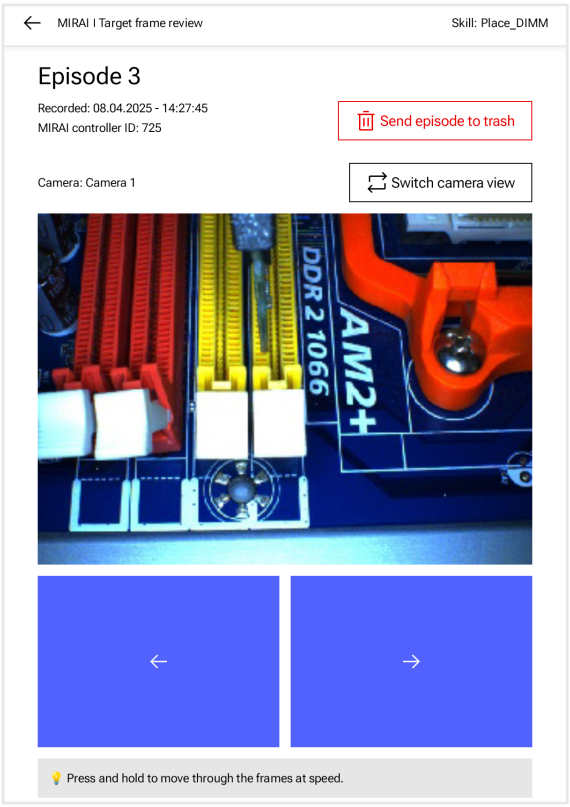
NOTE: You can only review local episodes. Episodes recorded on other controllers, such as in shared skills, cannot be reviewed.

2 **Review episode consistency:**

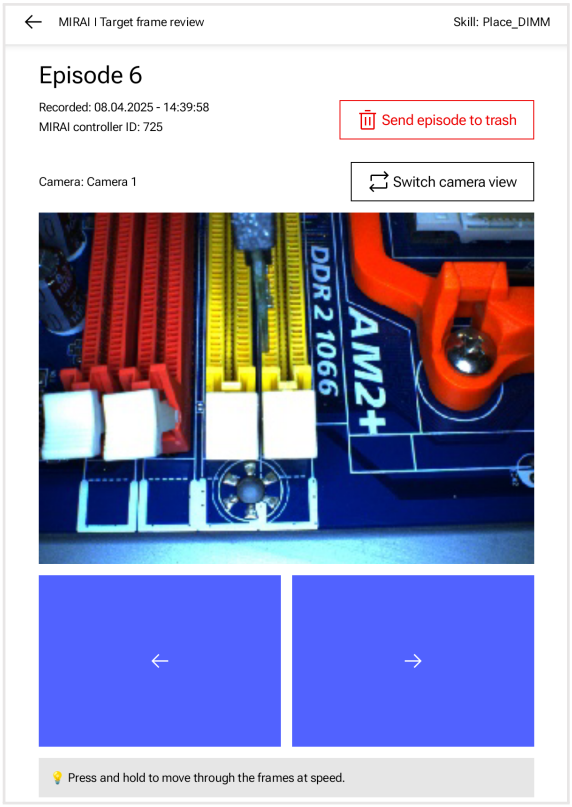
a. Navigate through the frames:

- Use blue navigation buttons: → (forward) and ← (backward)

- Press and hold either button for rapid scrolling.
- For dual-camera skills, tap **Switch camera view** to toggle between cameras.



Intended target pose



Inconsistent target pose: Note the position of the DIMM above the yellow slot.

b. Evaluate positioning consistency: Focus on the robot's end effector position relative to the target across all episodes. Even subtle differences will be evident when moving at speed. Look for the following cases and respond as needed:

Consistent positions: End effector maintains stable positioning with no "jumps" (background variations are acceptable). Keep these episodes for skill training.

Isolated inconsistencies: For occasional position shifts, tap **Send episode to trash** to remove specific problematic episodes.

Frequent inconsistencies: If nearly every image shows position shifts, your training data lacks the necessary consistency – consider starting over.

3 **Refine episodes through iteration if needed:**

a. Record additional episodes after removing inconsistent ones, ensuring precise target positioning for each new recording.

b. Review all episodes together for collective consistency.

c. Continue this iterative process until all episodes show consistent end effector positioning.

You are now ready to create a skill version and test your skill. For details, refer to 10 Testing, Refining, and Embedding MIRAI Skills.

6.7 Troubleshooting: Positioning skill training

Refer to this section to resolve issues with positioning skills.

Issue	Possible causes	Recommended action
Recording assistant: The tool's random rotation exceeds the value set in the MIRAI Training App.	This is expected behavior. The Recording Assistant automatically adjusts the tool's orientation to transition from the target pose to the orientation of poses in the handover region. If these orientations differ significantly, the resulting rotation may exceed the configured value, as this adjustment is independent of the user-defined settings.	Expected behavior – no action needed.
During hand-guiding Unexpected movement direction – moving against your guidance or resisting in the opposite direction.	**A.** The force/torque sensor was calibrated while being touched. **B.** the force/torque sensor is not installed correctly.	**A.** Recalibrate, ensuring no force is applied to the force/torque sensor. **B.** Ensure the force/torque sensor is mounted correctly. Refer to the printed symbols on the sensor (e.g., +Y, -X) and align them with the robot flange.

7 Multi-target positioning skills

Multi-target positioning skills allow robots to handle tasks involving multiple target objects, such as picking objects one by one. This section covers skill functionality (7.1) and setup requirements (7.2) and guides you through recording training data (7.4).

7.1 Introduction to multi-target positioning skills

Multi-target positioning (MTP) allows robots to identify and manipulate randomly scattered objects. The skill guides the robot to the nearest object and positions the gripper for picking. After the object is picked, the robot returns to a predefined position to reassess the workspace and the cycle repeats until all objects are collected.

For the skill to work effectively, objects must meet these requirements:

- Belong to the same category (similar shape, size, and appearance)
- Lie flat on the surface
- Not overlap each other

Key differences from general positioning skills

Like single-target positioning skills, MTP skills guide the robot to a target. But MTP skills include an extra step: identifying the *closest target*. This requires some adjustments to the training routine, outlined below and described in each section.

Camera setup: A single, wide-angle lens is used to keep the entire target object in view. (Section 5.2)

Starting position: Recording can start from any position. The robot automatically moves to the closest target object. (Section 5.3.2)

Training motion: A narrow hand-guided spiral is used to view one object at a time. (Section 5.5.3)

7.2 Setup requirements for multi-target positioning skills

To train a multi-target positioning skill, the following components are required:

- Force/torque sensor
- Wrist-mounted camera
- Single camera with a wide-angle lens

Additional information:

- Axis translations cannot be restricted.
- Rotations: None or one axis

Why a wide-angle lens improves MTP training

A wide-angle lens captures a broader field of view than a normal lens, allowing more of the scene to be included in the frame. A broad field of view is crucial for MTP for two reasons:

1. It shows multiple targets in a single frame, which is necessary to identify the *closest* target.
2. It ensures the entire target object is visible, even when the robot is close to the table, which is necessary for accurate positioning above a single object in a group.

These characteristics enhance the skill's ability to first *identify* and then *accurately pick* objects.

7.3 Prepare a recording plan for a multi-target positioning skill

Before recording episodes for your multi-target positioning (MTP) skill, ensure you consider how to position at your target and what scenes you will show. This section provides guidance on different factors you will need to consider.

> **What is variance?**
> Variance refers to visual differences across scenes that affect skill performance. For multi-target skills, key variance sources include:
>
> *Object variability:* Differences in object appearance and orientation
> *Arrangement variability:* Different quantities and spatial patterns (scattered vs. clustered)
> *Lighting conditions:* Variations in brightness and shadow patterns
> *Background changes:* Different visible backgrounds in camera view

7.3.1 Define target pose

Establish a precise "target pose" – the exact gripper position and orientation relative to each target object. Document this pose clearly, as you must recreate it consistently for every target in each episode. This consistency teaches MIRAI the correct gripper alignment for successful picking.

7.3.2 Assess camera coverage and background variance

Evaluate the complete camera field of view by moving through the workspace while watching the camera view in the MIRAI Training App. Ensure only elements present during production appear in view, particularly when using wide-angle lenses that capture extended workspace areas. Remove any temporary items that won't be present during skill execution.

7.3.3 Define handover points

Z-axis rotation considerations: When defining handover positions for skills requiring z-axis rotation, provide at least 180 degrees of rotational freedom. The robot chooses the shortest rotational path to target. Insufficient range causes singularities – for example, if a 100-degree rotation is needed but only 90 degrees are available in the preferred direction, the robot attempts the constrained path and encounters a singularity.

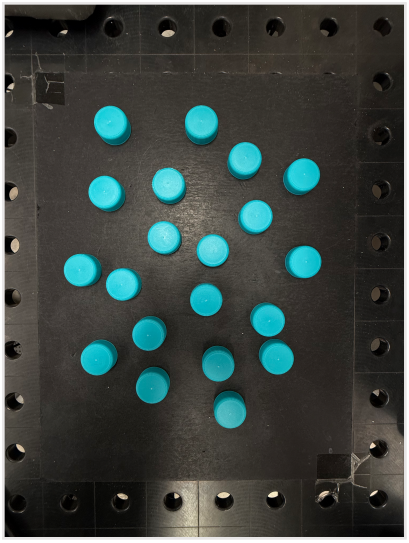
7.3.4 Arrangement variability

Unlike single-target skills, MTP skills must account for a wide range of object quantities and spatial arrangements. Your recordings should reflect the diversity of scenarios expected during real-world deployment. For example, if your application will include different numbers of targets, plan episodes that include different numbers of targets (e.g., 2, 3, 5+). You must also consider the expected layouts of your workpieces (e.g., clustered, evenly spaced, irregular) and record episodes accordingly. This ensures the slill works well across production conditions.

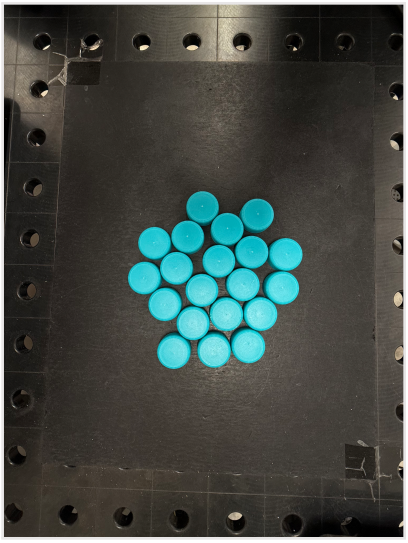
7.3.5 Plan episode count

Record 15-20 episodes for initial MTP skill testing. Use the episode counter (top right corner) to track progress – the bracketed number shows saved recordings. Adjust episode count based on scene complexity and object arrangement variations.

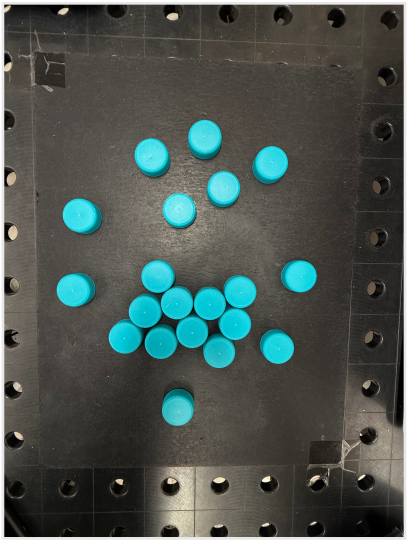
Number of objects: You do not need to show MIRAI the exact number of parts used in the skill execution. For example, if you have up to 10 parts, start with episodes with 1 part, 3 parts, 5 parts, 8 parts, 10 parts. Then record the expected arrangements for each number of parts (e.g., scattered, clustered, and/or mixed).



Parts are scattered across the workspace.



Parts are clustered on the workspace.



Parts are both scattered and clustered.

7.4 Record data for a multi-target positioning skill

To record data, follow these three steps, repeating them for each new episode:

1. **Position and calibrate:** Send the robot to the reference position and calibrate the force/torque sensor.
2. **Save target poses:** Position the tool above each object in the scene.
3. **Record data:** Hand-guide the robot through the entire scene and show each each object in detail.

These steps are described in the sections below.

Look for **info boxes** throughout this section for best practices to optimize your training data.

WARNING

When hand-guiding the robot, stay aware of your surroundings to prevent injury. Keep your hands clear of pinch points, moving parts, and sharp objects.

7.4.1 Open a data recording session

1. From the main menu, tap **Ability overview**.
2. Go to your skill and tap **Open** to go to the Details page.
3. Scroll down to "Step 1: Collect data" and tap **Record an episode**.

7.4.2 Position the robot and calibrate the force/torque sensor

Before recording each episode, you will be prompted to position the robot and calibrate the force/torque sensor.

The **reference position** is an optional, user-defined point for recording data and testing skills. It ensures the robot can start from the same position for consistent recording and testing conditions.

① **Position the robot:**

a. Tap either **Send to reference position** or **Manual positioning**.

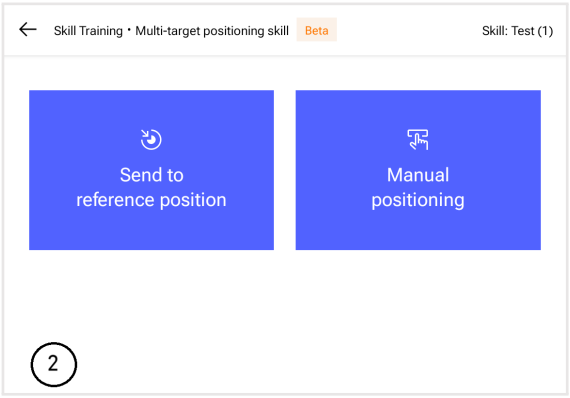
TIP: For multi-target skills, we recommend using the reference position. Choose a vantage point that displays all potential targets, allowing MIRAI to identify the nearest one.

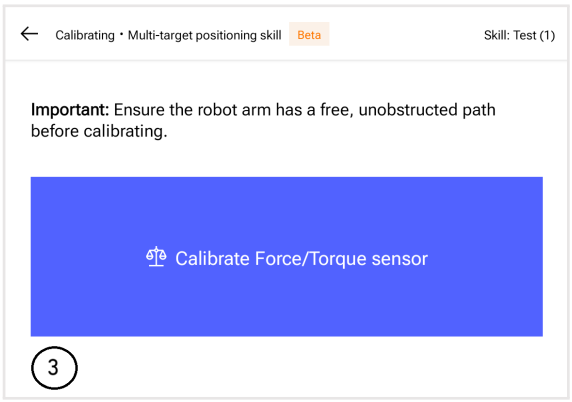
② **Calibrate the force/torque sensor:**

a. Let go of the robot arm (any force will affect calibration).

b. Tap **Calibrate force/torque sensor**.

Then proceed to the next screen to save target poses.





Note

To improve skill accuracy, move the tool center point (TCP) from the center of the robot's tool flange to the end of the tool. This ensures MIRAI accurately judges distances to target objects. For details, see 4 Creating skills in the MIRAI Training App.

7.4.3 Save target poses for a multi-target positioning skill

Before recording each episode, you will save a target pose for each object. This crucial step ensures skill accuracy and may take time. You will also save a target pose for an empty scene to indicate what the robot should do after all objects are picked. Follow this section for guidance on positioning.

Why target poses are key for accurate skills

During training, you will carefully position the gripper at the target, creating the *target pose*. During skill execution, the gripper is guided to this same position. This is why the accuracy of your skill depends on the quality of target pose training. By saving images of the gripper at the target during training, you provide a visual reference that teaches MIRAI what correct positioning looks like.

Think of it like teaching someone to parallel park. A good instructor demonstrates the ideal parking position – not up on the curb or crooked. In the same way, each target pose shows MIRAI exactly how the end effector should be positioned relative to the target.

This precise positioning during skill training gives MIRAI clear, consistent data, leading directly to better accuracy and successful picks during skill execution.

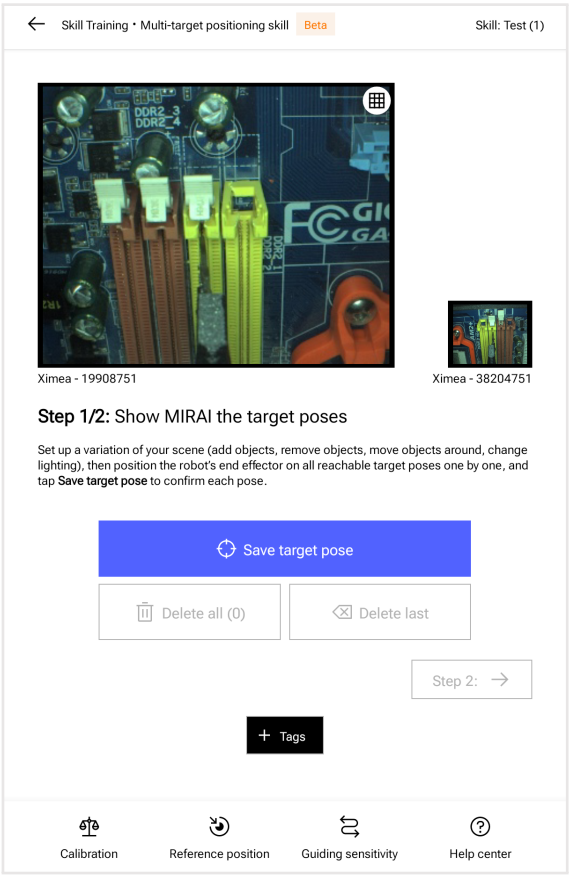
① **Save a target pose for an empty scene**

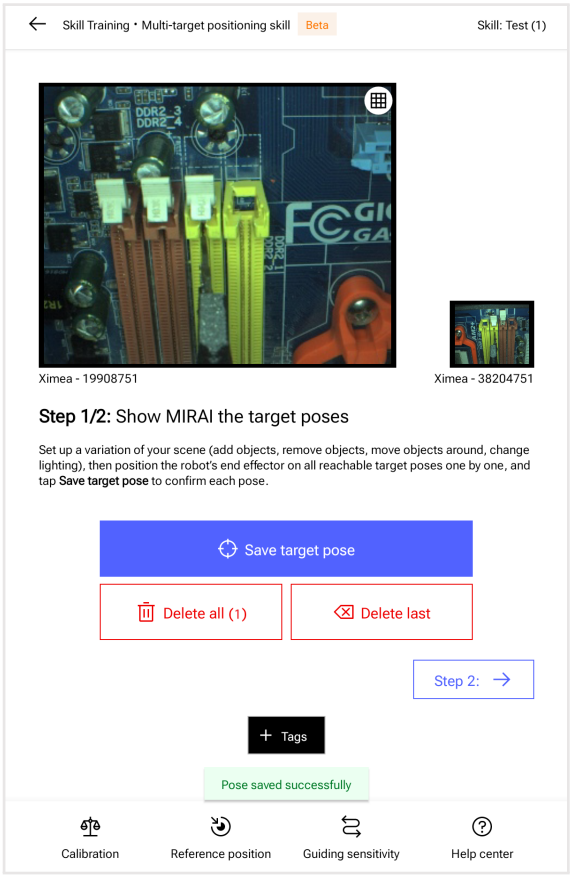
Use an empty scene without any target objects to indicate where the robot should go after picking all objects and where it should wait for the scene to change. Typically, one pose (or one pose per condition) is sufficient at this stage. During skill testing, observe how MIRAI behaves and add additional empty-scene poses if needed.

a. Clear the scene of all target objects (workpieces).

b. Guide the robot to the desired position and tap **Save target pose**.

TIP: Choose a target pose above the workspace to ensure workpieces will be in view.





2 **Save a target pose for each object in the scene**

a. Set up the workpieces in the scene, referring to your training plan.

b. Position the robot's end effector at the first target object, recreating the ideal target pose from your training plan.

TIP: To maintain a consistent vertical distance between the workpiece and the tool, lower the tool tip until it just touches the workpiece. Then, enter the desired offset on the teach pendant to raise the tool by that amount.

c. Tap **Save target pose** to confirm. A green "Pose saved successfully" message will appear.

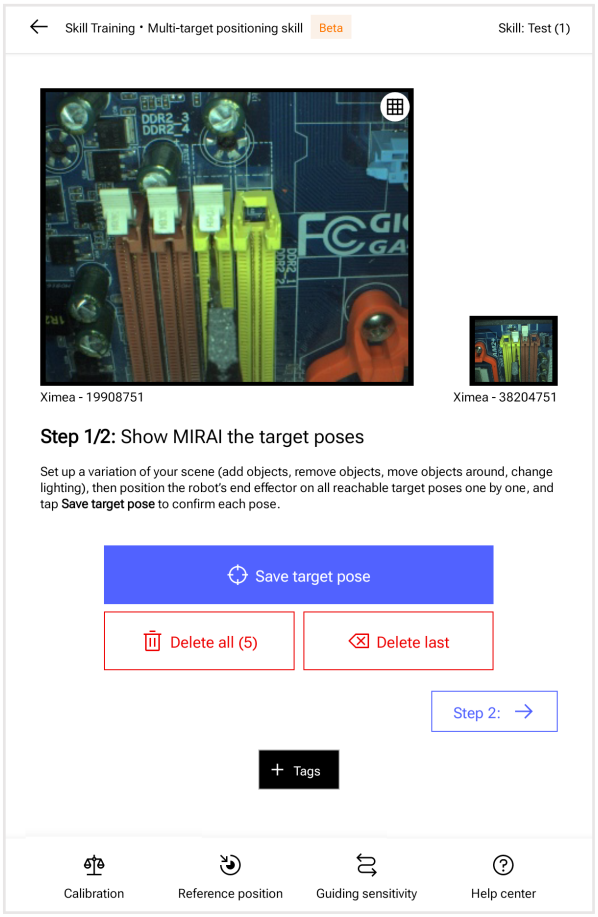
TIP: You can delete a saved pose if it is incorrect (see below).

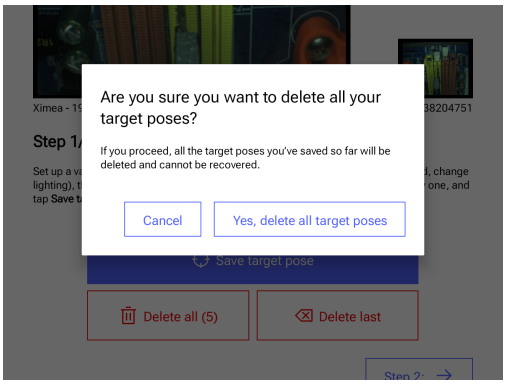
d. Move to the next workpiece. Position the end effector correctly and tap **Save target pose**.

e. Continue until you have saved a target pose for each workpiece in the scene.

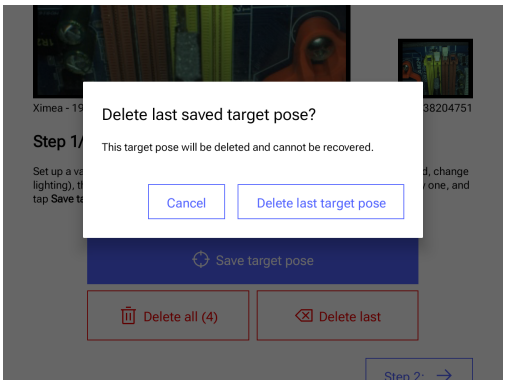
f. Tap **Step 2** to proceed to recording the episode.

> **How to delete target poses**
> If you see that a target pose is incorrect, delete it to ensure skill accuracy. You can also delete all poses – for example, if you decide to change the target pose after saving multiple examples.
>
> - *To delete all poses:* Tap **Delete all**. The number on this button indicates the number of saved target poses. When the confirmation prompt appears, tap **Yes, delete all target poses**.
> - *To delete a single pose:* Tap **Delete last**, which deletes the last pose viewed. When the confirmation prompt appears, tap **Delete last target pose**. You can now save a new target pose.





Confirmation prompt: Delete all poses



Confirmation prompt: Delete last pose

7.4.4 Recording tutorial

After positioning the robot and saving target poses for each object, you will hand-guide the robot while recording data. This section includes detailed technique guidance for effective data collection, including recording practice episodes you will delete. Experienced users may skip steps ⑦ to ⑨ and proceed directly to recording and saving episodes for training.

> Look for **TIPS** throughout this section for practical advice.
>
> Look for **info boxes** for more in-depth guidance to improve training.

WARNING

When hand-guiding the robot, stay aware of your surroundings to prevent injury. Keep your hands clear of pinch points, moving parts, and sharp objects.

1. **Check and adjust guiding sensitivity**

   a. First, test the current sensitivity:
      i. Position the gripper in the target pose above a workpiece.
      ii. Move the gripper in small circles around the workpiece, gradually moving upward and outward.
      iii. If rotations are enabled, rotate the wrist.

   b. Evaluate the sensitivity:

   It should allow these movements to be performed easily and consistently across multiple episodes.

   c. If adjustment is needed:
      i. Tap **Guiding Sensitivity** on the bottom bar.
      ii. Adjust the sensitivity and tap **Apply Settings**.
      iii. Test the movement again and readjust if necessary.

   Note: If you are using rotations, reducing the sensitivity helps achieve smoother trajectories.

   QUICK FIX

   Unexpected movement direction during hand-guiding: The robot moves against you or resists in the opposite direction.
   - **Sensor calibrated while being touched:** Recalibrate the force/torque sensor without any contact during the calibration process.
   - **Incorrect sensor installation:** Verify the force/torque sensor mounting alignment by checking that the printed directional symbols on the sensor exterior (+Y, -X, etc.) align correctly with the robot flange orientation.

2. **Add tags to describe the episode:**

   a. Tap + Tags.

   b. Enter specific keywords ("tags") that describe the episode.

   Why use tags?

   Tags are keywords attached to episodes. Tags give you a quick overview of the content of your episodes and help identify gaps in your recording plan. By using meaningful names, you can better organize, track, and plan recordings across different conditions and scenarios.

   Common tags for MTP skills include:
   - Workpiece arrangement
   - Number of workpieces
   - Lighting conditions
   - Background scene
   - Person recording

3. Tap **Start recording**.
   - The button will turn red and display a timer to indicate that recording is in progress.

④ **Begin by hand-guiding the robot arm around the scene.**

This episode will be deleted. Its purpose is to help you get comfortable with the recording technique.

a. From the starting position, guide the robot above the workspace, showing the entire scene.

b. Release the arm, allowing MIRAI to identify and move to the closest target.

> **TIP:** Use a near-to-far strategy: End your scan of the scene above the target nearest to the handover position, and proceed farther away is each object is targeted.

⑤ **Focus on the single target object in view:**

a. When the robot has moved to the target pose, take hold of the arm and focus on the single workpiece directly beneath the gripper (see **Hand-guiding fundamentals**).

b. As you guide the arm, focus on the following:

- Keep the entire target in the camera frame.
- Keep only a single target object in view.
- Stay close to the target for most of the recording time (see **Recording time allocation**).
- Use a **narrow spiralling motion** to move upwards above the target (see **Hand-guiding technique**)
- If you have rotations, rotate while spiraling.

c. After you have recorded the target, spiral upwards to the top of the workspace, moving the robot to capture the entire scene.

d. Release the arm and again let it go to the closest (new) target (see **Release and return technique**)

e. Repeat until you have recorded data for each object on the workspace.

> **Hand-guiding fundamentals**
>
> **Dual awareness:** Keep one eye on the workspace to avoid collisions and another eye on the camera view to ensure the target remains in view.
>
> **Hand placement:** Place your hand just below the force/torque sensor along the robot arm. Holding the wrist near the force/torque sensor ensures that even tiny motions from the tip of the tool/gripper are properly captured in the training. Reaching from higher up the arm will be more difficult to steer and may result in jerky or unpredictable pathing.
> IMPORTANT: You must delete episodes that show your hand in the camera view. Be especially careful to avoid this while rotating the arm.

> **Recording time allocation**
>
> **Maximum episode duration:** 3 minutes
>
> Follow the 40/30/20/10 rule for distance-based time allocation:
> - 40% of time up close (< 5mm from target)
> - 30% of time near (< 10mm from target)
> - 20% of time at medium distance (< 50mm from target)
> - 10% of time far away (> 50mm from target)
>
> ***Note:*** This guideline is most important when recording your first 20 episodes, but becomes less critical as you record more episodes and build up your dataset.

Hand-guiding technique

Goal: Focus on a single target, keeping it and only it in view.

General Motion Strategy: From the target position, slowly move upward in a narrow spiraling motion. The narrow spiral aims to prevent other workpieces from entering the camera view. If rotations are enabled, constantly rotate the tool to show the target from different angles.

Release and return technique

When you let go of the robot arm, it will move to the closest target via the shortest direct path. This technique:

- Provides a preview of the final trajectory from a given position
- Shows the system direct paths to the target
- Helps introduce more variance around the target per episode

6. When you are finished, tap **Stop recording**.

7. **Discard this episode:**

   - Tap **delete**.
   - You will then be prompted to move the robot to the reference position and to calibrate the force/torque sensor.

8. **Now try a few practice episodes:** Aim to identify necessary adjustments to your setup or technique.

   a. Tap **Start recording** and walk through an episode from your training plan.
   b. When you are done, tap **stop recording**.
   c. Then tap **Discard** to delete the episode.
   d. Try recording an episode for a different condition to get a feel for the variance you intend to show.

9. **Evaluate your experience and make any needed adjustments.**

   - Did the recording go as expected?
   - Do you need to change anything, such as your timing or guiding technique?

   After you are satified, move on to recording the episodes to train your skill.

10. **Record episodes to train your skill:** Follow steps 2 to 6.

11. Save or discard the episode:

    - Tap **Save** to keep the episode.
    - Tap **delete** if the episode shows anything that would not be seen during production.

When to delete episodes

To ensure MIRAI learns correct behavior, delete any episode showing actions or visuals that wouldn't occur in production. Remember: MIRAI learns from what it sees, so unintended behavior can lead to poor performance.

Stop recording and delete the episode if any of the following occur:

- A hand or any unintended object enters the camera view.
- The target object is missing from the camera view.
- When focusing on a single target object at close range, another target object enters the camera view.
- The target object is moved during recording.
- Lighting changes abruptly during recording.
- The robot reaches joint limits or enters a singularity.
- The tool collides with the target or any fixture.

You will then be prompted to move the robot to the reference position and to calibrate the force/torque sensor.

⑫ **Repeat the process to record further episodes**.

Record 15-20 episodes, not more. Use the episode counter (top right of the screen) to track your progress.

Create variations in your scene:

- Record scenes with relevant objects:
  - Start with many pickable objects in the scene. In subsequent episodes, vary and lessen the number of objects.
  - Move objects around between episodes to cover the full range of possible object positions. This can be done systematically (e.g., moving objects row-by-row on an imagined grid) or randomly (e.g., if object positions are determined by an upstream process component).
  - If the skill deals with varying appearances of objects, such as corrosion patterns or different colors and shapes, change the setup to cover a wide variety of appearances.
  - Change the lighting conditions.
- Record scenes with only non-relevant objects.
- Record at least one empty scene (i.e., with no relevant objects present).

When finished, tap the home button to return to the Skill Details page.

Continue working through your recording plan. After you have recorded your initial episodes, send your data to the Micropsi cloud to create the first version of your skill.

7.4.5 Troubleshooting for multi-target positioning skills

When you test your skill, you might see behavior you want to fine-tune for skill execution. This section describes common issues and how to resolve them.

Error	Possible causes	Solution
The robot doesn't move in a perfectly straight line – it makes slight side-to-side adjustments.	Some swaying is normal, becuase MIRAI continuously updates the path to the nearest target in real time. Note that MIRAI's perception of the closest target may differ from what a human sees.	This is expected behavior—no action needed.
The robot does not rotate toward the target using certain axes of rotation/joints	This indicates that MIRAI lacks training data for specific rotational axes when approaching from certain positions.	**A.** Verify all required degrees of freedom are enabled in the skill configuration. If missing, create a new skill with complete DOF set. **B.** If all DOF are enabled, record additional training episodes demonstrating rotation through all available axes from the problematic approach angles. See Section X.X for detailed training procedures.
The robot does not reach the target.	MIRAI's current view doesn't match the training data closely enough. This can be due to: **A.** Damage to the camera or mount **B.** Changes in workspace conditions **C.** Incomplete training data	**A.** Inspect the camera mount and lens for damage. **B. Check the workspace:** • Ensure lighting matches the training conditions. • Verify the visible background is consistent with training. → If the camera or workspace has changed, restore the original setup. If this is not possible, retrain the skill under the new conditions. **C. Assess the scene and compare it with the training data:** • Note the number of objects, their positions, and appearance in the workspace. • Review the training data for similar scenes. → If such scenes are missing, record new episodes and retest the skill.
The robot approaches the target and then drifts away.	This can be due to: **A.** A change in camera position **B.** Incomplete training data: MIRAI does not have enough information about the areas close to the target and does not know how to recover from drifting near the target.	**A. Check the camera:** • Use the camera realign feature (see 14.2). **B.** Record new training episodes, focusing on the area around the target.

Error	Possible causes	Solution
The skill *consistently* places the robot tool off-target.	Systematic targeting errors indicate changed conditions or imprecise training data. This can be due to: **A.** Changes in lighting conditions **B.** Camera focus or aperture drift **C.** Different part appearance **D.** Inconsistent training data	**A. Check lighting conditions:** • Compare current lighting to training conditions. • Verify consistent illumination across the workspace. → If lighting has changed, record new episodes under current conditions. **B. Inspect camera settings:** • Check set screws for focus and aperture settings. • Verify camera mount stability. → If the camera has moved, use camera realignment feature (see 14.2). **C. Verify part consistency:** • Compare current parts to those used in training. → If parts differ from training, record new episodes with current parts. **D. Review training data:** • Review target poses across episodes and check for inconsistencies in position or lighting. → If episodes are inconsistent, delete them and create new skill version.

Error	Possible causes	Solution
The skill suddenly stopped working.	When a previously working skill begins missing targets or showing drift, this indicates: **A.** Camera or mount issues **B.** End-effector tool problems	**A. Inspect camera system:** • Check and tighten camera mount screws if loose. • Verify focus and aperture screws are secure. • Inspect camera lens and housing for damage. → After any camera adjustments, use camera realignment feature to restore original configuration. See 14.2 **B. Check end-effector tool:** • Inspect tool for physical damage or wear. • Compare current camera view of tool and background to training conditions. • Verify tool mounting is secure and properly aligned. → If tool or mounting has changed, restore original setup or replace damaged components.

8 Training a Motion Skill

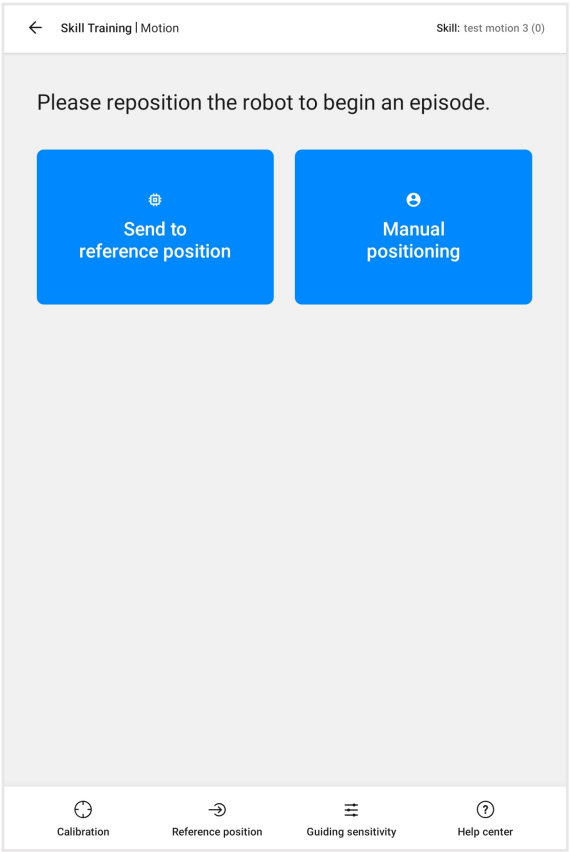
8.1 Before Recording an Episode

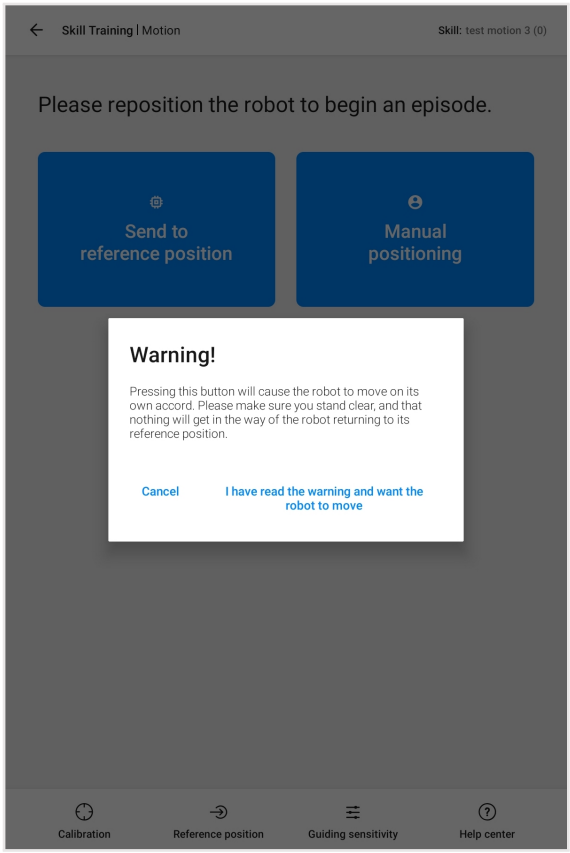
WARNING

During the training process, robot movement can be controlled by hand-guiding. If **hand-guiding is activated**, the robot can be moved in any direction. Please pay attention to not squeeze your free hand and fingers by accident or shear any body parts. Be especially careful when the robot is handling workpieces with sharp edges or pointed contours.

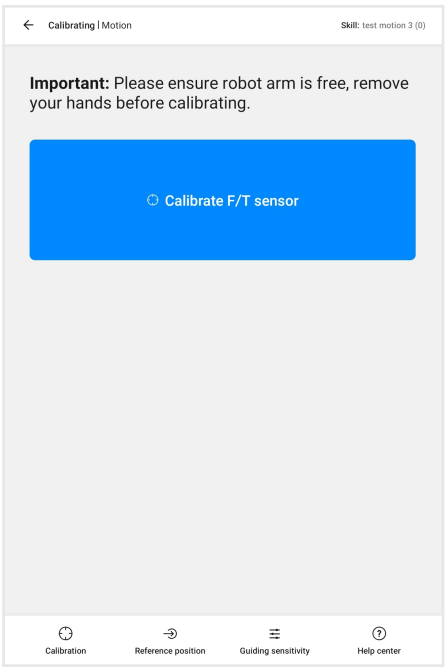
On the "Details" screen, tap **Record episodes** to enter the training loop.

Before beginning to record a new episode, please **ensure that the robot is in a correct starting position**, either by sending the robot automatically to the predefined reference position (**Send to reference position**) or manually guiding it to a user defined starting point (**Manual positioning**).



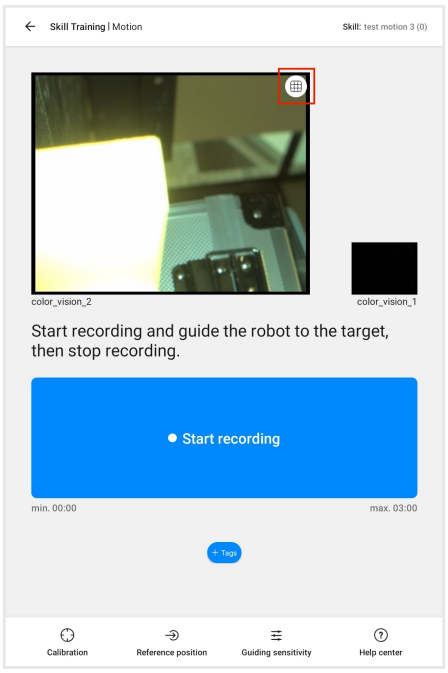


(13) Remove hands from robot to **calibrate the force/torque sensor** and tap **Calibrate force/torque sensor** to proceed.



14 Before recording, **check the image preview** and guide the robot through the intended trajectory, keeping in mind the following points:

- Make sure that during the entire path, the target object(s) and the tool are in the camera frame.
- Ideally, keep distracting objects out of the view, such as hands or task-unrelated objects. (While it is possible to train reliable skills even in the presence of distractors, it comes at the cost of increased demonstration effort).
- You can adjust the guiding sensitivity by tapping **Guiding Sensitivity** on the bottom bar. Please see section 8.3 Guiding Sensitivity for more details.
- The camera grid feature  (found at the top right corner of the camera view) can help with positioning the robot more accurately or easily.



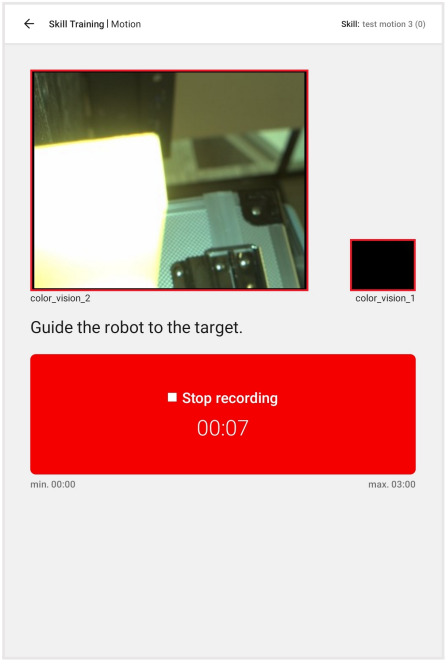
⑮ Tap + Tags to add a text label. You can tag the episodes or series of episodes with meaningful names that best describe the condition, environment, or the object. This will help you plan and keep track of your recording steps. Please see section 8.5 Adding tags (keywords) to episodes for more details.

8.1.1 Recording Episodes

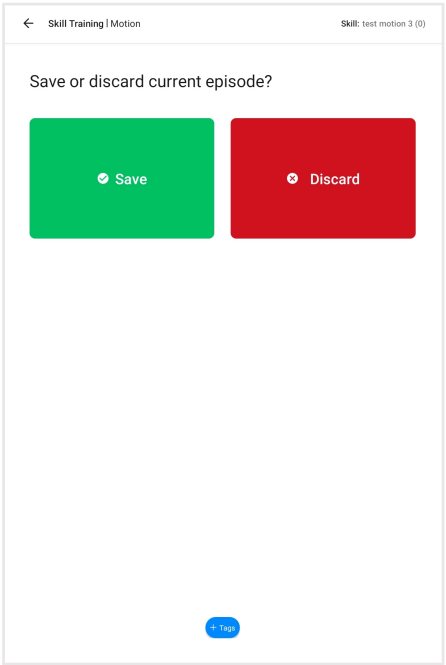
① Tap **Start recording**.

② Immediately, guide the robot to perform the desired motion. Make sure that **the movements are smooth, direct, and straight**.

③ Tap **Stop Recording** after the task is done. The counter gives you an indication of the total time recorded. The recordings are usually shorter; the maximum limit for recording is 3 minutes.



4 If the episode recorded is optimal, tap **Save** to save it; if not, tap **Discard** to delete it.



Discard the episode if:

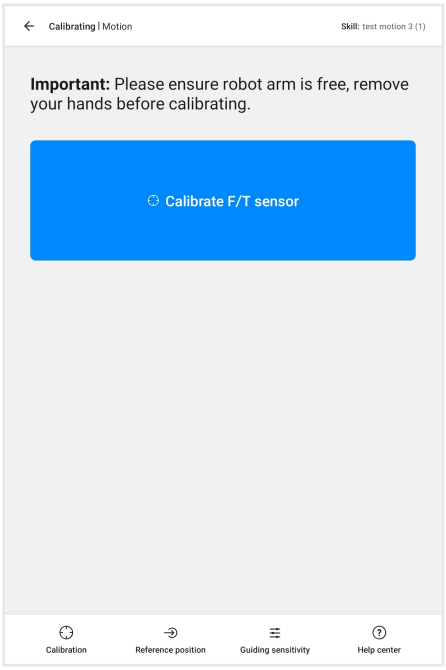
- the task could not be completed.
- there were any pauses in the motion, or if the motion contained unplanned detours/deviations from the direct path.

The system will try to imitate all recorded motions faithfully - so smooth recordings yield smoothly running production skills.

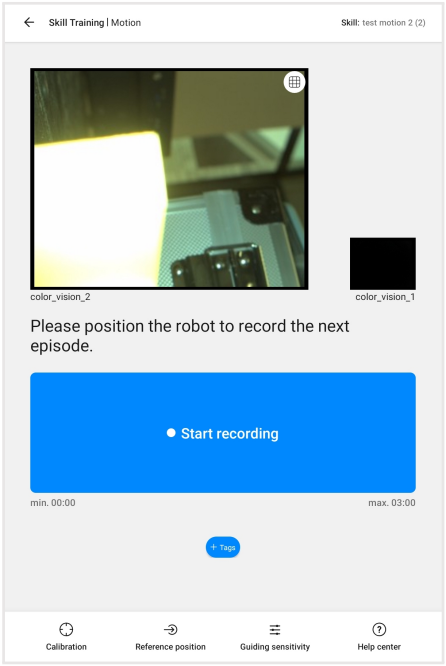
Note

The **episode counter** (on the top right corner of the screen, in brackets) shows the number of good recordings collected so far.

5 **Calibrate the force/torque sensor**. This measure is required after every episode to avoid drifting during guiding.



6 **Choose a different starting position** and tap **Start recording** to proceed with the next episode. The number of successfully recorded episodes is indicated on the top-right corner.



Ensure that **all variances are captured** in order to derive a meaningful and functional skill. Capture relevant variances in around 150-200 episodes to expose the skill to the spectrum of changes and variances that are expected to be seen in the setup during execution (this usually takes about 2-5 hours, depending on the complexity of the task, the skill of the trainer, and the breaks the user takes).

7. After recording the desired number of training the episodes, **leave the training loop** through the home button (top left corner) which opens the "Details" overview screen.

Word of advice:

Experience shows that the first few recordings could be difficult for users who are new to the system. **Once you find your routine and what works for you, the process becomes much easier!**

8.1.2 Training Tips

To achieve good results (a MIRAI skill that is effective and robust in handling dynamic changes and variances within the setup), it is important to pay attention to how to train the robot and capture this through the camera(s) across various recordings. The initial episodes you record will teach MIRAI to find the target position. Subsequent episodes will teach MIRAI how to find the target while handling variances in the target, as well as variances in the environment.

Following is an overview of training tips for demonstrating and recording episodes:

- **Each episode should include some form of variation** for MIRAI to learn from. MIRAI does not learn from being shown the same thing twice. Consider the sources of variance in your application (for more details see section 8.2 Sources of Variances).
- **Hand placement:** Place your hand just below the force/torque sensor along the robot arm. An episode will fail if your hand falls into the field of view of the cameras - be very mindful of this while rotating!
  - Holding the wrist near the force/torque sensor ensures that even tiny motions from the tip of the tool/gripper are properly captured in the training. Reaching from higher will be more difficult to steer and may result in more jerky or unpredictable pathing.
- **Pick a consistent strategy and stick to it:** For best results, avoid showing the AI two different paths to reach the same target. When you show the system more than one path, it can't decide which way is best and ultimately will try to blend these two paths together creating a "middle path" and fail. We recommend moving along the shortest path from the starting position to target position, whenever possible.
- **Move with a consistent speed:** MIRAI will pick up any inflections in speed along a path or during a motion. For example, when moving the tool, it accidentally moved too quickly in the beginning and then changed to a slower rate towards the end of the episode - the AI will copy this motion identically. If your desired result is to maintain a smooth and consistent rate of speed, this will need to be properly demonstrated in your training episodes.
- **Be ready to 'Stop recording':** When setting up your workstation, tablet placement will help your trainings run much more efficiently and lead to fewer failed episodes. We recommend placing the tablet within reach so that you can quickly tap "stop recording" as soon as your episode is complete. This will be especially important when training motion skills. Any delay at the end of the episode will be mimicked when performing the skill later. For this reason, make sure to tap **Stop recording** immediately after the task is done.
- **Avoid random motions and actions** that are not related to the task. To achieve effective trajectories (fast, shortest path possible), movements and trajectories demonstrated in the episodes should represent these trajectories. The MIRAI skill will reflect the trajectories that will be recorded.

  - Do not record trajectories that are non-optimal (including unnecessary pathways) or that vary highly from one recording to the other (in case there is no reason for doing so).
  - Do not move the robot tool into areas and orientations where it should never go while recording episodes.
- **Do not collide with the target** during training. This is likely to displace the target and the tool, creating bad data.
- **Train for overshoots:**
  - Show MIRAI what it looks like to overshoot the target. For example, what it looks like to move past the target, above the target, below the target, to the side of the target, or around the corner from the target etc. Start from unlikely positions. Showing MIRAI what an overshoot looks like is the key to teaching it how to recover from an overshoot.

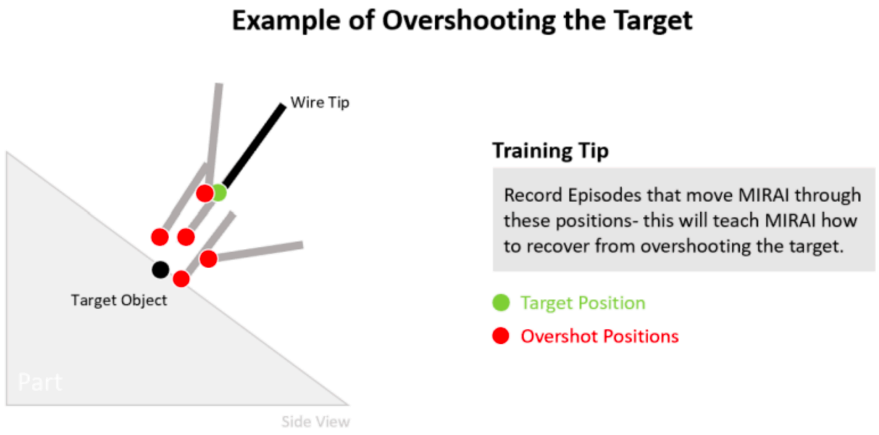


Figure 38: Overshooting the target

  - For about a third of the recordings, place the robot in a starting position slightly besides the ideal path. Example: In a task that involves touching a target object, an ideal motion path is one that does not overshoot, i.e., the robot never travels beyond the target. Nevertheless: Do start some recordings from positions slightly beyond the target - demonstrating how to find back to the target, just in case.
  - It is important to **start these 'recovery recordings' with the robot already positioned in a point besides the ideal path** – if instead, you start from the same start position as in normal episodes, only deviating into the weeds somewhere along the way, then the system would learn to perform this deviation during normal operation, which would be undesirable.
  - In other words: During the setup phase, pretend that a small mistake has already happened. Then, during recording, teach just how to correct it.
- **Save backups of your MIRAI controller:** It is good practice to occasionally create backups of your MIRAI controller – especially once you have developed your skill to a satisfactory level. In the event that your MIRAI controller gets damaged or destroyed, it is helpful to have a backup of your MIRAI controller so that it may be easily restored. Creating a backup requires a USB flash drive – see chapter 16 Backup and Restore.

8.2 Sources of Variances

In order to receive a robust skill that deals with the variances in the setup and the task, it is required to capture these variances (ideally including their border cases) across the training episodes. The following are

some sources of variances that are often encountered in automation tasks:

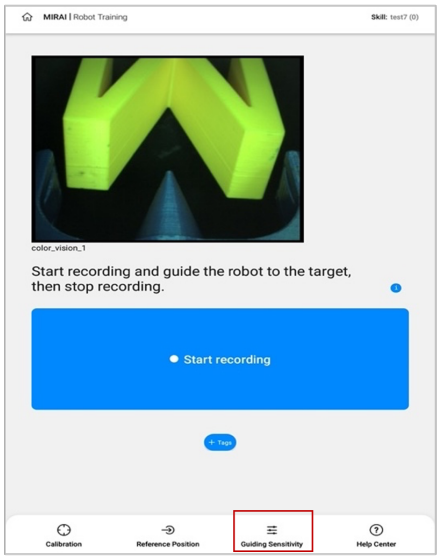
- Variations in the **start and end points** of trajectories in the task
  - Will the skill deal with varying positions of certain objects, relative to the camera? Note the range of possible object configurations that might occur in production. What are the extreme positions? Change the object positions between recordings, so that the recordings cover the full range of possible object positions in between those extremes, without 'gaps'. This can either be done systematically (e.g., moving objects row-by-row on an imagined grid in small increments) or in random order (e.g., if object positions are determined by an upstream process component).
- Variations in the **color or/and shapes** of the working parts
  - Will the skill deal with varying appearances of certain objects, for example, varying corrosion patterns, varying colors, or shape irregularities? If so, regularly change the setup in between recordings so that the 200 recordings cover a wide variety of object appearances or shapes.
- **Changing background** or moving objects in the background (only relevant if the background is visible in the recorded episodes)
- Changes in **lighting conditions**, e.g. changing daylight, etc. Direct exposure to sunlight will likely have a negative impact on the robustness of the skill and should be avoided if possible.
- Imprecisions and variations in **gripping positions** at the TCP

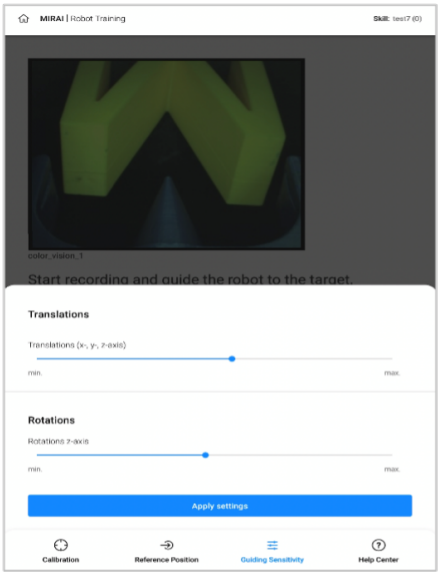
8.3 Guiding Sensitivity

Note

This feature is not available for skills which are operating **without a force/torque sensor**.

You can change the guiding sensitivity according to your preferences. Tap **Guiding Sensitivity** on the bottom bar. Start recording screen (before recording) to adjust the guiding sensitivity for rotational and translational movements. For skills with rotations, you might want to make it less sensitive (stiffer) in order to get a smooth and controlled trajectory. Tap **Apply Settings**, and then move the robot arm to check if the settings are fine.



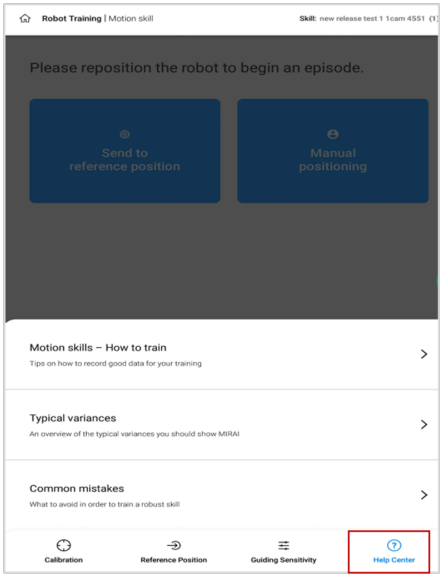


Note

Do not use the maximum sensitivity with large robot models like R-2000iC/165F/210F and R-2000iC/270F.

8.4 Help Center

For tips on training motion skills, showing MIRAI variance, and avoiding mistakes, tap **Help Center** on the bottom bar.



8.5 Adding tags (keywords) to episodes

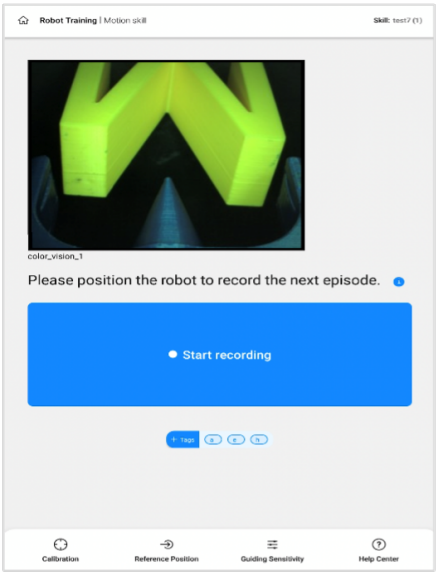
You can add “tags” or labels for each episode you record. Please note that **this is not mandatory for training a skill**, but it is recommended to add tags or labels to better organize and structure episodes. For example, tags can help keep track of the following instances:

- Variance in lighting conditions. E.g., “sunlight”, “spotlight”, “cloudy”
- Variance in the color, shape or appearance of the target object. E.g., “red”, “rectangle”
- Variance in different starting positions or angles, E.g., “left”, “up”, “top-right”
- Changing background or moving objects in the background
- Adding tags under the name of the person who recorded episodes
- “trial” or “practice” episodes
- Tagging “recovery episodes”, where the robot is already positioned in a point besides the ideal path. In other words, show the robot how to correct a mistake.
- Indicating who did the respective recordings in case several people are working on a skill

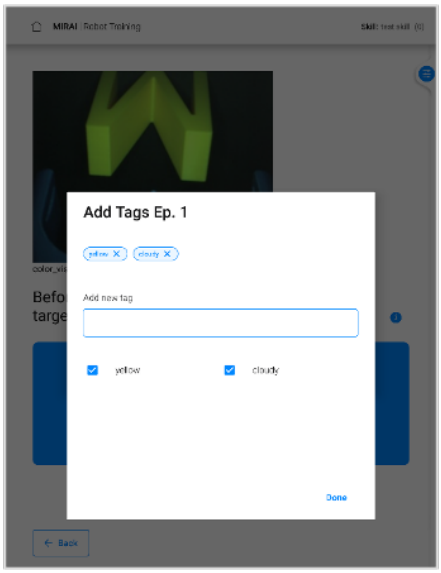
Depending on your task, you may use the same set of tags for multiple episodes or you may use different sets of tags. Each episode can have up to 10 tags, but the total number of tags for a single skill can exceed 10.

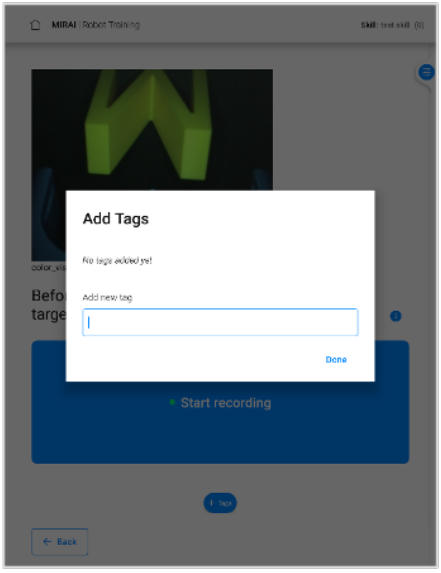
The following steps give you step-by-step instructions:

① Before you start recording a new skill, tap the + Tags button.

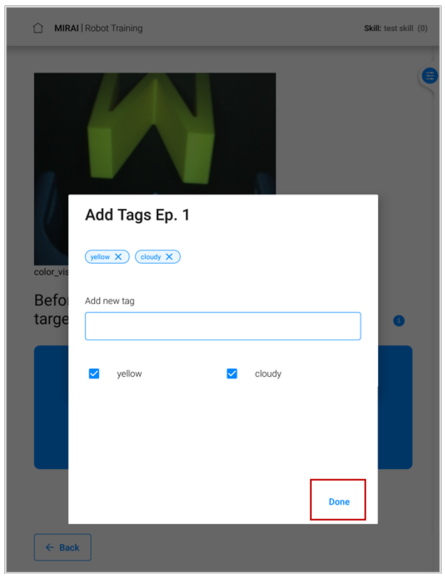


② On the pop-up screen (see below), select the tags from the existing list or add a new tag.

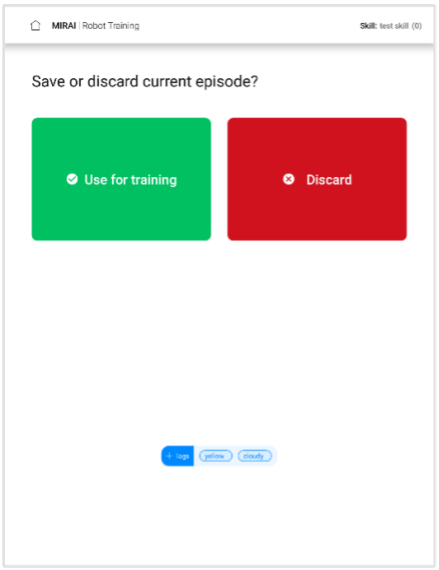


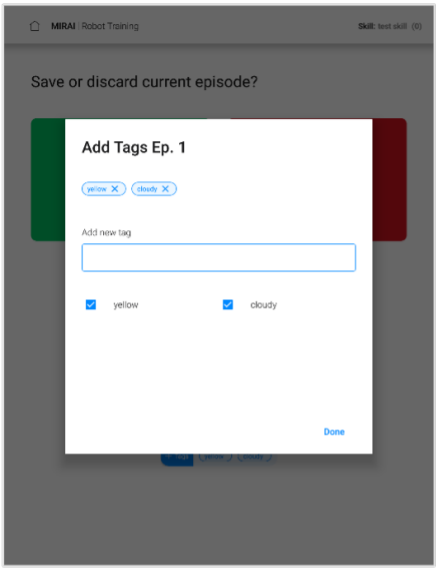


③ The selected tags are shown at the top of the screen. Tap **Done** once you have added all the tags. You can also modify these tags later.

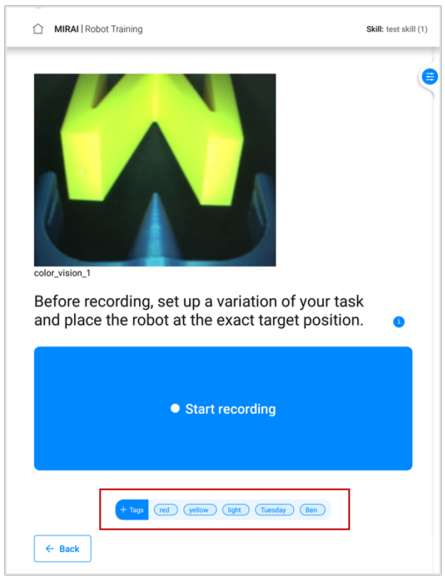


4 Proceed with recording the episodes. You can also modify the tags before saving episodes by tapping **+ Tags**.

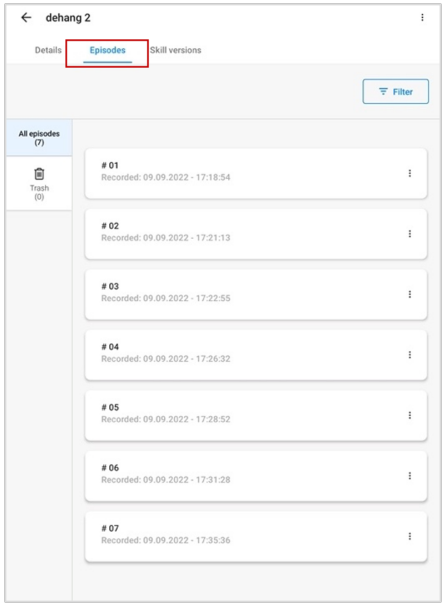




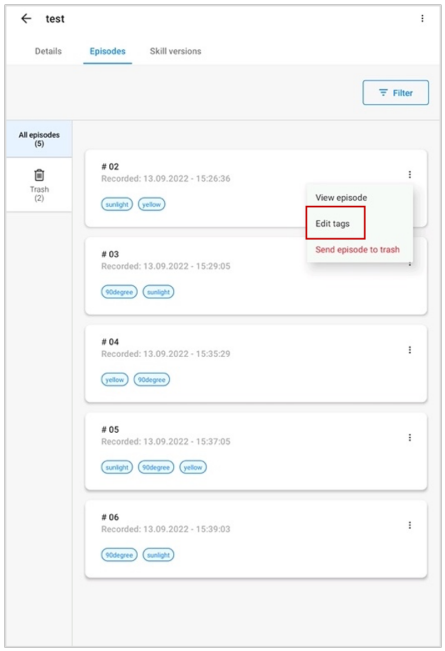
5 Proceed with recording the subsequent episodes. Note that the tags that you selected from the previous episode are automatically carried over to the subsequent episodes. You can add or remove tags immediately before or after the next episode is recorded.

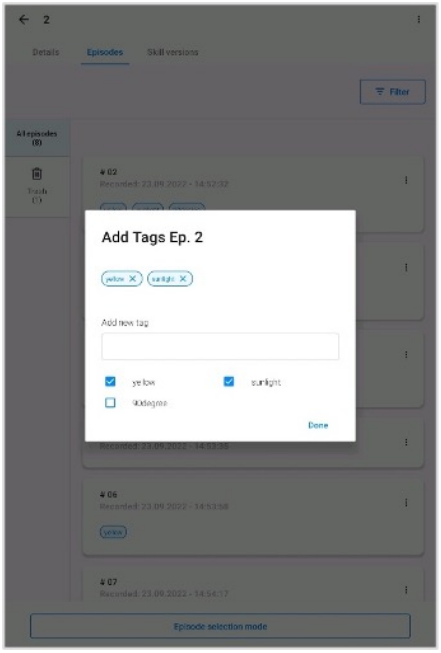


6 Once you have recorded all the episodes, on the skill overview screen, you can review or modify tags. Tap the ”Episodes” tab.

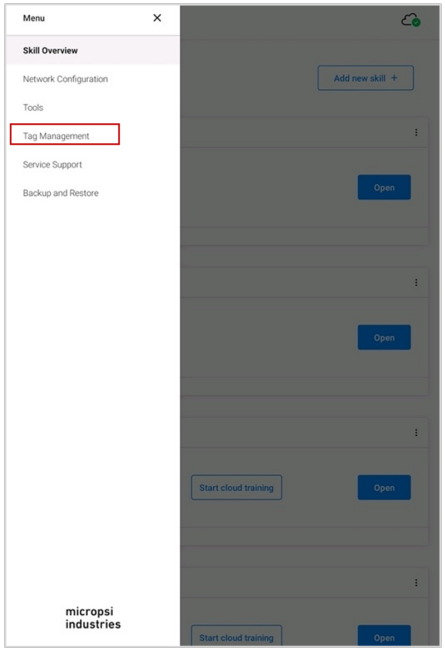


7 Open the three-dot menu and select “Edit tags” to add or remove tags for the respective episode.

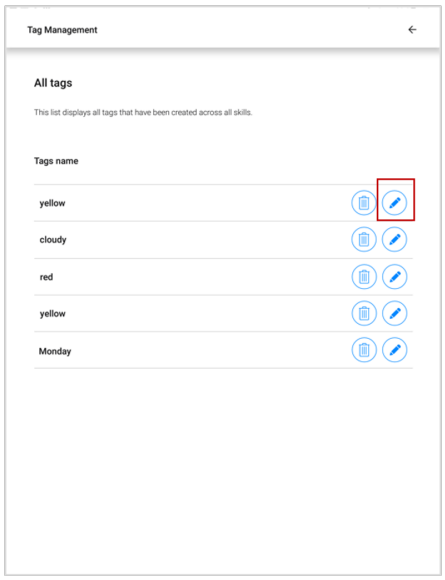


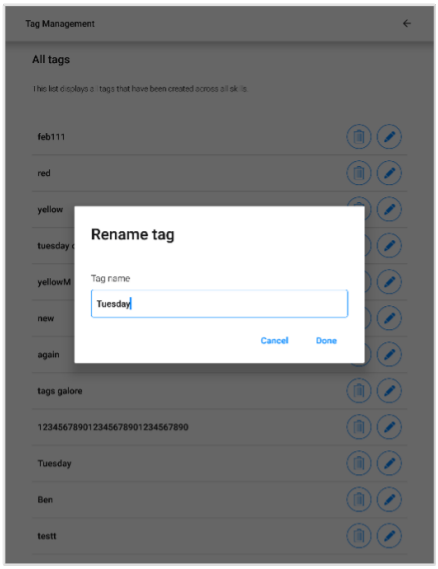


8 To manage tags, on the main screen, go to the top left menu and select **Tag Management**.

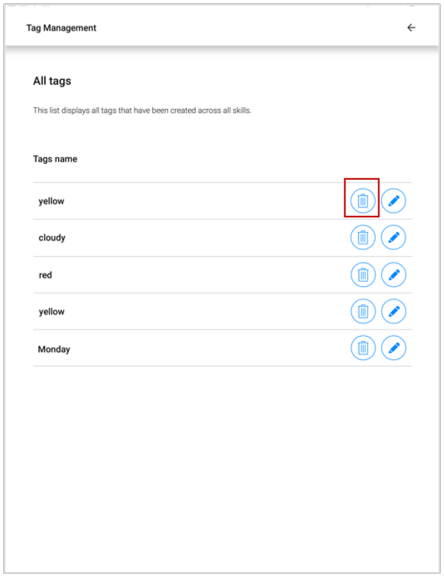


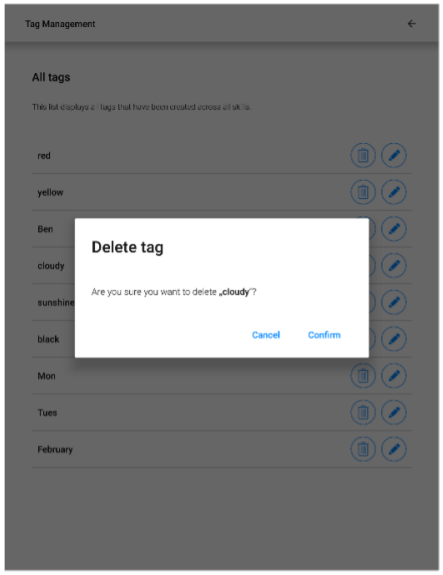
9 Tap **Edit** to rename a tag. Please note that this would mean updating the tag name across multiple episodes, possibly across different skills.

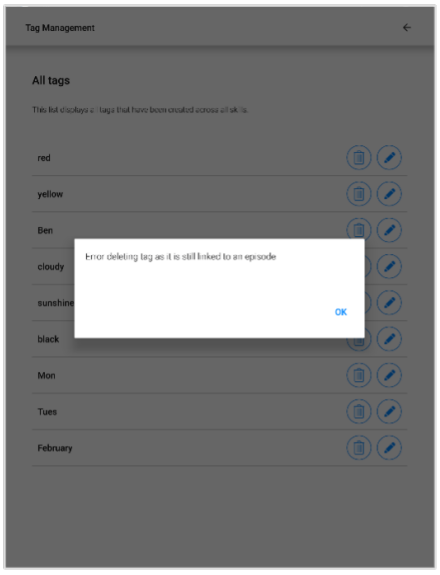




⑩ Tap **Delete** to delete a tag. Please note that you can delete a tag once it is not associated with an episode.




Tag Management
All tags
red
yellow
Ben
cloudy
sunshine
black
Mon
Tues
February
Delete tag
Are you sure you want to delete „cloudy"?
Cancel
Confirm

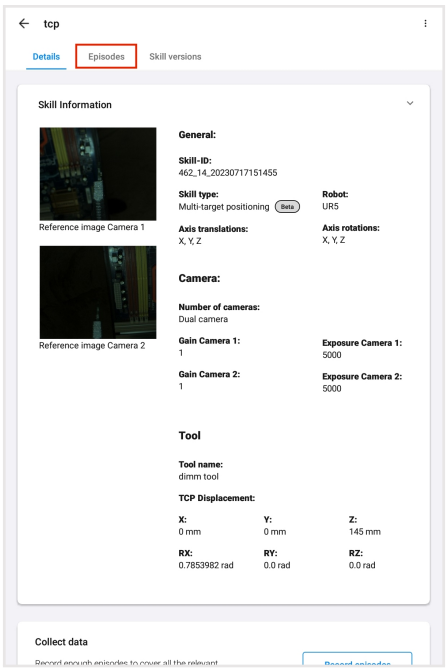

Tag Management
All tags
red
yellow
Ben
cloudy
sunshine
black
Mon
Tues
February
Error deleting tag as it is still linked to an episode
OK

9 Reviewing and Cloud Training

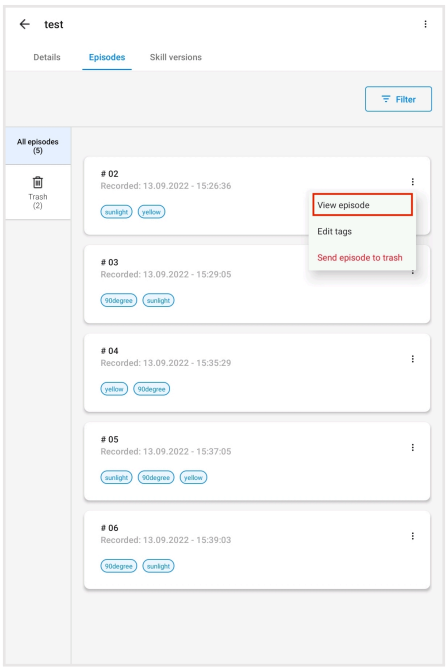
9.1 Reviewing and Rerating Episodes

You can review the recorded episodes by replaying them. This allows you to regrade episodes, based on the potential issues during recording, for example, if the hand was in the camera view, or the lighting was sub-optimal.

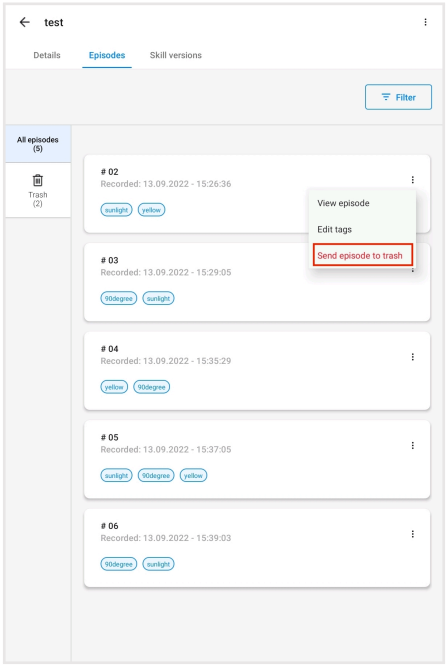
Once you have recorded all the episodes, on the "Episodes" screen, you can view the recorded episodes.



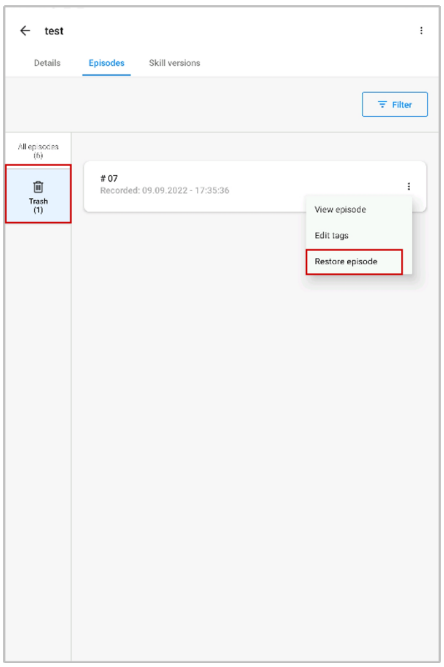
- Select **View episode** from the three-dot menu of the respective episode and it will start playing automatically.



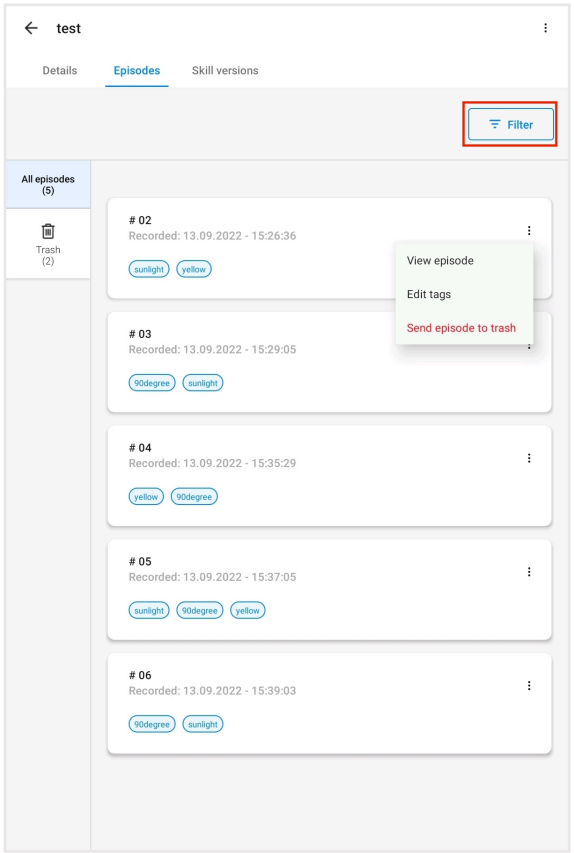
- In case there is some inconsistency in the recorded data, you can discard an episode by tapping **Send episode to trash**.

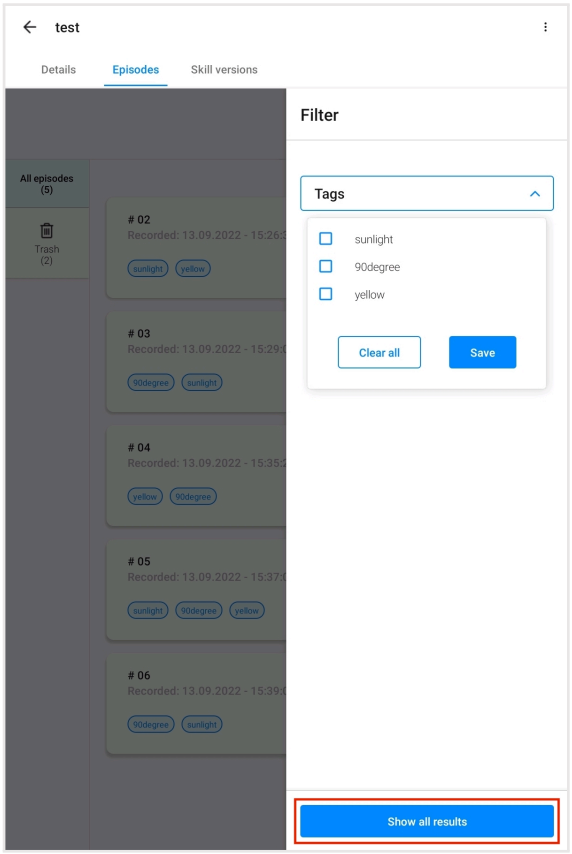


- To recover a discarded episode, go to the " Trash" tab and select **Restore episode** from the three-dot menu.

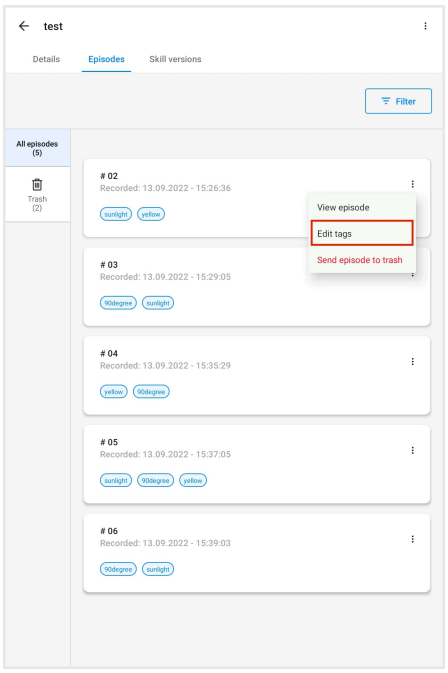


- You can search and focus on a relevant set of episodes by using the Filter function. Filter the episodes according to the existing tags and tap **Show results**.





- To modify the tags, tap **Edit tags**.



9.2 MIRAI Cloud Training and Skill Versions

Important

If you are using a proxy to connect to the Micropsi cloud server, ensure it allows connections to the following hosts:

For training:
https://crunch.micropsi-industries.com (TCP Port 443)
https://crunch.micropsi.io (TCP Port 443)

For support:
mirai-vpn.micropsi-industries.com (UDP Port 1194)

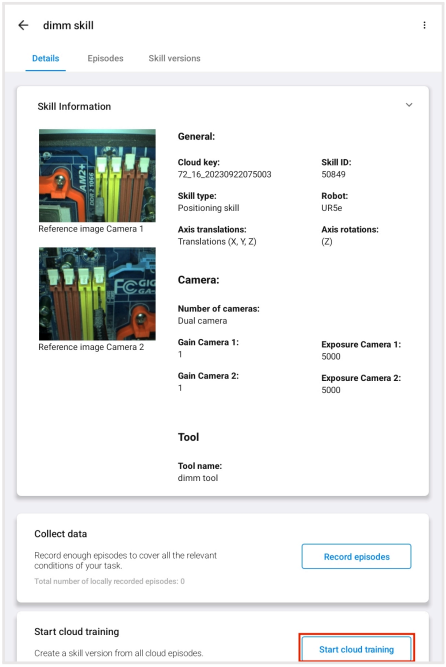
For software updates:
https://apt.tools.micropsi.io (TCP Port 443)
http://deb.debian.org (TCP Port 80)
http://security.debian.org (TCP Port 80)

Details on the cloud connection and how Micropsi Industries handles data are in Chapter 17 Data FAQ.

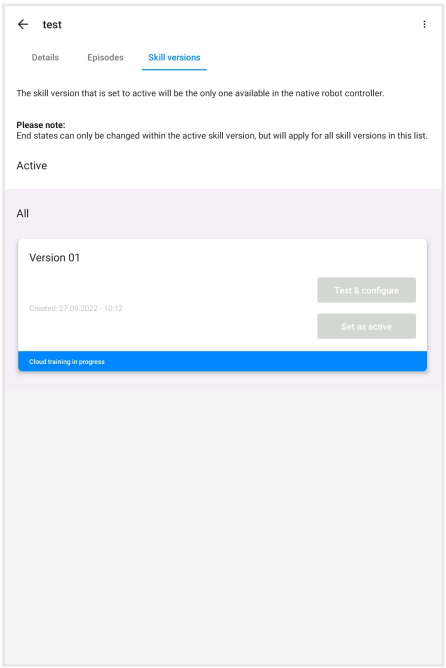
Note

The cloud training button will be active once you have recorded **at least 5 successful episodes** with at least 30 seconds worth of recorded data. The number of recorded episodes (for that version) are indicated in the “Collect data” section.

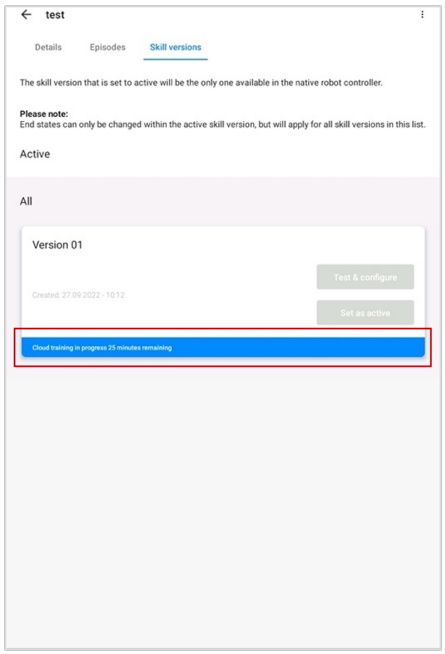
1. In the "Details" overview screen, tap **Start cloud training** to create a new skill version.



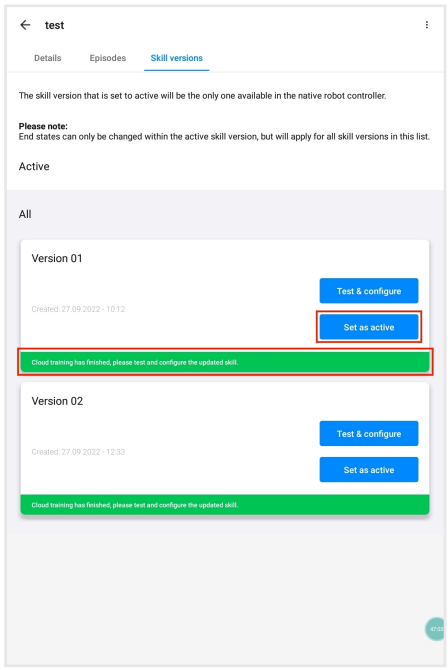
② After starting the training, the respective version will appear on the “Skill versions” screen.



After cloud training has started, this message will appear: "Cloud training in progress." The estimated time remaining will be displayed under the skill version. This process will take approximately 45 minutes. For dual camera skills it is double the time.

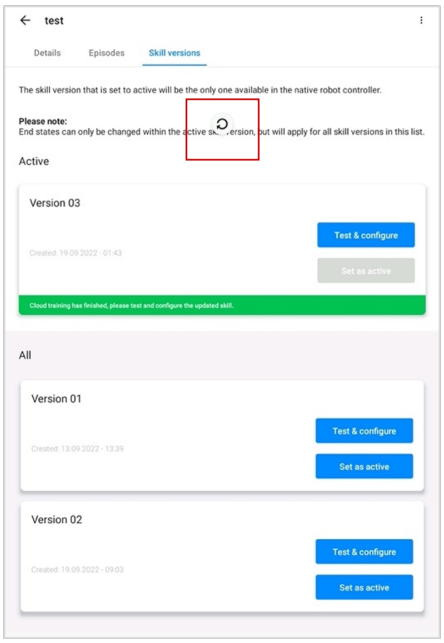


3 Once the skill version is available, you will see the text in green indicating that an updated skill can be tested and activated. “Skill versions” feature enables you to create different versions of the same skill to find out what works best for your use case. With skill versions, it is possible to activate one of the previous versions while still preserving the newer versions. To **activate the version of your choice**, tap **Set as active**.



Note

The displayed status info of each skill is updated every 15 minutes. By pulling the screen downward and releasing it again, you can refresh this information anytime.

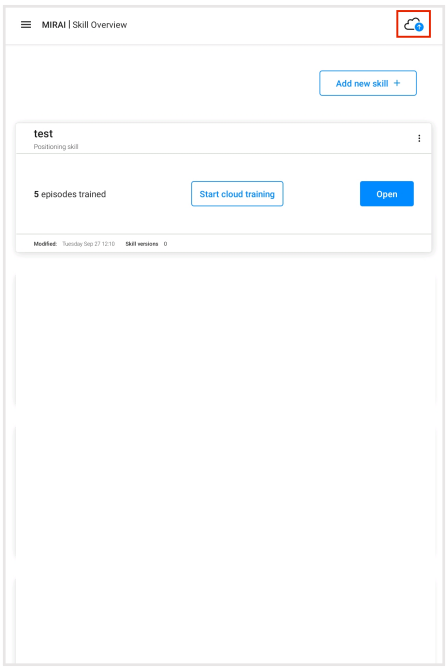


You may encounter the **following exceptions** and will see an error message indicating the reason why cloud training has not been successful. These could be:

- Error due to no data connection – please ensure that the connection of the MIRAI controller to the Micropsi cloud server is working and stable.
- Training data still being uploaded, cloud training will start once data upload is finished – the upload of the training data has not finished, yet. Please try again after a short period. In case this error persists, please check the connection of the MIRAI controller to the Micropsi cloud server.

9.3 State of Uploads

The data is intermittently transmitted from the MIRAI controller to the Micropsi cloud server in the background in case a WAN connection is available. This is indicated by the status icon (“Training data upload in progress”). While the data is being transmitted, WAN/Internet should not be disconnected from the controller, and the tablet should be connected to the MIRAI-xx Wi-fi network of the MIRAI controller.



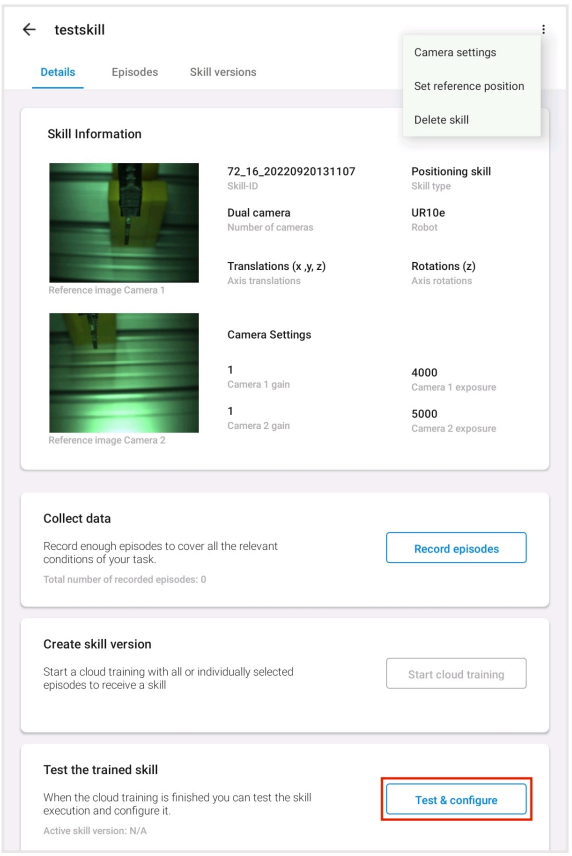
The other possible states are:

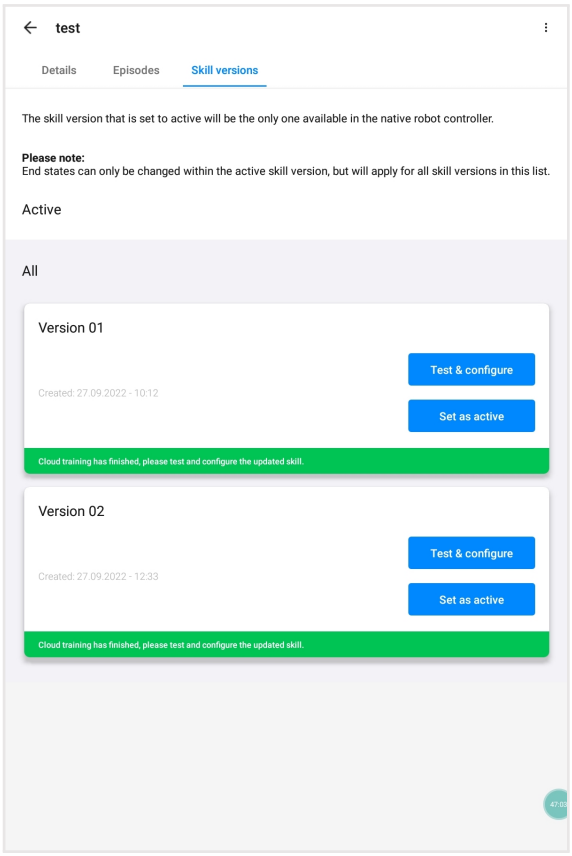
- **Status unknown:** connection issue: Check your internet and Wi-fi settings.
- **No uploads in progress:** It is safe to disconnect WAN/internet from the controller.

10 Testing, Refining, and Embedding MIRAI Skills

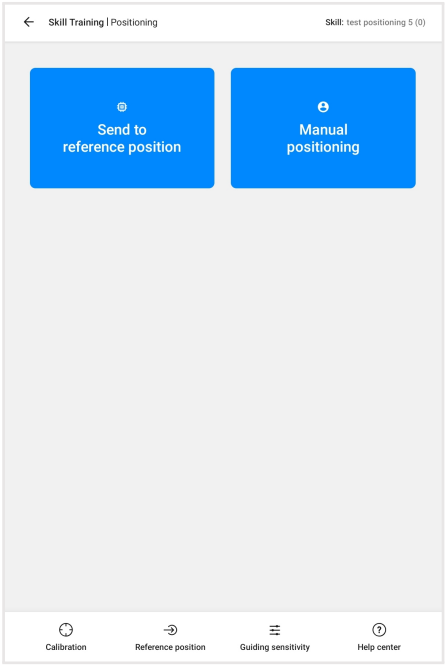
10.1 Testing Newly Trained or Updated Skills

① In the skill detail screen, tap **Test & configure** and choose the skill version you want to execute, test, and configure.





② Before you start testing a skill, **position the robot to a suitable starting point** by either sending it to the reference position (which was saved during skill creation) or manually guiding the robot arm. When working without a force/torque sensor, navigate the robot with the teach pendant instead.



3. *If working with a force/torque sensor:* Similar to the training loop, remove your hands from the robot arm, and ensure it is not pushing against any obstacles. Tap **Calibrate force/torque sensor** to prepare the test session. (This step is omitted when working without a force/torque sensor)

4. **Guide the robot arm to a starting position** that allows testing the new or newly updated skill. Ideally, this position represents a scenario similar to the original starting positions used during training or starting positions that are expected to be seen in the real world. Set up a variant of the task, e.g., repositioning or changing some objects exactly as during the training.

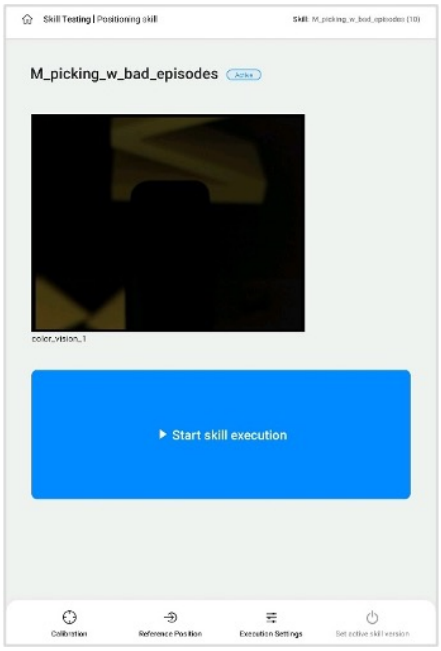
5. Check the **image preview** to ensure visibility, exactly as during training.

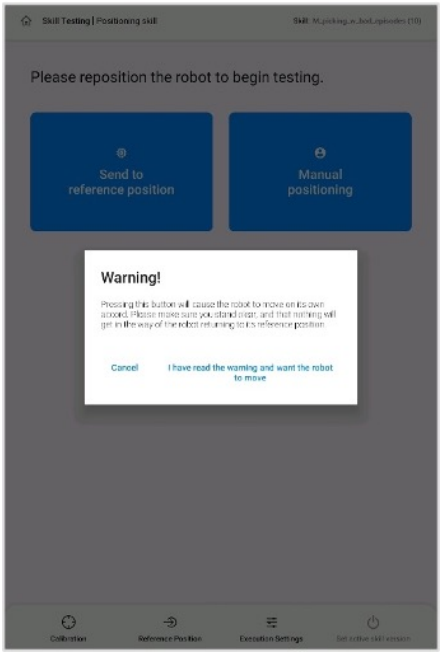
6. The **Execution Settings** (the various options to define end state conditions) are being covered in section 11 End State Triggers.

7. Tap **Start Skill Execution**. A safety warning will appear once to inform that the robot will now be moving, controlled by the MIRAI system. Confirming the warning will start the skill execution – the robot starts moving.

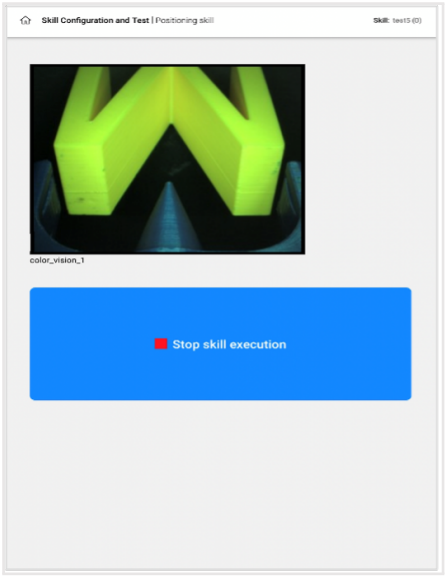
Note

When working with a force/torque sensor in your robot setup: During the whole testing procedure (before and during skill execution), the **guiding feature is always enabled**, allowing the user to manually intervene if needed.

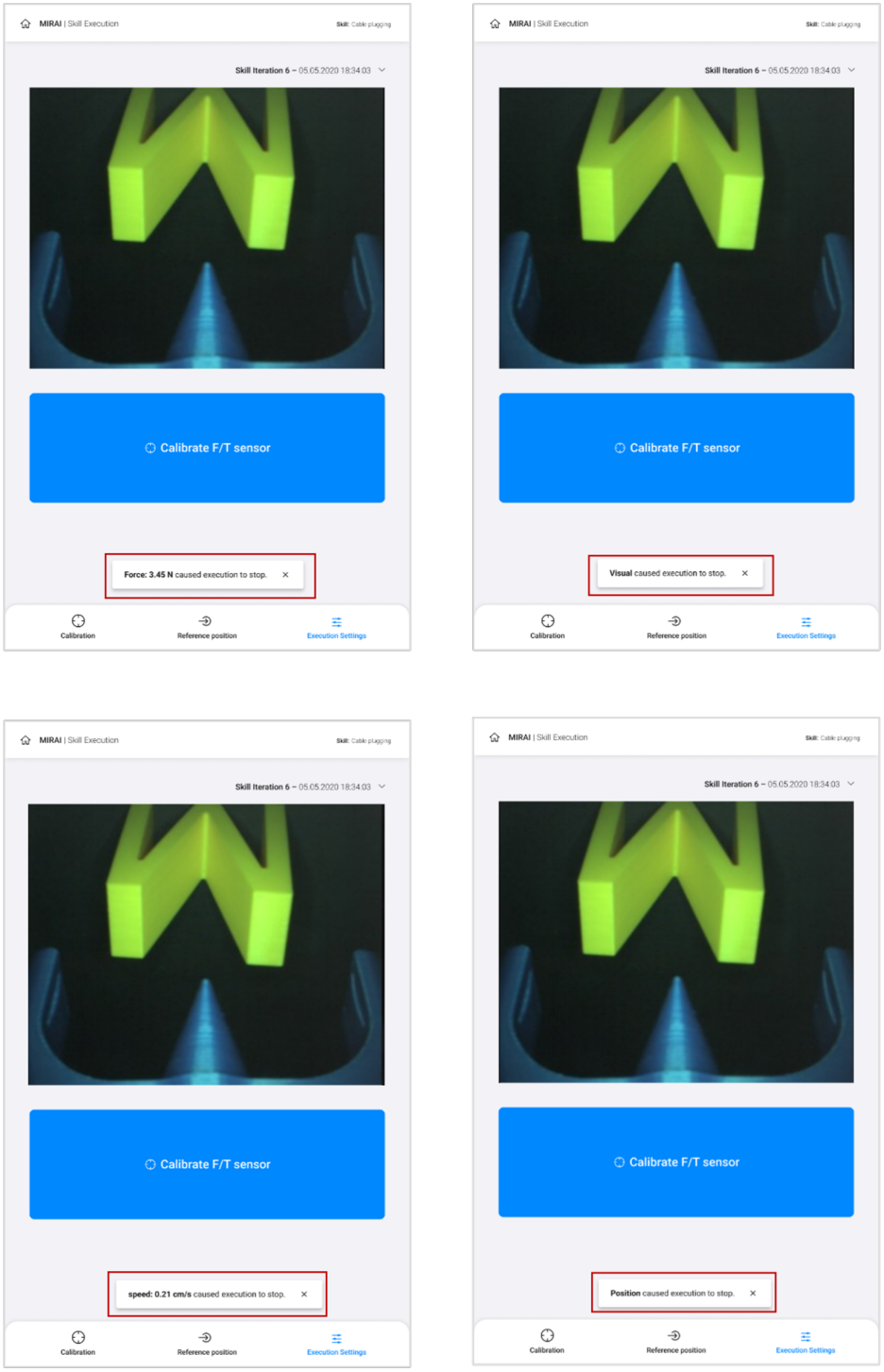




8 **Watch the execution of the skill**. Tap **Stop Skill Execution** to temporarily halt the skill execution. After the execution is stopped, **Start Skill Execution** will appear again, so the skill can be restarted at any time.



9 When the skill reaches **one of the end state conditions**, it will automatically stop. The skill can be restarted at any time. The end state that triggered the skill execution to stop will be displayed at the bottom of the screen with the respective value when available. This information will remain visible until the next test session is started or the “x” to close is tapped. To set end states please refer to section 11 End State Triggers.



⑩ Tap **Calibrate force/torque sensor** to prepare the next test session (if working with a force/torque sensor), and so on.

⑪ If the skill **behaves as intended** in a variety of conditions, it is **ready to be tested in production**.

If the testing phase **shows weaknesses**, this can be remedied through **additional training**, focusing on the situations that need improvement. See section 10.2 Steps to Improve a MIRAI Skill Through Additional Training for details. Or look into these two sections for troubleshooting, 10.3 and 10.4, if you

are experiencing specific problems.

10.2 Steps to Improve a MIRAI Skill Through Additional Training

It is common that a test run (see section 10.1 'Testing Newly Trained or Updated Skills') after the first training round uncovers weaknesses in the new skill. For example, motions might be imprecise in certain task configurations, or may overly depend on a particular lighting situation or object appearance. In all cases, the reason is that the situations in which the skill is still weak are not yet sufficiently represented in the recorded demonstrations. Therefore, the remedy is to record additional demonstrations focused on the situations in which the skill needs help.

- Enter the recording loop.
- Ensure the robot arm is at the correct start position.
- Remove hands from robot & ensure it is not pushing against any obstacles, then calibrate the force/torque sensor using the on-screen button.
- Set up a variant of the task, e.g., repositioning or switching out various objects.
  - If the skill test revealed **weaknesses for one particular kind of situation** (such as one region of the work area, a particular lighting situation or object appearance), then most of the new recordings should show this kind of situation.
  - If the test revealed **overshooting behavior** (i.e., the robot travels towards the target, but instead of reaching it continues to travel beyond it), use half of the additional recordings to train for overshoots - demonstrating how to find back to the desired position. For **Positioning skills** see end of section **?? ??** for more information and for **Motion skills** 8.1.2 Training Tips.
- Check the image preview to ensure visibility, exactly as during the first training round.
- Record a demonstration, as during the first training round.
- Guide the robot back to the start position.
- The process repeats itself with the calibration screen.
- Typically, substantial improvements are achieved with **50-200** additional recordings of this kind for Motions skills and **5-10** for Positioning skills.

Note

The more episodes you record, the shorter each episode can be – it is okay for refinement episodes to be only **30-45 seconds** long.

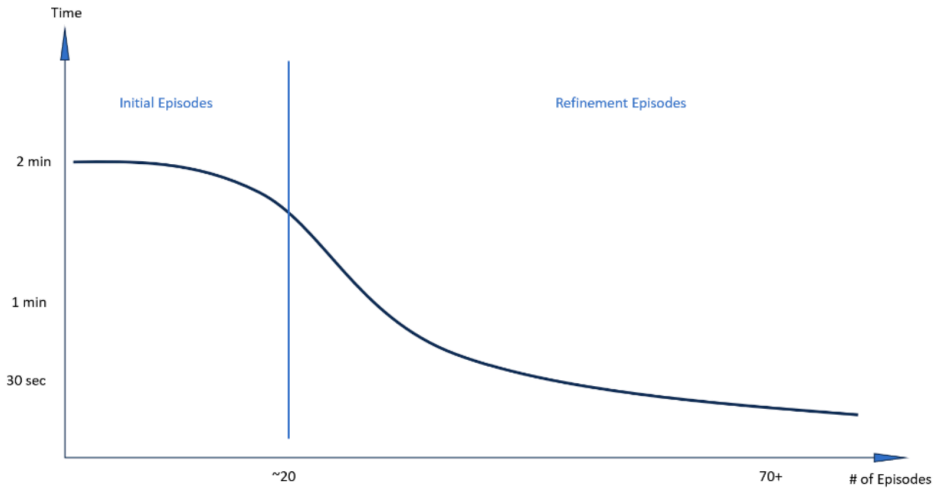


Figure 39: Decreasing length of episodes

- Refinement episodes do not need to cover as large of an area as the initial episodes you record – **refinement episodes can be limited to the specific weak spot or area** of variance that you are trying to teach MIRAI.
- Once the recording of episodes is finalized, **start another cloud training** by tapping **Start Cloud Training** in the respective skill section. The status will change to "Cloud training in progress" till the new skill update is delivered.

Word of advice:

The **confidence and competency** of the skill will increase with the **number of good episodes** that you record.

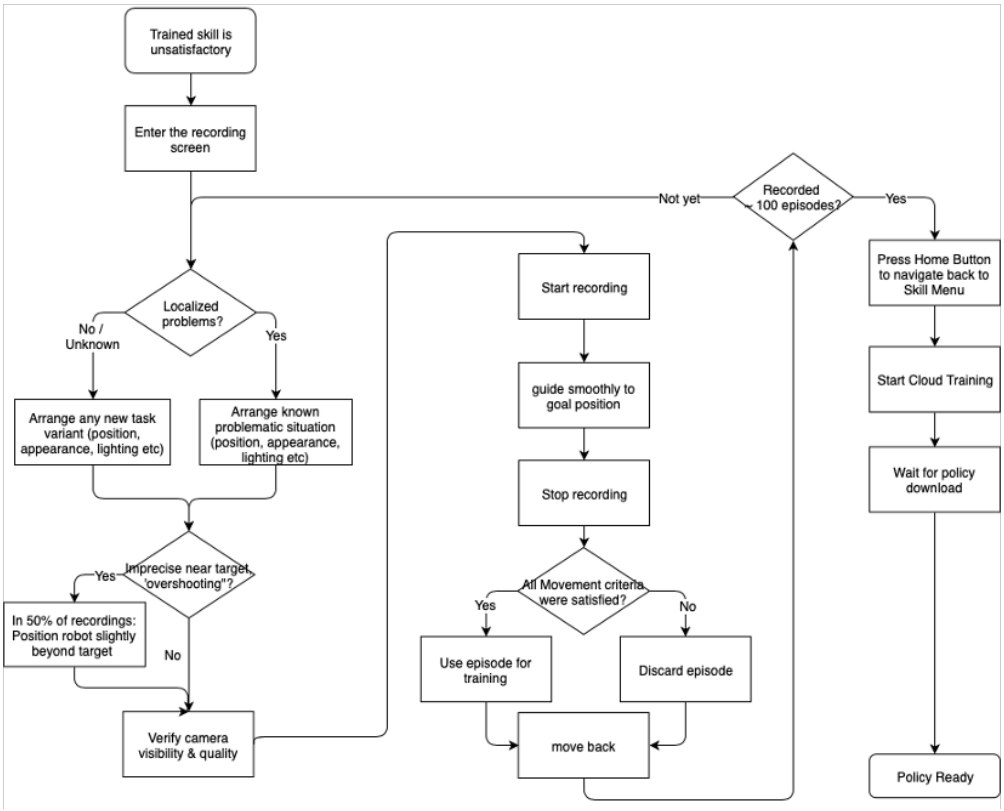


Figure 40: Additional training session flowchart

10.3 Troubleshooting Positioning Skills

Issues you may observe while testing the current skill version:

- **The robot avoids taking a direct path from some specific area to the target?**
  - This indicates that MIRAI has not been shown how to move along this specific pathway. Note that this rarely occurs with Positioning skills, as a direct path is generated automatically. It is only a problem if a large area is blocked or missing in the recordings.
    - **Fix:** Record new episodes that show MIRAI a path from the target to this specific area, by moving the robot along this pathway to the area; make sure to rotate all axes of rotation along this pathway as well.
- **The robot doesn't find its way to the target at all?**
  - This indicates that the current camera image is completely unknown to the skill, and so MIRAI does not know how to get to the target from area on display. There could be different reasons why:
    - **Fix:** Check your hardware first. Does the camera image appear normal? Could there be a reason that your camera image is blind? Is the camera mount or lens broken, is the light off?
    - If the camera image looks good...
    - **Fix:** Check your training plan next. Did you forget about a variance of your skill?
    - If the training plan seems thorough...
    - **Fix:** Record new episodes which show MIRAI this area, by moving the robot to this area from the target position.

- **The robot reaches a point near the target and then drifts away?**
  - There are two possible reasons for this to happen: The first option is that the camera is slightly offset.
    - **Fix:** Use the camera realign feature (for more information refer to chapter 14.2 Managing camera alignment and focus).
    - If the camera image is aligned...
  - This indicates that MIRAI does not have enough information about the areas close to the target and does not know how to recover from drifting near the target.
    - **Fix:** Record new episodes that show MIRAI these nearby areas, by moving the robot to these areas from the target position.
- **The robot does not want to rotate toward the target using certain axes of rotation/joints?**
  - This indicates that MIRAI has not been trained to utilize specific axes of rotation when approaching the target from specific areas.
    - **Fix:** First check that you have the required degrees of freedom for your skill enabled. If there are degrees of freedom missing, create a new skill with all the required degrees of freedom.
    - If all the degrees of freedom required for this skill are enabled...
    - **Fix:** Record new episodes that show MIRAI how to utilize all available axes of rotation, by moving the robot around the target while also rotating the tool through each available axis.
- **The skill places the robot tool off-target in a consistent way?**
  - If the skill is being performed on a new part/a part variation/or a part that looks different than others MIRAI was trained on, then this simply indicates that MIRAI needs to be shown the new part.
    - **Fix:** Record new episodes using this new part/part variation.
    - If this is a familiar part, then see the next possibility...
  - If the lighting has changed in your work environment, then this may indicate that MIRAI has not been trained under the current lighting conditions. Note that a change in room lighting should usually have a much greater effect than a small offset. However, a change in natural light due to a seasonal changes or different weather conditions can potentially be the reason.
    - **Fix:** Record new episodes under the current lighting conditions.
    - If the lighting in your work environment has not changed, then see the next possibility...
  - This may indicate that the aperture or focus has changed – e.g., the set screws for the aperture/focus have fallen off the lens.
    - **Fix:** Use the camera re-alignment feature to return the cameras to their original configuration (for more information refer to chapter 14.2 Managing camera alignment and focus). Setup Cameras if necessary. Then iteratively execute a skill with different focus settings.
    - If the camera alignment and the aperture/focus is correct...
  - This may indicate that MIRAI has been shown inconsistent target positions.
    - **Fix:** Check the target position for each episode by using the Target Frame Review feature (see Section 6.6 Check target pose consistency). If you find episodes with incorrect target positions, delete those episodes and create a new version of the skill with the remaining episodes. If you have further questions, or require a more complete review, reach out to the Micropsi Engineering team.

- **The skill was working, but now it no longer works?** E.g., the skill was good yesterday, but now the robot drives the tool to random positions or drifts with no aim.
  - This may indicate that the cameras have become mis-aligned, or that the aperture/focus has changed – e.g., the mounting screws have loosened, the mounts got damaged, the set screws for the aperture/focus have fallen off the lens.
    - **Fix:** Use the camera re-alignment feature to return the cameras to their original configuration (for more information refer to chapter 14.2 Managing camera alignment and focus). Setup Cameras if necessary.
    - If the camera alignment and the aperture/focus is correct...
  - This may indicate that the robots tooling has been damaged (specifically parts that are in the camera's FOV) – e.g., gripper fingers are damaged, or the fixturing is damaged.
    - **Fix:** Investigate your robot's tooling, as well as other relevant equipment in the workspace; replace damaged parts if necessary.
- **Unexpected movement direction while guiding:** During hand guiding or while recording an episode you may experience an issue where the robot moves against you or resists in the opposite direction.
  - This issue can happen if the force/torque sensor is not installed correctly.
    - **Fix:** Make sure that the sensor has been mounted correctly. Refer to the printed symbols along the outside of the sensor, (e.g. +Y, -X). These must line-up with the robot flange.

Note

On **FANUC robots**, the tool coordinate system is rotated by 90 degrees (see the MIRAI User Manual for FANUC Integration for more information).

10.4 Troubleshooting Motion Skills

Issues you may observe while testing the current skill version:

- **The robot avoids taking a direct path from some specific area to the target?**
  - This indicates that MIRAI has not been shown how to move along this specific pathway.
    - **Fix:** Record recovery episodes that show MIRAI a path from this specific area to the target or trajectory, depending on the task. Recovery episodes are recordings that subsequently show the skill how to recover from a bad position.
- **The robot doesn't find its way to the target at all?**
  - This indicates that the current camera image is completely unknown to the skill, and so MIRAI does not know how to get to the target from area on display. There could be different reasons why:
    - **Fix:** Check your hardware first. Does the camera image appear normal? Could there be a reason that your camera image is blind? Is the camera mount or lens broken, is the light off?
    - If the camera image looks good...
    - **Fix:** Check your training plan next. Did you forget about a variance of your skill?
    - If the training plan seems thorough...
    - **Fix:** Record new episodes which show MIRAI this area, by moving the robot from this area to the target position.

- **The robot reaches a point near the target and then drifts away?**
  - There are two possible reasons for this to happen: The first option is that the camera is slightly offset.
    - **Fix:** Use the camera realign feature (for more information refer to chapter 14.2 Managing camera alignment and focus).
    - If the camera image is aligned...
  - This indicates that MIRAI does not have enough information about the areas close to the target and does not know how to recover from drifting near the target.
    - **Fix:** Record new episodes that show MIRAI these nearby areas, by moving the robot from these areas to the target position.
- **The robot does not want to rotate toward the target using certain axes of rotation/joints?**
  - This indicates that MIRAI has not been trained to utilize specific axes of rotation when approaching the target from specific areas.
    - **Fix:** First check that you have the required degrees of freedom for your skill enabled. If there are degrees of freedom missing, create a new skill with all the required degrees of freedom.
    - If all the degrees of freedom required for this skill are enabled...
    - **Fix:** Record new episodes which involve rotations through each available axis. This will show MIRAI how to utilize all available axes of rotation.
- **The skill places the robot tool off-target in a consistent way?**
  - If the skill is being performed on a new part/a part variation/or a part that looks different than others MIRAI was trained on, then this simply indicates that MIRAI needs to be shown the new part.
    - **Fix:** Record new episodes using this new part/part variation.
    - If this is a familiar part, then see the next possibility...
  - If the lighting has changed in your work environment, then this may indicate that MIRAI has not been trained under the current lighting conditions. Note that a change in room lighting should usually have a much greater effect than a small offset. However, a change in natural light due to a seasonal changes or different weather conditions can potentially be the reason.
    - **Fix:** Record new episodes under the current lighting conditions.
    - If the lighting in your work environment has not changed, then see the next possibility...
  - This may indicate that the aperture or focus has changed – e.g., the set screws for the aperture/focus have fallen off the lens.
    - **Fix:** Use the camera re-alignment feature to return the cameras to their original configuration (for more information refer to chapter 14.2 Managing camera alignment and focus). Setup Cameras if necessary. Then iteratively execute a skill with different focus settings.
    - If the camera alignment and the aperture/focus is correct...
  - This may indicate that MIRAI was trained with bad data - for example, if inconsistent target positions or not enough recovery episodes near the target position were recorded.
    - **Fix:** Request a data review from the Micropsi Engineering team – if bad start frames are found in your skill version, they will inform you of what episodes to delete. Once you have deleted the bad episodes, create a new version of your skill with the good episodes.

- **The skill was working, but now it no longer works?** E.g., the skill was good yesterday, but now the robot drives the tool to random positions or drifts with no aim.
  - This may indicate that the cameras have become mis-aligned, or that the aperture/focus has changed – e.g., the mounting screws have loosened, the mounts got damaged, the set screws for the aperture/focus have fallen off the lens.
    - **Fix:** Use the camera re-alignment feature to return the cameras to their original configuration (for more information refer to chapter 14.2 Managing camera alignment and focus). Setup Cameras if necessary.
    - If the camera alignment and the aperture/focus is correct...
  - This may indicate that the robots tooling has been damaged (specifically parts that are in the camera's FOV) – e.g., gripper fingers are damaged, or the fixturing is damaged.
    - **Fix:** Investigate your robot's tooling, as well as other relevant equipment in the workspace; replace damaged parts if necessary.
- **Unexpected movement direction while guiding:** During hand guiding or while recording an episode you may experience an issue where the robot moves against you or resists in the opposite direction.
  - This issue can happen if the force/torque sensor is not installed correctly.
    - **Fix:** Make sure that the sensor has been mounted correctly. Refer to the printed symbols along the outside of the sensor, (e.g. +Y, -X). These must line-up with the robot flange.

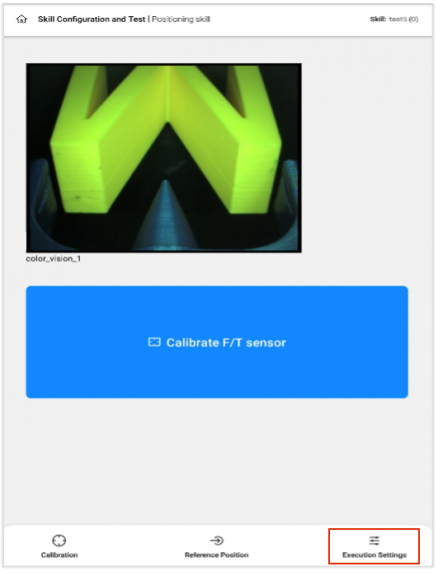
Note

On **FANUC robots**, the tool coordinate system is rotated by 90 degrees (see the MIRAI User Manual for FANUC Integration for more information).

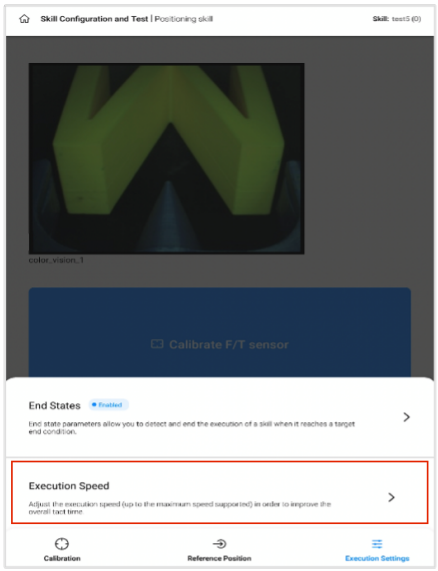
10.5 Adjusting skill speed

Adjust the execution speed (up to the maximum speed supported) in order to improve the overall tact time. This section allows you to set the standard speed for your skill. The advanced settings allow you to further fine-tune the speed profile.

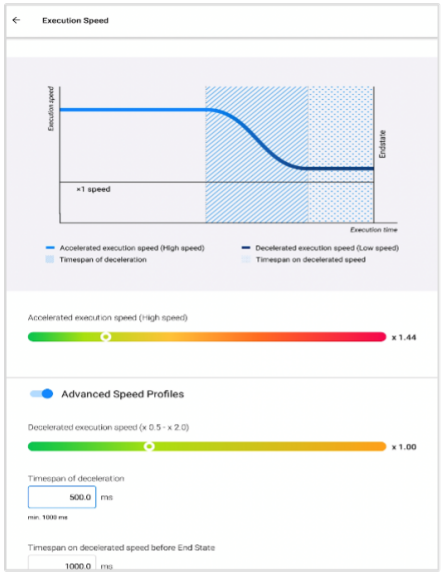
1. On the screen shown below, tap **Execution Settings**.



② Select "Execution speed."



③ The graph illustrates the acceleration speed. The default value is x1 (i.e., 1 times) of the speed at which the skill was recorded. Move the slider towards left or right to **change the execution speed**.



④ Select the **Advanced speed profiles** checkbox to **proceed to the advanced settings**.

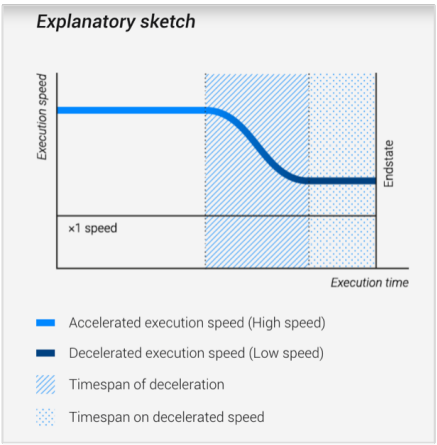
Note

The Advanced speed profiles feature is currently **not available for Positioning skills**.

5 The updated graph now illustrates the speed settings for the advanced speed profiling. You can now adjust the accelerated execution speed to a maximum of x5. Move the slider to adjust the decelerated execution speed.

Note

Moving to the **orange/red zone** means that your skill will be executed at a high speed. Use this cautiously, first experimenting with safer speed execution values.



6 Set a value for the **timespan of deceleration**. (Refer to the explanatory sketch shown at the top of the screen) Note that this value is in milliseconds.

7 Set a value for the **timespan on decelerated speed** before end state, in milliseconds.

8 Review your settings, and then tap **Apply Settings**. You can now proceed with (or resume) skill testing.

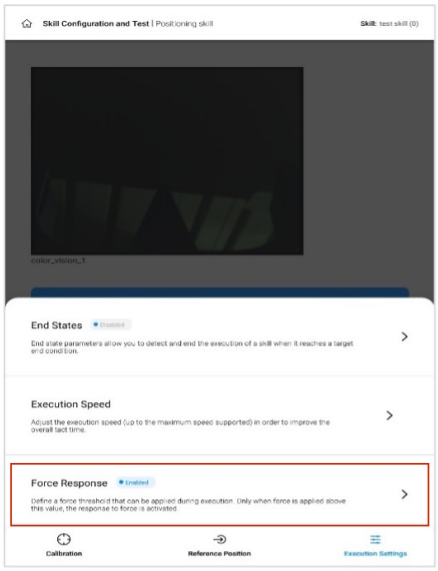
10.6 Force Response

Note

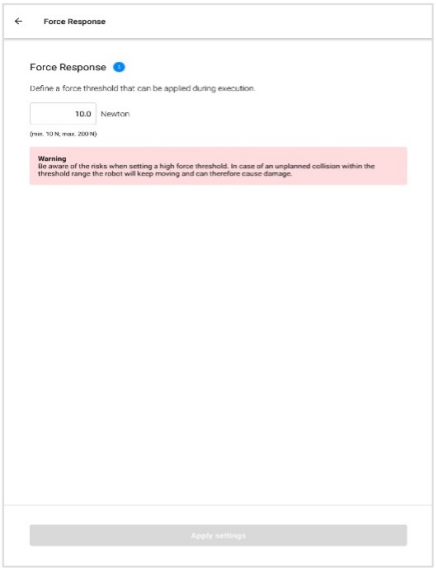
This feature **cannot be used** when operating without a force/torque sensor.

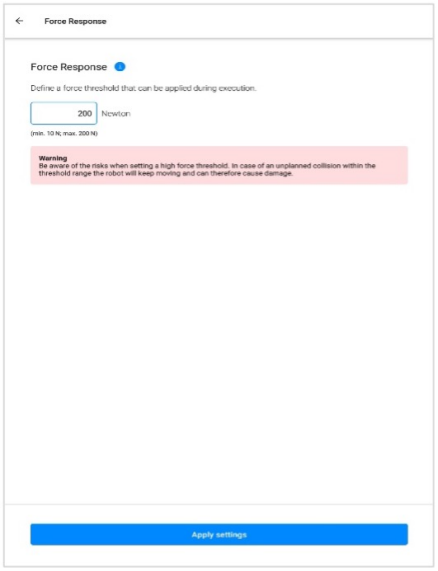
Force response helps define a force threshold that can be applied during skill execution. Only when force is applied above this value, the response to the force is activated.

1. Tap **Execution Settings** in the bottom bar and then select **Force Response**.



2. Enter the value of the **force threshold** (in Newton) that can be applied during execution, in the respective text box. The force threshold value must be set between 1 N and 99,999 N. Only forces exceeding this value will trigger a response.





Note

During skill execution, the force/torque sensor reacts to potential collisions. For applications requiring force, the force response will only activate when the applied force exceeds the force response value.

10.7 Implementing MIRAI Skills into the Robot Program

Please see the MIRAI Integration Guide for your preferred robot platform to implement and execute MIRAI skills in the robot program in the native controller.

- **Universal Robots:** To implement MIRAI Calls into the UR Polyscope program applicable, see “Using MIRAI Skills with UR Polyscope” in the MIRAI User Manual for UR Integration.
- **FANUC:** To add and synchronize trained MIRAI Skills on the FANUC controller, see “Using MIRAI Skills with FANUC Teach Pendant” in the MIRAI User Manual for FANUC Integration.
- **KUKA:** To add a MIRAI Skill in your KUKA program via the KUKA smartPAD, see “Using MIRAI Skills with KUKA” in the MIRAI User Manual for KUKA Integration.

11 End State Triggers

The end state is the condition which helps MIRAI understand that a skill has ended, and it must hand the control of the robot back to the native controller. End state triggers must be set to detect and end the execution of a skill.

11.1 End State Triggers Used by MIRAI

MIRAI uses the following triggers for end state detection:

End State Name	Description	Best Use Cases
Visual End State Recognition (Smart Detection)	This smart feature allows MIRAI to automatically determine when the skill should end by comparing the current live camera feed with the scenes or images recorded during training.	• most cases (and therefore, can be enabled and tried first) • when it is unclear which end state is the best • when the visuals are clearly defined within the camera's field of view
Proximity-based Available from version 16.0.0	When the robot gets close to its target, this informs MIRAI that the end state has been successfully reached based on the set threshold values (distance and orientation). *Notes:* The skill will only stop executing when all the specific criteria are met. This end state can only be used for positioning skills.	• most cases (and therefore, can be enabled and tried first) • when it is unclear which end state is the best • when the visuals are clearly defined within the camera's field of view
TCP Force-based	When the force measured at the TCP exceeds (above or below) a certain force threshold, this tells MIRAI that the end state is successfully reached. *Note:* This end state is not available when working without a force/torque sensor.	*For tasks which end by:* • touching a surface (e.g., end points for surface measurements or insertion) • stopping a surface interaction (e.g., by reaching the end of a working area) • dropping a part (e.g., after picking)
TCP Speed-based	When the movement at the TCP exceeds (above or below) a certain speed threshold, this tells MIRAI that the end state is successfully reached.	*For tasks which end by:* • slowing down or stopping when reaching a target position in space • stopping after a successful insertion • slowing down when reaching the end path (e.g., gluing, painting, etc.)
TCP Position-based (Beta)	MIRAI recognizes the desired end state when the end effector crosses a defined plane. *Note:* This feature is released as a beta version (accuracy is at ±5mm on low speed setting (≤1)).	
Anomaly-based Available from version 16.0.0	An anomaly-based end state halts skill execution if the current image looks unfamiliar (based on a threshold you set). Using this end state can increase the reliability of your skill, by detecting conditions which were not encountered during training (e.g., changes in the lighting of an environment, tooling setup, materials used in production, etc.).	We recommend testing and iterating different thresholds to find what works best for your skill. Once you find a good threshold, it would be best to enable this end state in all cases.

Additional Important Notes:

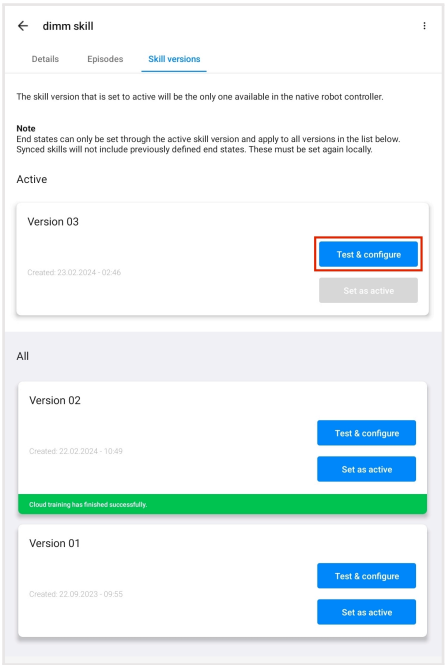
- **All end state options can be used in parallel.** In case these triggers are used simultaneously, the condition that is met first will trigger the desired end state.
- **End states need to be configured on every MIRAI Setup.** This is especially important when syncing a skill created on another controller, since the end states are not shared (see chapter 13 Shared Skills - Using a Skill on Different Setups for more details about this feature).
- **How to use proximity- and anomaly-based end states for "old" skill versions.** All "old" skill versions, (i.e., those created on a MIRAI controller(s) with a software version lower than 16.0.0) do not have the proximity-based and anomaly-based end states available. **If you would like to utilize these end states on a skill created via a MIRAI controller with an older version, you must:**

  - Update your controller's software version to at least 16.0.0.
  - Start a new cloud training to create a new skill version.
  - Configure the new end states you would like to use.
- **Implications when switching between "old" and "new" skill versions.** After updating your MIRAI controller to version 16.0.0 or higher, the following should be considered:
  - "Old" skill versions will still not have proximity- and anomaly-based end states.
  - If you create "new" skill version (trained with MIRAI 16.0.0 or higher) and set it as active, then after doing so, decide to activate an "old" skill version again and change its end state settings, the proximity- and anomaly-based end state on the "new" skill version will be deactivated.

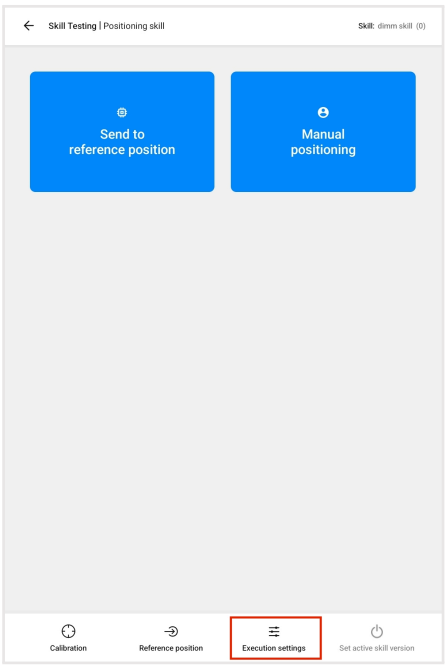
11.2 Configuring End States on MIRAI

Based on the information provided above, review your task at hand and determine which condition/s would best indicate when the final state of a successful skill execution is reached. To put in place a robust and reliable end state recognition for different tasks and use cases, triggers for the end states you choose must then be set and tested, and subsequently, adjusted if necessary.

1. **Select the skill** for which you want to set end states.
2. Once you land on the respective skill's "Details" screen, select the **Skill version** tab.
3. Ensure the skill version you want to work on is set as active, and then proceed by tapping **Test & configure**.



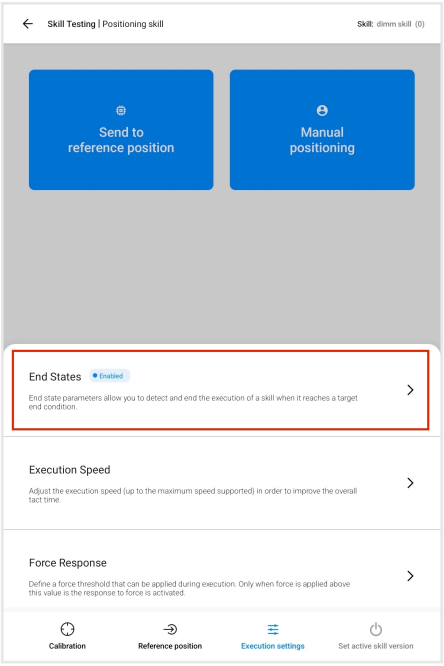
4. When you land on the "Skill Testing" screen, tap **Execution Settings**.



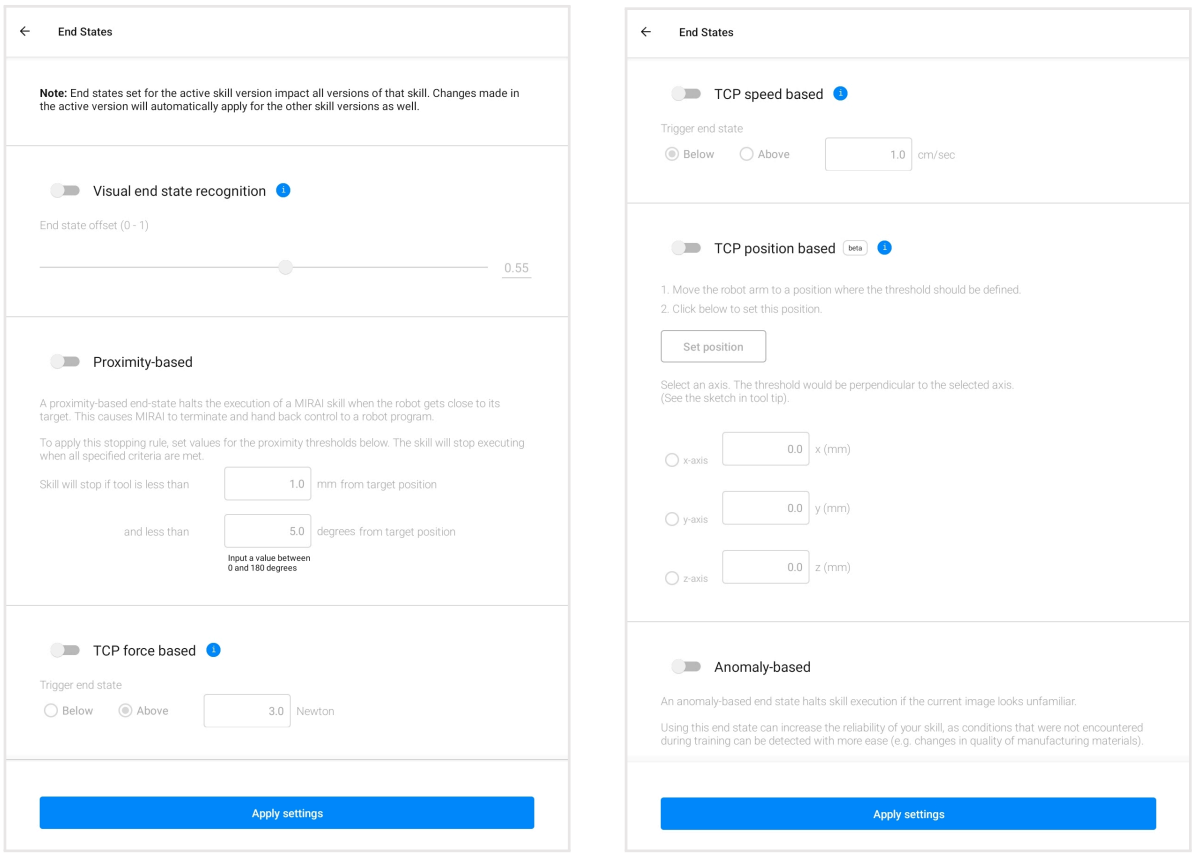
5 From the "Settings" screen, select **End States**.

Note

These options are not mutually exclusive. The other settings will be further discussed in section 10.5 Adjusting skill speed and section 10.6 Force Response.

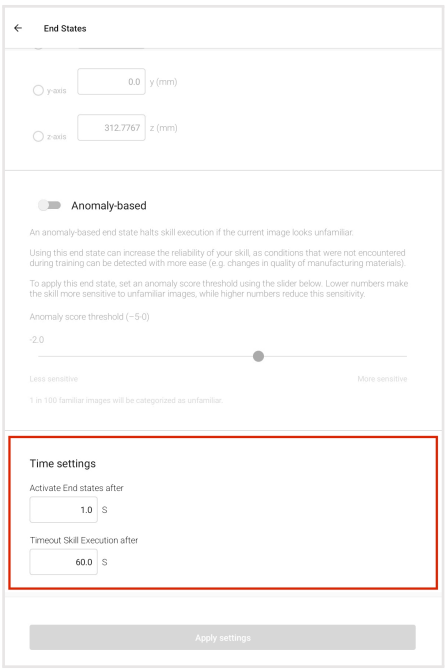


6 **Enable the end states** you want to utilize, then configure them accordingly by following the instructions found in the respective subsections that follow (subsections 11.2.1 to 11.2.6).



7 At the bottom of the screen (after the last end state option, i.e., Anomaly-based for Version 16.0 or higher or TCP position-based for lower versions), you will see Time settings:

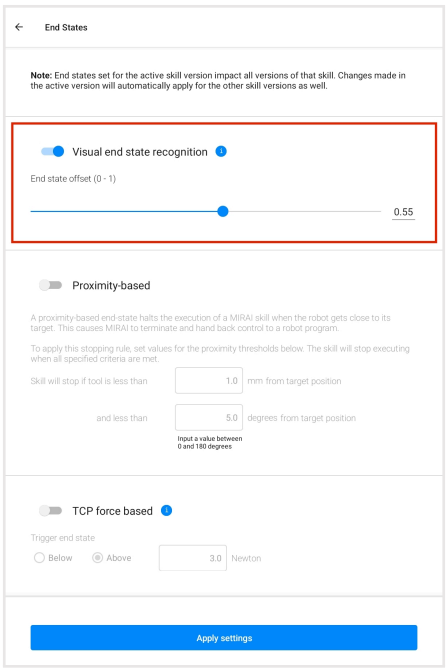
- Under **Activate End states after**, enter a value in the field that defines the **delay in seconds**, after the start of skill execution, when the end state settings should be activated. This is done to prevent a wrong early end state detection that can happen while the robot arm is set in motion and is accelerating. The value is preset to 1 second, which should suffice for most cases.
- Under **Time Skill Execution after**, enter a value in the field that defines the **maximum timespan (in seconds)** in which a skill should be executed. This is meant to serve as a local timeout for a skill in case it does not reach a defined end state. After the timeout period, the skill will be automatically stopped by the system.



⑧ Once you have enabled and configured your selected end states, make sure to confirm them by tapping **Apply Settings** at the bottom of the screen; otherwise, the changes will be lost.

11.2.1 Visual End State Recognition

When the visual-based end state is enabled, you must **set a threshold value**. You can do so by using the slider or tapping the numerical value field on the right-hand side and manually typing your desired value. A higher value means the positioning at the end of the skill must match the images from the live camera feed and should be recorded as precisely as possible, while a lower value means the skill could stop significantly before the end state (e.g., A 0.55 value indicates that when MIRAI has reached 55% of the desired final position, the skill is then terminated).



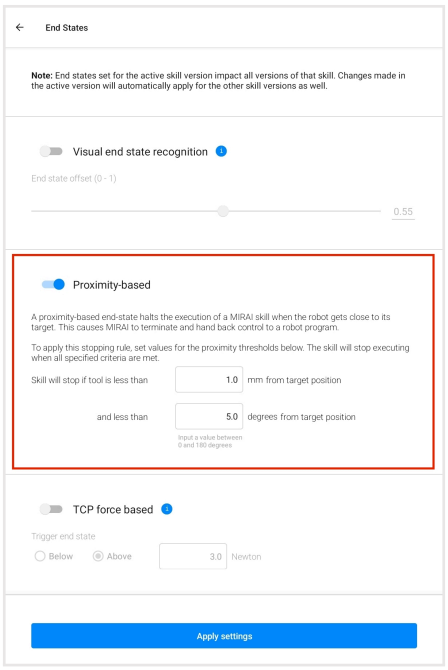
11.2.2 Proximity-based end state

available from version 16.0.0

When utilizing the Proximity-based end state, you must **set values for the proximity thresholds (both in mm and degrees)**. In the boxes, enter the distance value from the target position based on where you want the skill execution to stop. For example, if you want the skill to stop when the tool is less than 1.0 mm and less than 5.0 degrees away from the target position, enter these values.

Note

The skill will only stop executing when **all** the specific criteria are met. This end state can only be used for positioning skills.

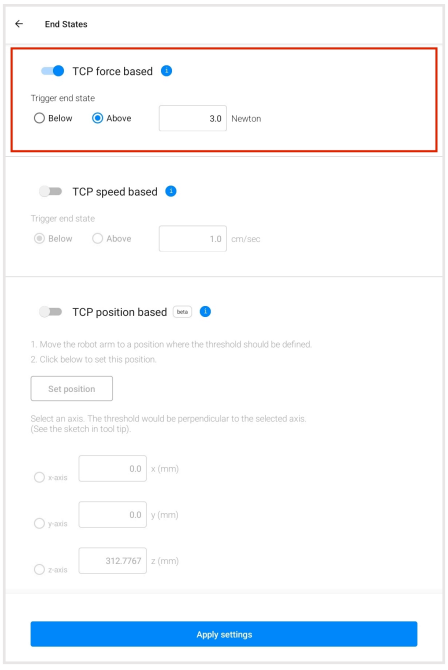


11.2.3 Force-based end state

When enabling the TCP force-based end state, you must define whether the skill should be stopped **above or below the force value** indicated, as measured at the force/torque sensor. The default force value is set to 3 Newtons; nonetheless, you can **enter a value between 0 to 100 Newtons**.

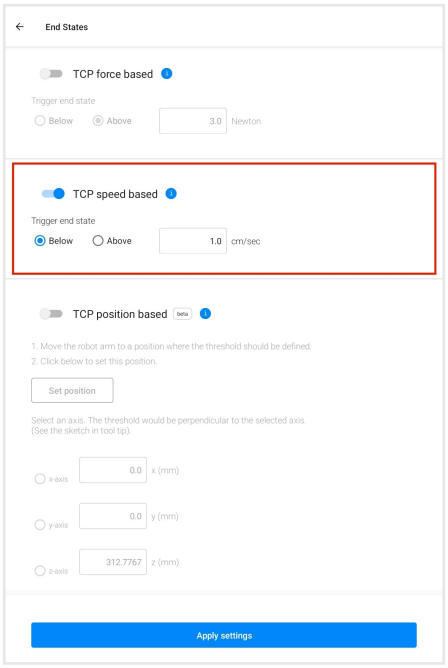
Note

This end state trigger is **not available** when working without a force/torque sensor.



11.2.4 Speed-based end state

When using the TCP speed-based end state, you must enter a **speed-based threshold value from 1 to 1000 (mm/s)**, then select whether the skill should be stopped **below or above the speed value** you entered.

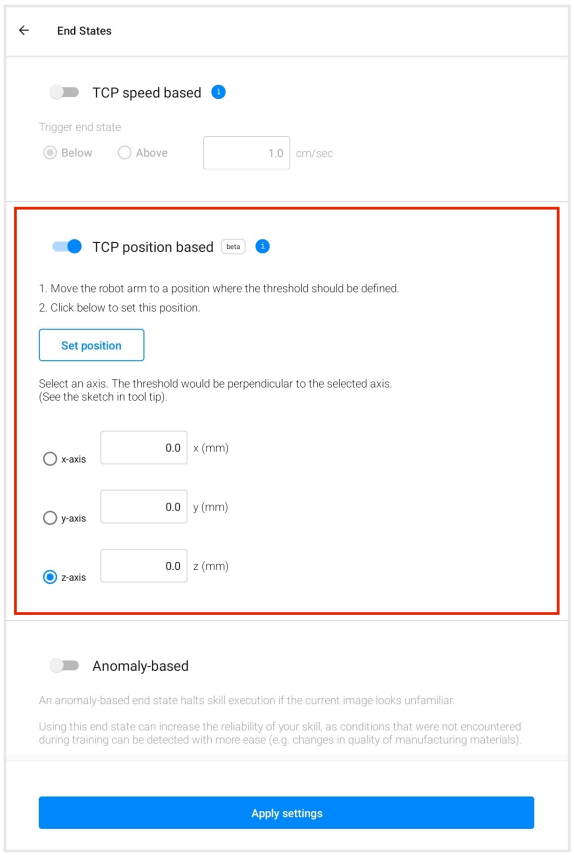


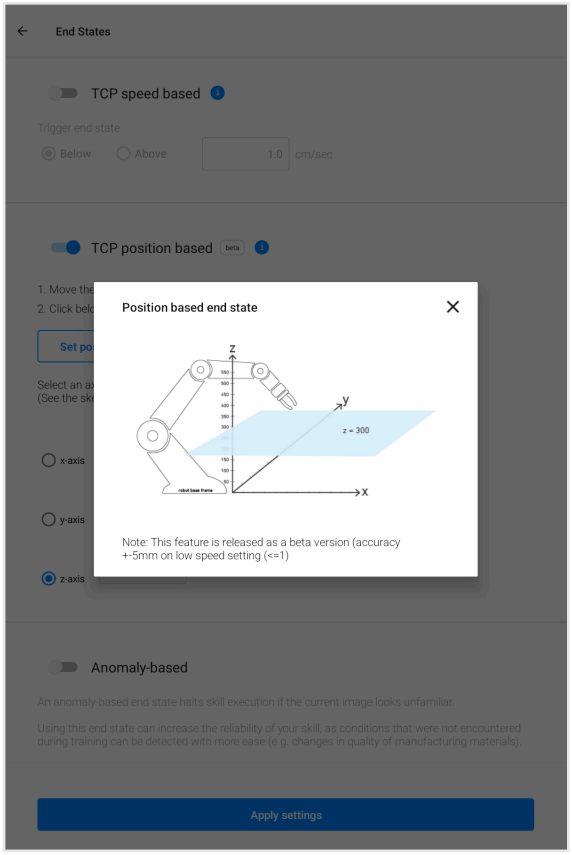
11.2.5 Position-based

To set a TCP position-based end state, move the robot arm to **a position where the threshold should be defined**, then tap **Set position** in the app to set this position — the X, Y and Z value for this position will automatically update. Afterwards, **select an axis (X, Y, or Z)**, and the threshold would be perpendicular to the selected axis.

Note

This feature is currently released as a beta version (accuracy is at ±5mm on low speed setting (≤1)).

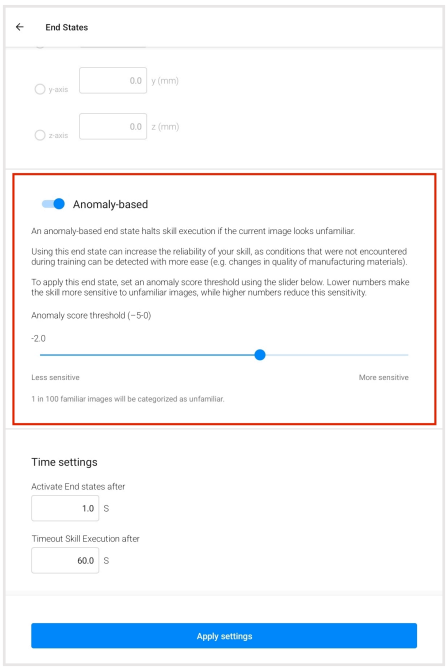




11.2.6 Anomaly-based

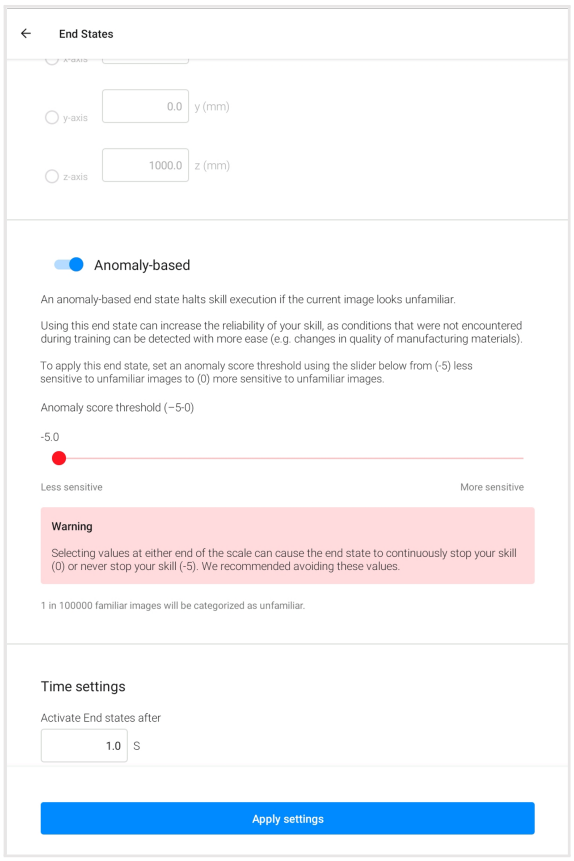
available from version 16.0.0

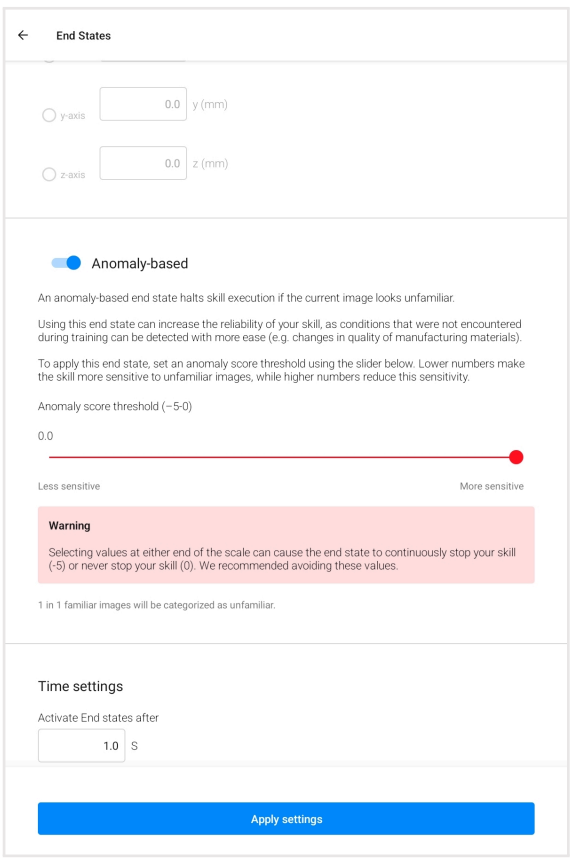
When applying the anomaly-based end state, you need to set an **anomaly score threshold (which ranges from -5.0 to 0.0)** by using the slider on the screen. The default value is set at -2.0, where 1 in 100 familiar images will be categorized as unfamiliar. Choosing a lower number will make the skill less sensitive to unfamiliar images, while selecting a higher number will increase the sensitivity.



Note

Selecting values at either end of the scale can cause the end state to never stop your skill (i.e., selecting -5, where 1 in 100,000 images will be categorized as unfamiliar), or continuously stop your skill (i.e., selecting 0, where 1 in 1 familiar image(s) will be categorized as unfamiliar). That said, it would be best to **avoid choosing these extreme values**.





11.3 Returning Values for MIRAI End States on the Robot's Native Controller

To determine how you can get the return values of the MIRAI functions called during skill execution on your robot's native controller (which will allow you to understand which end state caused the skill to stop), kindly check the following resources:

- *Universal Robots:* MIRAI User Manual for UR Integration, Polyscope Functions and Variables
- *FANUC:* MIRAI User Manual for FANUC Integration, Returning Values of MIRAI Functions
- *KUKA:* MIRAI User Manual for KUKA Integration, Returning Values of MIRAI Functions

12 Operations monitoring using MQTT, OPC UA, and/or FTP

Operations monitoring is the continuous collection and analysis of system performance data during production. This data provides insights into how MIRAI operates in real time, allowing you to:

- Establish performance benchmarks
- Identify opportunities for process optimization
- Troubleshoot issues efficiently

MIRAI transmits two primary types of monitoring data:

1. **MIRAI status codes:** Boot sequence, operational states, and error codes (see 12.1).
2. **Skill execution data:** Details on different aspects of skill execution (see 12.2).

Communication protocols

MIRAI supports three protocols for transmitting operations data:

Protocol	Description	Best for
MQTT	Lightweight messaging protocol for IoT applications.	Real-time monitoring with minimal overhead.
OPC UA	Machine-to-machine protocol for industrial automation.	Integration with industrial control systems.
FTP	Protocol for transferring files between a client and server.	Batch transfers of larger datasets.

Implementation features

MIRAI provides flexible options for collecting and transmitting skill execution data:

- All monitoring features are optional and do not affect MIRAI's core functionality.
- Each protocol can be configured individually.
- Data transmission can be turned on or off at any time.
- Both unauthenticated and login-based connections are available (*for MQTT and FTP only*).

Protocol capabilities

The table below summarizes the data types available through each protocol. Refer to the subsection for each protocol for further information and configuration instructions.

Protocol	Data	Format
MQTT	• MIRAI status • Metadata, execution data	• Numerical value • JSON string and plain strings
OPC UA	• MIRAI status	• Numerical value
FTP	• First and last skill execution images • Metadata, execution data	• PNG file • CSV file

PERFORMANCE NOTE

A stable connection to the configured server is required for optimal performance. If the server becomes unavailable, the system will attempt to reconnect, which may delay execution or interrupt reporting without notice.

12.1 MIRAI status codes

MIRAI status codes, transmitted as numerical values, indicate the system's current operational state in real time. Two protocols transmit MIRAI status: MQTT and OPC UA.

Status information is reported:

- During system initialization (boot process)
- When the system becomes fully operational
- When errors arise

During system initialization, sequential status codes track the system's progress through the boot process. These numerical codes signal each stage of initialization, as detailed in the table below.

Status code	String	Status description: Boot sequence
10	STATUS_BOOT_HELLOWORLD	Linux boot complete
11	STATUS_BOOT_DEVICES	Establishing external device connections
12	STATUS_BOOT_INFERENCE	Setting up inference engine
13	STATUS_BOOT_VPN	Setting up remote support VPN **NOTE:** *VPN Activation requires separate manual authorization.*
14	STATUS_BOOT_RUNNER	Setting up runner
15	STATUS_BOOT_APIS	Setting up APIs
16	STATUS_BOOT_DB	Setting up database connections
17	STATUS_BOOT_NETWORK	Loading network configuration
18	STATUS_BOOT_ROBOT_COMM	Loading robot communication interfaces
19	STATUS_BOOT_MULTIPROCESS	Setting up multiprocessing
20	STATUS_BOOT_CLOUD	Setting up cloud connectivity
21	STATUS_BOOT_API_OBJECTS	Setting up API objects
22	STATUS_BOOT_SYNCING	Setting up syncing and polling (from/to cloud)
30	STATUS_BOOT_READY	Runtime ready (will immediately transition to 255)

When operational, MIRAI transmits one of three possible states, as detailed in the table below. During normal operation in production, status alternates between 254 and 255.

NOTE: Classification status codes are unavailable. When a classifier is running, MIRAI reports status 255.

Status code	String	Status description: Operational states
253	STATUS_TRAINING	Skill training and testing, not ready for other requests
254	STATUS_EXECUTING	Skill execution, not ready for other requests
255	STATUS_NORMAL	Normal, ready for requests

Error codes are transmitted for the following issues:

NOTE: The error code persists until it is overridden by the next successful execution; it does not first reset to normal status (255).

Status code	String	Status description: Error codes
100	STATUS_ERROR_GENERAL	Cannot execute: unspecified error
110	STATUS_ERROR_CAMERA_COMM	Camera unreachable
111	STATUS_ERROR_CAMERA_TEMPERATURE	Camera temperature too high
120	STATUS_ERROR_ROBOT_COMM	Robot unreachable
130	STATUS_ERROR_PERF_COMPUTE	Insufficient performance

Status code 1 is a reserved, non-meaningful value used to test connections (*currently OPC UA only*).

Status code	String	Status description: Test signal
1	STATUS_CHANNEL_TEST	Channel test – do not respond

12.2 Skill execution data

Skill execution data captures details about each instance of skill execution during production. This information facilitates historical trend analysis and supports troubleshooting efforts.

Two protocols transmit skill execution data: MQTT and FTP.

Skill execution data is divided into two categories:

- **Metadata:** Contextual data about system environment and message creation (*MQTT only*).
- **Execution details:** Concrete values for defined metrics such as skill duration and skill execution result.

The corresponding data fields for each category are detailed in the two tables below.

METADATA (MQTT only)		
Data field	**Description**	**Notes**
system	The system generating the message.	Always “MIRAI”.
version	MIRAI version and Training App version.	*Format:* ”23.0.0 - v23.0.0” *First entry:* MIRAI software *Second entry:* MIRAI Training App
client	The client identifier.	
message_type	The type of message.	Always “execution_data”.
message_type_version	The version of the message type.	Currently always 1, will be incremented when the message format is changed.
message_created	The message timestamp.	*Format:* YYYY-MM-DD HH:MM:SS.µs in the controller’s local timezone (default: CET/Berlin).
message_created_utc	The message timestamp in UTC.	*Format:* YYYY-MM-DD HH:MM:SS.µs.

EXECUTION DATA		
Data field	**Description**	**Notes**
skill_id	The identifier of the skill.	Currently empty, but will contain the skill ID in a future version (without an increment of message_type_version).
skill_name	The name of the skill.	
tracing_token	A unique identifier optionally set via an RPC call to `/xmlrpc/set_tracing_token` to track individual products through the manufacturing process.	Ask your Micropsi Industries engineering contact for details.
execution_result	The end state stopping the skill execution.	*Categories:* speed, force, visual, anomaly score, distance, timeout. Refer to 11 End State Triggers.
true_fps	The true frames per second during skill execution.	
start	The start time of skill execution.	*Format:* YYYY-MM-DD HH:MM:SS.µs.
stop	The stop time of skill execution.	*Format:* YYYY-MM-DD HH:MM:SS.µs.
duration	Duration of skill execution.	In seconds.
start_to_motion latency	Time between skill execution command and MIRAI being ready to move the robot.	In seconds. Use for relative comparison across executions.
speed	The speed when the skill was stopped.	In millimeters per second. See 11.2.4 for details.
force	The force applied when the skill was stopped.	In newtons. See 11.2.3 for details.
done_probability	Same number reported in end state detection for visual end states.	**Deprecated – do not use.**
anomaly_score	A measure of how similar the last frame of the skill execution looks compared to recorded episodes.	Values are skill-specific (see 11.2.6). Increasing values over time may signal inaccuracy. Monitor trends to maintain skill performance.
distance_mm	Estimated 3-D distance from the target when the skill was stopped.	In millimeters. See 11.2.2 for details.
distance_deg	Estimated shortest rotational distance from the target when the skill was stopped.	Expressed as an angle between 0° and 180°. See 11.2.2 for details.

NOTE: If **distance_mm** or **distance_deg** shows "9223372036854775807" (the highest possible integer in Python), it indicates that a timeout occurred before the skill could estimate the distance to target, or that the skill does not support distance estimates because it was trained on a MIRAI version prior to 17.0.0.

12.3 Status and execution results using MQTT

This feature transmits MIRAI status and execution data each time a robot program triggers a skill execution, collecting data for historians and dashboards. If multiple MIRAI skills are executed in a single robot program, data for each skill will be transmitted.

Key capabilities:

- MIRAI status sent as a single numerical value
- Execution data sent as summaries and/or individual data fields
- Integration with industrial historian systems
- All MQTT protocol versions supported (5.0, 3.1.1, and 3.1)

12.3.1 MQTT overview

MQTT is a lightweight data exchange protocol for IoT messaging. It uses a publish/subscribe model that decouples message senders (publishers) from message receivers (subscribers).

All devices (**clients**) communicate through a central **message broker**. The broker filters incoming messages from publishers and distributes them to subscribers based on **topics**, which are hierarchically organized keywords. Publishers send messages to specific topics, and subscribers receive messages by subscribing to those topics.

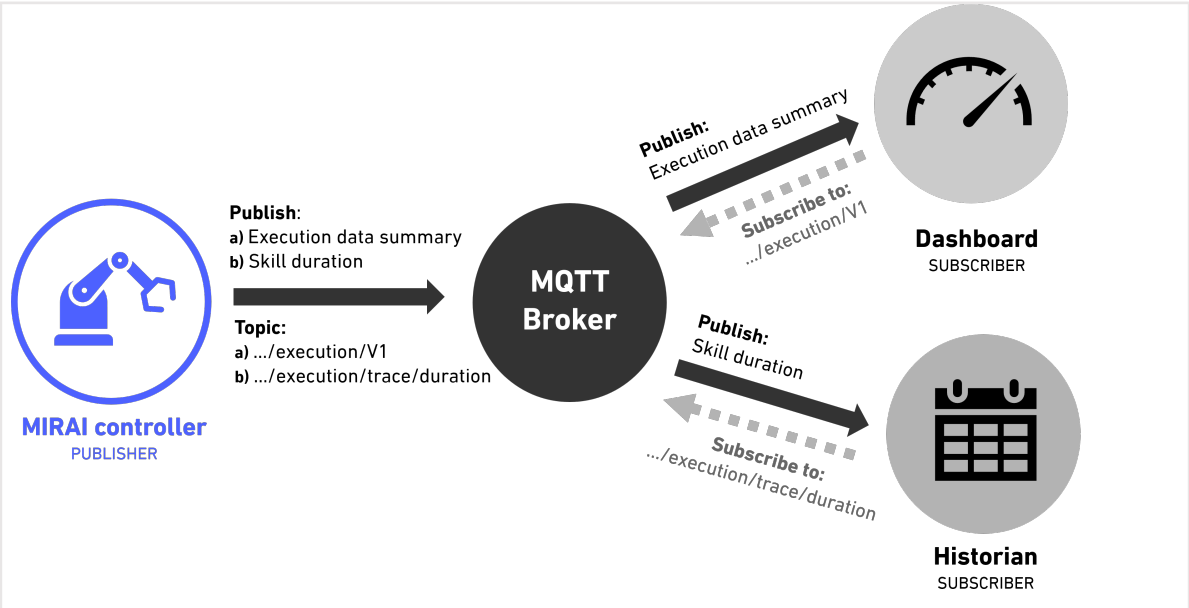
MQTT client: A device that uses MQTT.

- Clients that send messages are *publishers*.
- Clients that receive messages are *subscribers*.

MQTT broker: System coordinating messages between clients.

MQTT topic: Hierarchical string identifier for message routing, delimited by slashes (/).

- *Base topic:* Top-level identifier for organizing MQTT messages.
- *Subtopics*: Secondary path component that categorizes messages within a base topic.



12.3.2 Using MQTT subtopics to publish different data

MIRAI can publish three different types of data using MQTT. Select the information to receive by subscribing to specific subtopics. Subtopics are shown below with an ellipsis representing the user-defined base topic.

- **MIRAI status:** Subscribe to subtopic `.../status` to receive MIRAI's operational state as a numeric code.
- **Execution data summaries:** Subscribe to subtopic `.../execution/V1` to receive all data fields together in a single JSON string (see the example below).
  TIP: *Use this option for comprehensive reporting across multiple data fields.*
- **Specific data fields:** Subscribe to subtopic `.../execution/trace/<data field>` to receive individual data fields as plain strings with numerical execution counters. For example, use `.../execution/trace/speed` to publish only speed values.
  TIP: *Use this option to generate sequential data for trend analysis.*

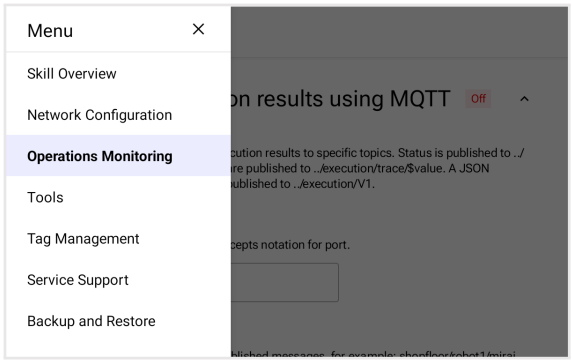
Here is an example of a JSON string that gets sent to the subtopic `.../execution/V1` under the base topic `MIRAI`. For descriptions of the individual data fields, refer to 12.2 Skill execution data. **NOTE:** Empty quotation marks indicate missing values, as seen for skill_ID in the example below.

```
MIRAI/execution/V1
{
  "metadata": {
    "system": "MIRAI",
    "version": "23.0.0 - v23.0.0",
    "client": "MIRAI-678",
    "message_type": "execution_data",
    "message_type_version": 1,
    "message_created": "2025-01-13 10:26:44.803065",
    "message_created_utc": "2025-01-13 09:26:44.803065"
  },
  "execution_data": {
    "skill_id": "",
    "skill_name": "Grip_cube",
    "tracing_token": "",
    "execution_result": "distance",
    "true_fps": 12.44778280079366,
    "start": "2025-01-13 10:26:43.266078",
    "stop": "2025-01-13 10:26:44.792455",
    "duration": 1.526377,
    "start_to_motion_latency": 0.32361912727355957,
    "speed": 0.011012558158122087,
    "force": 0.599620009337837,
    "done_probability": 0.016796132549643517,
    "anomaly_score": -0.1453857421875,
    "distance_mm": 2.3260265588760372,
    "distance_deg": 0.00232602655887660376
  }
}
```

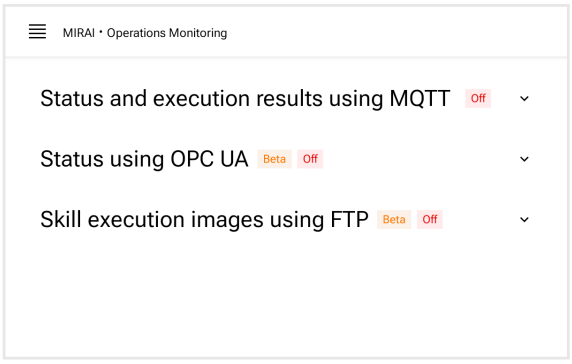
12.3.3 Set up and activate MQTT publishing

Follow these steps to configure MQTT in the MIRAI Training App:

1. **Access the MQTT screen:**
   a. Go to the main menu and tap **Operations Monitoring**.
   b. Use the drop-down arrow to expand **Status and execution results using MQTT**.



In the main menu, tap **Operations Monitoring**.



Choose an option to configure.

2. **Configure MQTT settings:**
   a. In the **Broker** field, enter the name or IP address of the broker.
   b. In the **Topic** field, enter the base topic for the messages MIRAI will publish.

      ❗ **IMPORTANT:** Topic names are case-sensitive and must match exactly between the MIRAI Training App and broker. Topic mismatches cause silent publishing failures.

   c. If needed, enter a **username** and **password**. The password is hidden using asterisks.
      - *If your server requires authentication,* leaving the fields blank or entering incorrect inputs will trigger an error pop-up: "Something went wrong. Server Response: Not authorized." Tap **OK**, then review and adjust your settings to proceed.
      - *If no authentication is required,* these fields can remain blank.
   d. OPTIONAL: Use the **Mask** parameter to omit plain-string publishing of data fields:
      - **To exclude all individual data fields** and publish only the JSON summary, enter: `execution/trace`
      - **To exclude a single data field** – for example, "skill name" – enter that data field as follows: `execution/trace/skill_name`
      - **To exclude multiple data fields**, enter each one in a comma-separated list like this: `execution/trace/skill_name,execution/trace/skill_id`

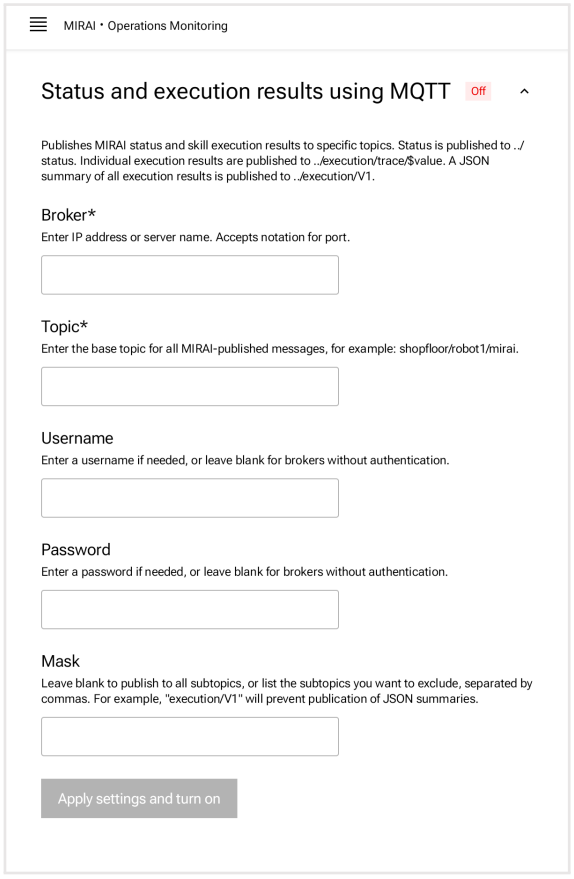
3. **Turn on publishing:**
   a. Tap **Apply settings and turn on**.
      NOTE: Tapping **Apply settings** or **Save settings and turn on reports** will test the broker address. If the broker is offline or the address is incorrect, an error message will appear. Tap **OK**, then review and adjust your settings to proceed.

b. *After activation*, check that the following appear (as shown in the screenshot below):

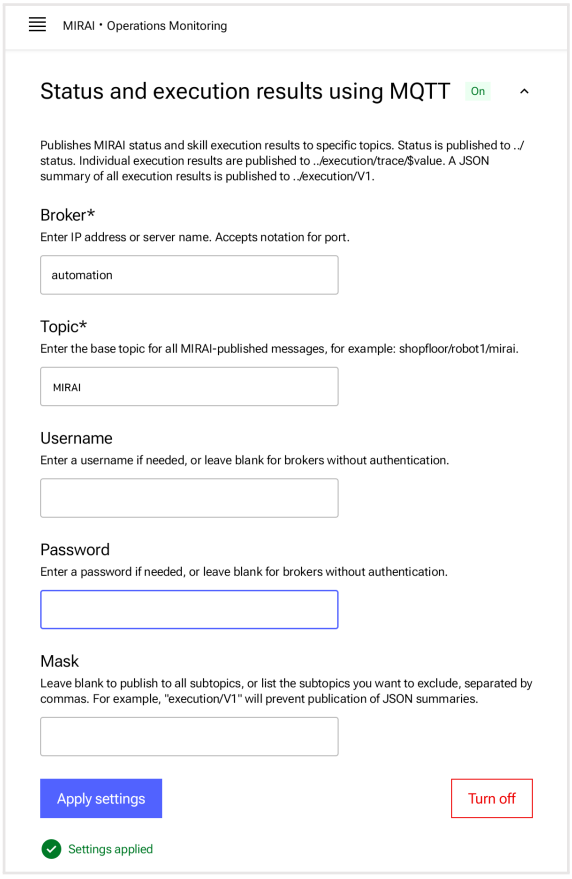
- **Settings applied** will appear under the activation button.
- An **On** indicator will appear next to the MQTT heading.
- A **Turn off reports** button will appear at the bottom of the screen.

IMPORTANT: If you modify the settings, you must tap **Apply settings** for the changes to take effect.

To turn this protocol off, see 12.6 Turning communication protocols off or on.



Fields are empty, communication is **Off**.



Settings applied, communication is **On**.

12.4 Status using OPC UA

Status using OPC UA is currently in beta.

This feature transmits MIRAI status as numerical codes to a PLC, enabling real-time monitoring of the system's operational state. During initialization, these codes indicate each step of the boot sequence. After MIRAI is operational, the codes reflect one of three possible system states (see 12.1 MIRAI status codes).

Key capabilities:

- Real-time transmission of MIRAI status codes
- Sequential boot process monitoring
- Integration with PLC and HMI systems

Use this feature to monitor system performance and quickly identify errors during startup and diagnose operational issues.

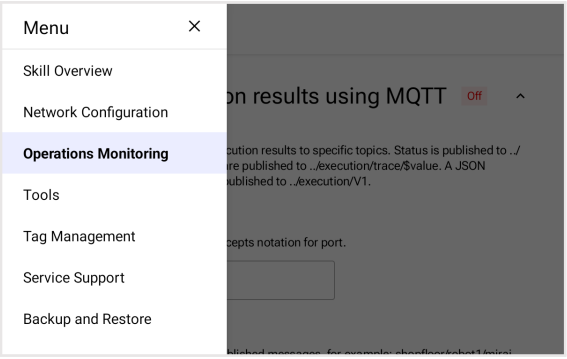
PERFORMANCE NOTE

If the server connection is lost, MIRAI will attempt to reconnect, which may delay skill execution.

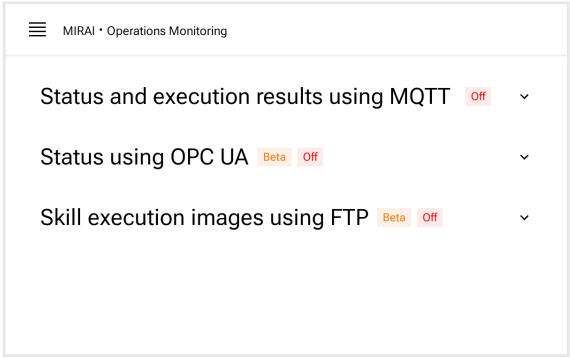
12.4.1 Set up and activate OPC UA communication

Follow these steps to configure OPC UA in the MIRAI Training App:

1. **Access the OPC UA screen:**
   - **a**. Go to the main menu and tap **Operations Monitoring**.
   - **b**. Use the drop-down arrow to expand **Status using OPC UA**.



In the main menu, tap **Operations Monitoring**.

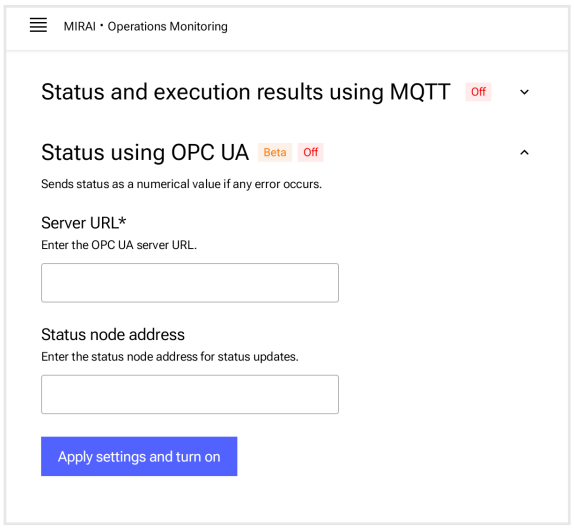


Choose an option to configure.

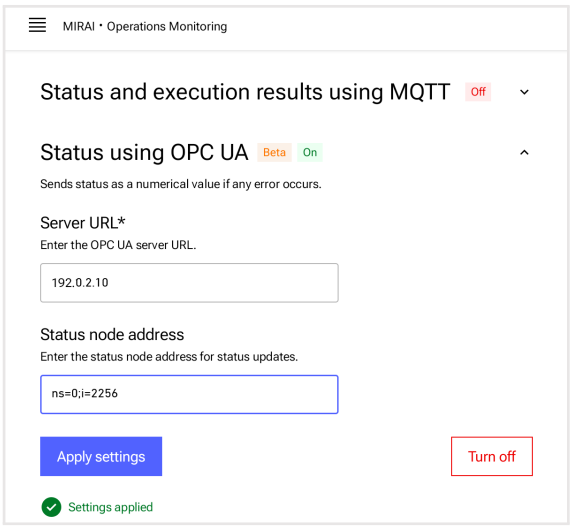
2. **Configure OPC UA settings:**
   - **a**. In the **Server URL** field, enter the OPC UA server URL.
   - **b**. In the **Status node address** field, enter the status node for status notices.
3. **Turn on OPC UA:**
   - **a**. Tap **Apply settings and turn on**.
4. *After activation*, check that the following appear (as shown in the screenshot below):
   - **Settings applied** will appear under the activation button.
   - An **On** indicator will appear next to the OPC UA heading.
   - A **Turn off reports** button will appear at the bottom of the screen.

IMPORTANT: If you modify the settings, you must tap **Apply settings** for the changes to take effect.

To turn this protocol off, see 12.6 Turning communication protocols off or on.



Fields are empty, communication is **Off**.



Settings applied, communication is **On**.

12.5 Skill execution images using FTP

Skill execution images using FTP is currently in beta.

This feature transmits the first and last frames of each skill execution as well as skill execution data. Images from skill executions in production enable visual inspection of skill start and end positions, supporting quality control and troubleshooting efforts.

Key capabilities:

- Automatic capture of initial and final frames during skill execution
- Transfer and storage of PNG images with skill execution data in CSV format
- Integration with industrial historian systems

Use this feature to diagnose issues and investigate potential anomalies. If multiple MIRAI skills are executed in a single robot program, data for each skill will be sent to the FTP server.

12.5.1 FTP data publishing formats

The FTP protocol transmits two types of data: skill execution images and skill execution details. Files are organized in a single folder using a sequential counter system that tracks execution order. Each skill execution generates image and CSV files with matching counter values, enabling straightforward correlation between visual and performance data.

Skill execution images

First and last images from each skill execution are transmitted as individual PNG files. For dual-camera setups, both cameras will transmit their respective first and last images, resulting in up to four image files per execution.

Image files are named as follows:

[counter]	-	[camera number]	-	[image order]	.	[file type]

Name components:

- **Counter:** Sequential number indicating execution order
- **Camera number:** `vision1` or `vision2`
- **Image order:** `first` or `last`
- **File type:** Standard image format extension

Example image filenames for skill execution 1 using a dual-camera setup:

```
001-vision1-first.png
001-vision1-last.png
001-vision2-first.png
001-vision2-last.png
```

Skill execution details

Skill execution details are transmitted as CSV files. The CSV file contains the same skill execution fields described in 12.2 Skill execution data, with the addition of an execution counter field.

CSV files are named as follows:

[counter]	-	[file name]	.	[file type]

Example CSV filename for skill execution 1:

```
001-params.csv
```

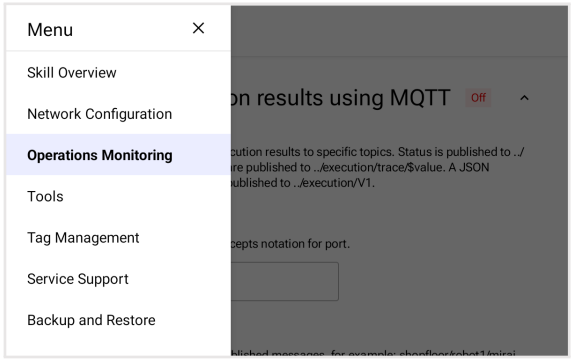
12.5.2 Set up and activate FTP communication

Follow these steps to configure FTP image transfer in the MIRAI Training App:

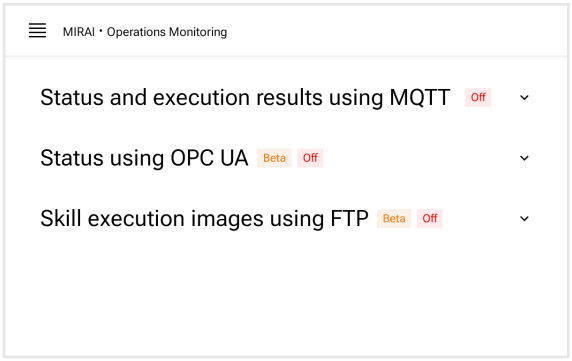
① **Access the FTP screen:**

a. Go to the main menu and tap **Operations Monitoring**.

b. Use the drop-down arrow to expand **Skill execution images using FTP**.



In the main menu, tap **Operations Monitoring**.



Choose an option to configure.

2. **Configure FTP settings:**

   a. In the **Server** field, enter the name or the IP address of the FTP server.

   b. If needed, enter a **username** and **password**. The password is hidden using asterisks.

      - *If your server requires authentication,* leaving the fields blank or entering incorrect inputs will trigger an error pop-up. Tap **OK**, then review and adjust your settings to proceed.
      - *If no authentication is required,* these fields can remain blank.

   c. In the **Directory** field, specify the directory to store the execution images.

3. **Turn on FTP:**

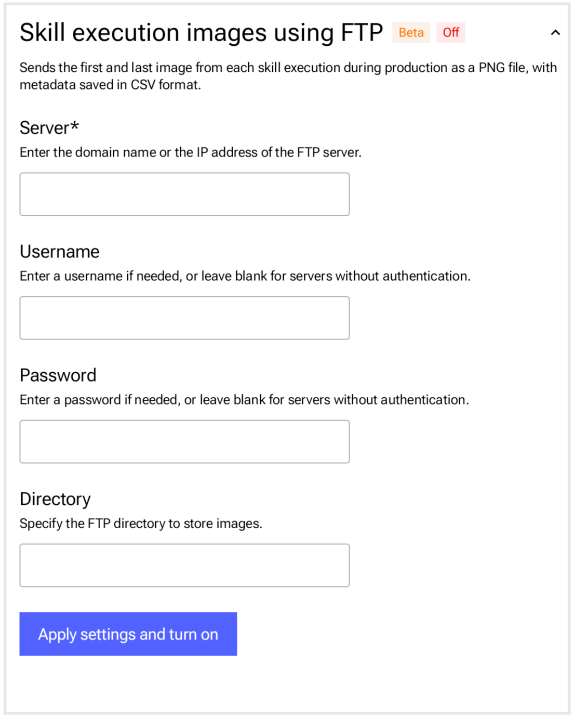
   a. Tap **Apply settings and turn on**.

4. *After activation*, check that the following appear (as shown in the screenshot below):

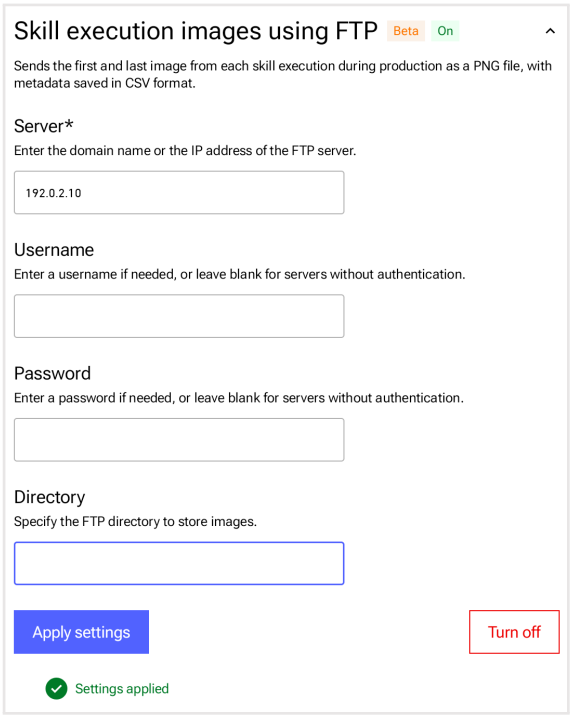
   - **Settings applied** will appear under the activation button.
   - An **On** indicator will appear next to the FTP heading.
   - A **Turn off reports** button will appear at the bottom of the screen.

   IMPORTANT: If you modify the settings, you must tap **Apply settings** for the changes to take effect.

To turn this protocol off, see 12.6 Turning communication protocols off or on.



Fields are empty, communication is **Off**.



Settings applied, communication is **On**.

12.6 Turning communication protocols off or on

The sending of status and skill execution information can be turned off or on at any time. You can view the current activation state for each protocol at the top of the Operations Monitoring screen:

- **On** indicates active communication.
- **Off** indicates inactive communication.

The following instructions apply to MQTT, OPC UA, and FTP.

1. **Access the Operations Monitoring screen:**
   - **a**. Go to the main menu and tap **Operations Monitoring**.
   - **b**. Use the drop-down arrow to expand the protocol of your choice.
2. **Turn communication protocols off or on:**
   - **To turn off a protocol:**
     - Tap **Turn off**.
     - When prompted, confirm by tapping **Turn off** in the pop-up message.
       The status will change to **Off**.
   - **To turn on a protocol:**
     - Tap **Apply settings and turn on**.
       The status will change to **On**.

     NOTE: If you modify the settings, you must tap **Apply settings and turn on** for the changes to take effect.

12.7 Troubleshooting for operations monitoring

This section provides solutions for protocol-specific issues that may arise. Each subsection addresses specific protocols with targeted troubleshooting procedures.

Important considerations:

- Some issues generate explicit error messages while others will not be visible to the user.
- Some issues have multiple possible causes, indicated by uppercase letters. These cases require a sequential diagnostic approach:
  1. Begin by verifying server connectivity and credentials.
  2. Test each potential solution before moving to the next.
- Document any error codes or messages exactly as they appear for more efficient troubleshooting

12.7.1 MQTT troubleshooting

Error messages		
Error message	**Possible causes**	**Solution**
ⓘ Something went wrong. Unable to connect to MQTT broker at <*server*>:<*port.number*>, **Error: Cannot reach the broker.**	**A.** The MIRAI controller is not connected to the LAN. **B.** MQTT broker is offline. **C.** MQTT broker details are incorrect.	**A.** Verify LAN connection (required when saving configuration in the MIRAI Training App). **B.** Verify that the broker is online. **C.** Verify that IP address and port are correct. *NOTE:* If no port is entered, MIRAI defaults to 1883 when user taps "Apply settings." If a different port is used, it must be specified.
ⓘ Something went wrong. Unable to connect to MQTT broker at <*server*>:<*port.number*>, username '<*username*>'. **Server Response: Not authorized.**	Username and/or password are empty or incorrect.	Verify that username and password are correct.

No error messages		
Issue	**Possible causes**	**Solution**
No information is being published to the MQTT broker after communication was configured.	**A. Topic name mismatch**: Topic entered in MIRAI Training App does not match broker configuration. ❗ MIRAI accepts any topic name input without validating against broker settings. **B. Connectivity issue:** The MIRAI controller was powered off *and* the MQTT broker went offline during this period. Upon restart, the controller silently deactivates publishing.	**A.** Verify the topic name is the same in the MIRAI Training App and broker configuration. Check for: • Case (e.g., "Status" vs. "status") • Leading/trailing spaces • Special characters **B.** Check connection: • Verify MQTT is set to **On** in the MIRAI Training App. • Verify that the broker is online. • Check broker parameters (address, port, credentials). • Confirm broker connectivity by tapping **Apply settings**.

12.7.2 OPC UA troubleshooting

Error messages		
Error message	**Possible causes**	**Solution**
ⓘ Something went wrong. **[Errno -2] Name or service notknown**	An extra space has been entered in the **Server** field.	Remove the space. Check for trailing spaces, which may not be obvious.
ⓘ Something went wrong. **[Errno -5] No address associated with hostname**	Missing or incorrect character in the **Server** field.	Verify server information is correct.
ⓘ Something went wrong. Unable to connect to OPC UA server at <*server*>. **[Errno 111] Connection refused**	**A.** OPC UA server is offline. **B.** OPC UA server details are incorrect.	**A.** Verify that the server is online. **B.** Verify server information is correct. Check IP address and port. *NOTE:* Only enter the port if it is different from the default OPC UA port 4840.
ⓘ Something went wrong. Unable to connect to OPC UA server at <*server*>. **[Errno 113] No route to host**	OPC UA server details are incorrect.	Verify server information is correct. Check IP address and port. *NOTE:* Only enter the port if it is different from the default OPC UA port 4840.

No error messages		
Issue	**Possible causes**	**Solution**
No information is being published to the OPC UA server after communication was configured. No error message appears.	**Connectivity issue:** The MIRAI controller was powered off *and* the OPC UA server went offline during this period. Upon restart, the controller silently deactivates OPC UA communication.	Check connection: • Verify that the server is online. • Check broker parameters (address, port, credentials). • Verify OPC UA is set to **On** in the MIRAI Training App. • Confirm server connectivity by tapping **Apply settings**.

12.7.3 FTP troubleshooting

Error messages		
Error message	**Possible causes**	**Solution**
ⓘ Something went wrong. **[Errno -2] Name or service notknown**	An extra space has been entered in the **Server** field.	Remove the space. Check for trailing spaces, which may not be obvious.
ⓘ Something went wrong. **[Errno -5] No address associated with hostname**	Missing or incorrect character in the **Server** field.	Verify server information is correct.
ⓘ Something went wrong. Unable to connect to FTP server at *<server>*. **[Errno 111] Connection refused**	**A.** FTP server is offline. **B.** FTP server details are incorrect.	**A.** Verify that the server is online. **B.** Verify server information is correct. Check IP address and port. *NOTE:* The MIRAI Training App does not automatically assign a port number.
ⓘ Something went wrong. Unable to connect to FTP server at *<server>*. **[Errno 113] No route to host**	Server details are incorrect.	Verify server information is correct. Check IP address and port. *NOTE:* The MIRAI Training App does not automatically assign a port number.
ⓘ Something went wrong. An unexpected error occurred. The original error was error_perm **('530 Authentication failed')**	Username and/or password is incorrect.	Verify that username and password are entered correctly.

No error messages		
Issue	**Possible causes**	**Solution**
No execution images or CSV files are received in the intended directory on the FTP server. No error message appears.	**A. Directory name mismatch**: Directory entered in the MIRAI Training App does not match server. ❗ MIRAI creates a new FTP directory if the name entered in the App does not match existing directories. **B. Connectivity issue:** The MIRAI controller is not connected to the LAN.	**A.** Verify the directory name is the same in the MIRAI Training App and the FTP server. Check for: • Case (e.g., "Status" vs. "status") • Leading/trailing spaces • Special characters **B.** Verify that the MIRAI controller is connected to the LAN.

13 Shared Skills - Using a Skill on Different Setups

MIRAI controllers that belong to the same group will automatically share their skills with each other, allowing you to use the same skill on different setups.

Note

To add a MIRAI Controller to your group, **please contact a Micropsi Industries representative**.

13.1 Preparing Your Physical Setup for Syncing

- To use a skill created on another MIRAI controller, you first need to synchronize that skill to your controller. This involves making sure that the hardware setup is suitable for the skill.
- To successfully synchronize skills across MIRAI controllers or setups, consider the following:
  - **Ensure that the applicable MIRAI controllers are added** to your group.
  - **Ensure that your physical setup is complete and ready.** Use the following table as your guide. Nonetheless, the listed components on the table are also covered in the skill synchronization process in the next section, 13.2 Syncing a Skill to Your Setup.

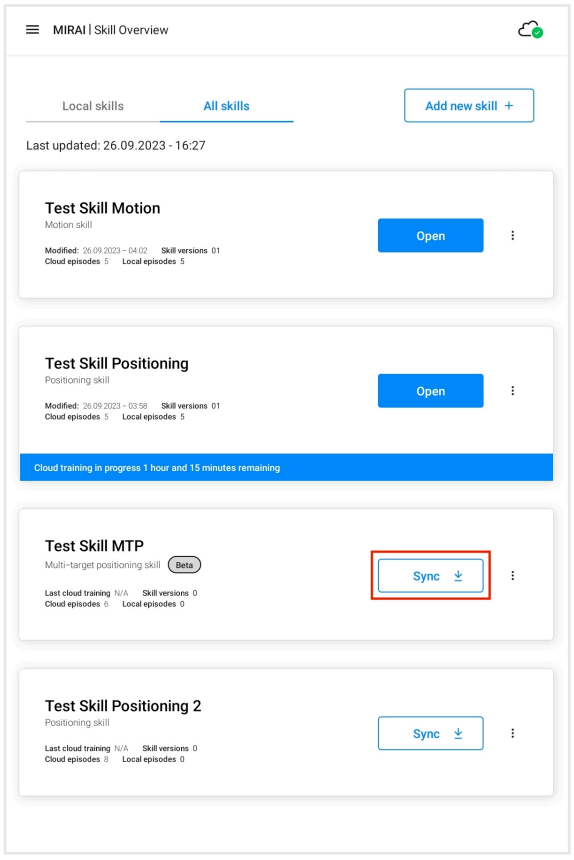
Physical Setup Components		Does it need to be identical across all Setups involved?	Recommendations / Tips for Your Physical Setup	Selection or Confirmation during skill synchronization in the MIRAI Training App
Robot Setup	**Robot**	**No**	To ensure a well-functioning skill, it is highly recommended to use the same robot model as the one on the original skill.	You can technically select any robot on the list provided by the app.
	Force-torque Sensor	**No**	Ensure that the camera view is identical across setups. When syncing a skill trained with a force/torque sensor to a setup without, special adjustments are necessary.	You can technically select any available force/torque sensor or "None."
	Tool Center Point (TCP) Displacement	**Yes**	Ensure that the TCP displacement settings are matched. This is important as the end effector would rotate according to the original TCP displacement values set.	This cannot be modified in the app.
Camera: Physical Setup and Settings	**Camera**	**Yes**	Ensure that the camera model is the same. It is not recommended to sync a skill created on a USB camera setup to a setup with a GigE camera, or vice versa.	You should select the appropriate camera in the app.
	Number of Cameras Connected	**Yes**	Cannot be modified. Ensure that the correct number of cameras are connected.	This cannot be modified in the app.
	Camera Mount Setup	**Yes**	Cannot be modified. Ensure that how the cameras were mounted are the same (i.e., "Wrist mounted" or "Static mounted").	This cannot be modified in the app.
	Camera Lens	**Yes**	It is highly recommended to use the same camera lens to ensure it has the same focal point as on the other setup.	You do not need to confirm the lens used in the app.
	Aperture & Focus	**No**	These are physical settings found on the camera itself and affect all other skills within a setup. Be careful when changing these settings.	You do not need to enter the aperture and focus settings in the app.
	Gain & Exposure	**No**	What matters the most is that the live camera view and reference images shown on the app screen look as similar as possible.	You need to check if these need to be adjusted while going through the Syncing process on the app.
Tool Setup	**End effector**	**Yes**	Ensure that you use the same end effector.	You should select the appropriate tool in the app.
	Tool Mass & CoG Values	**Yes***	* The values need to be similar; however, they may end up being slightly different.	* You need to go through the tool configuration process in the app separately. For more information on this, see section 4.3 Configure a tool.

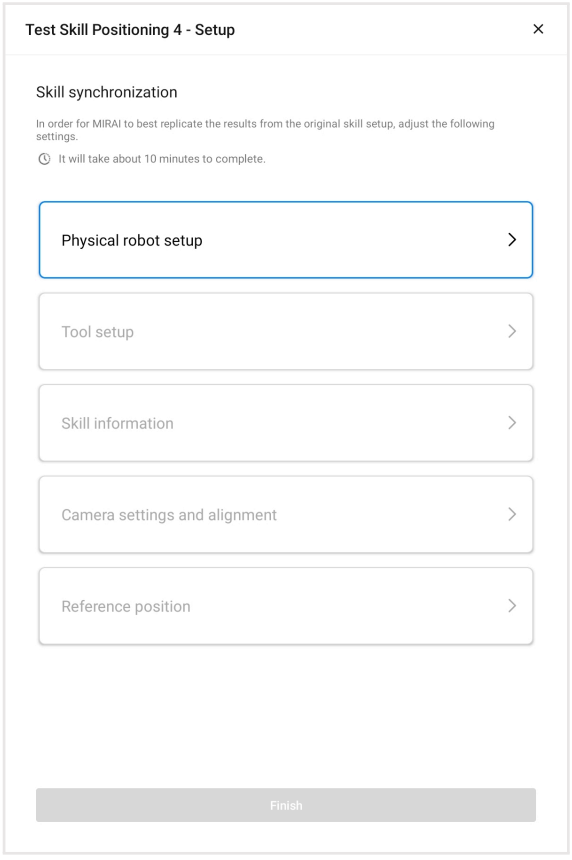
13.2 Syncing a Skill to Your Setup

Important

- ☑ The MIRAI controller you are syncing the skill to (i.e., the controller you are currently working with) must be **connected to the Micropsi cloud server** during the process. If the controller is not connected to the Micropsi cloud server, all the skills available in your group – based on their last uploaded versions – will still be shown; however, the **Sync** button will be greyed out. For more information on why a cloud connection is needed, please refer to chapter 17 Data FAQ
- ☑ For the skill to demonstrate results as similar to the original setup as possible, it is very important to **check and enter the appropriate settings** on each of the steps to **replicate the original conditions**.
- ☑ Ensure that the **camera view is identical across setups**. When syncing a skill trained with a force/torque sensor to a setup that does not use a force/torque sensor, special adjustments are necessary.

- To select a skill to synchronize, go to the "Skill Overview" page, and select the "All Skills" tab. Then tap **Sync** to the right of the skill name.
- Proceed with Skill synchronization. This process will involve five major steps (detailed below). The MIRAI app will guide you through each step. To begin, tap **Physical robot setup**. Then continue with the directions below until the last step.



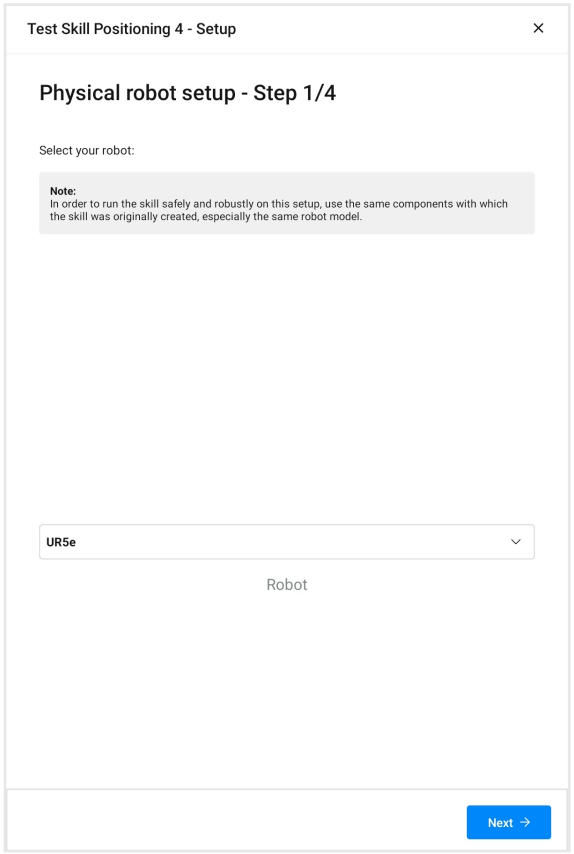


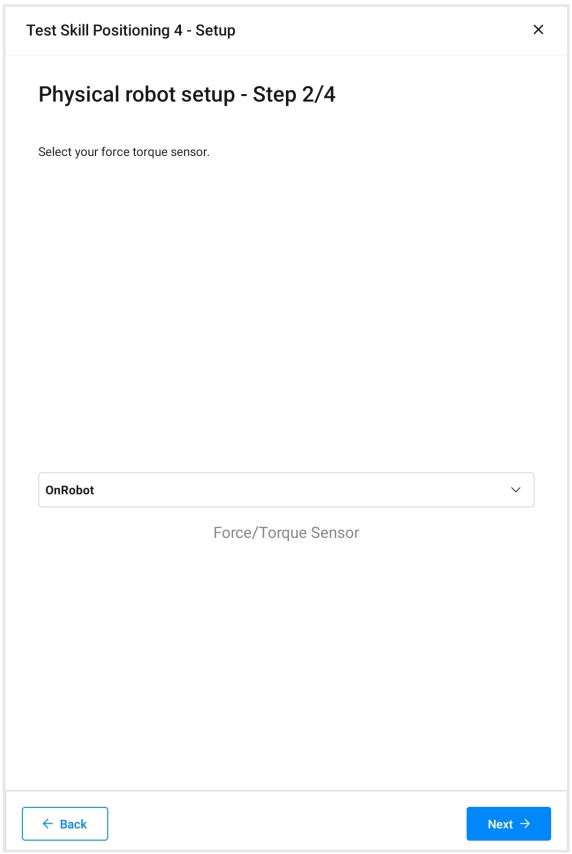
① **Physical Robot Setup:**

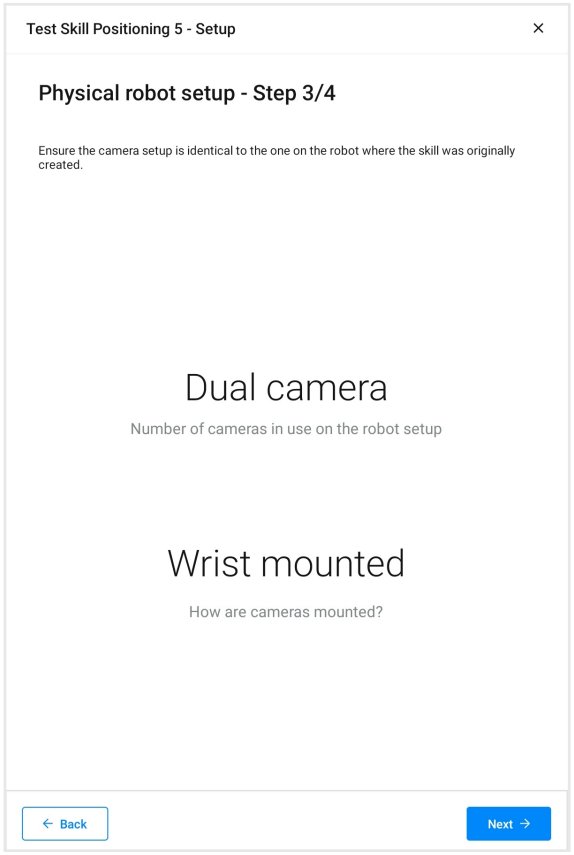
- Select the robot brand and model you are using from the "Robot" drop-down menu, then tap **Next**.

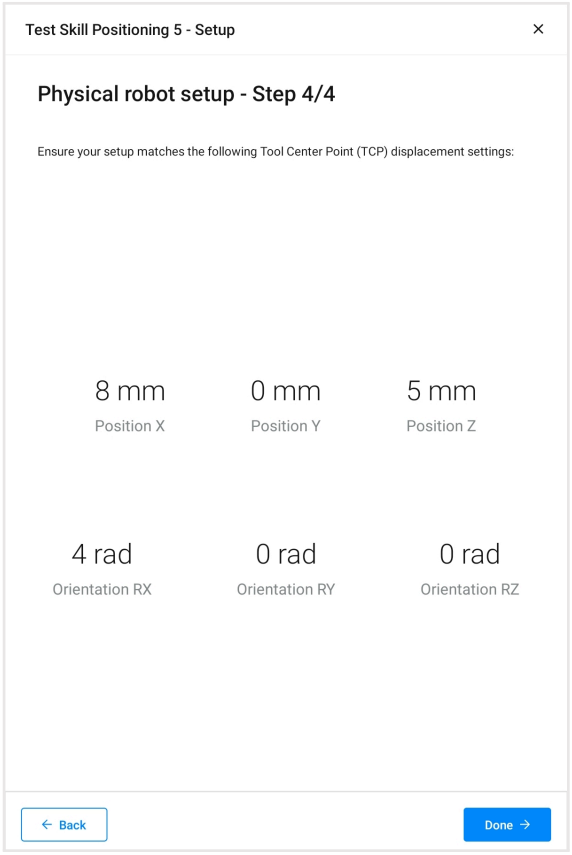
Note: You can technically select any robot on the list; however, to ensure a well-functioning skill, it is **highly recommended to use the same robot model** as the one on the original setup.

- Select the force/torque sensor you are using from the Force/torque sensor drop-down menu, then tap **Next**. *Note:* It is **highly recommended to use the same force/torque sensor** as the one on the original skill. If you use a different one, this may change the length of the stack with the end-effector.
- Make sure that the **camera setup you are using is identical to what is indicated** on the screen (i.e., the settings on the original skill), such as how many cameras were used, and how the camera was mounted; tap **Next** to confirm (*Note:* You cannot modify these settings at this stage). As mentioned in the checklist above, the lens to be used should also be the same one as on the original skill.
- Make sure **your setup matches the Tool Center Point (TCP) displacement settings indicated** (X, Y, and Z displacement values, and RX, RY, and RZ angular values) on the screen. This is important as the end effector would rotate according to the original TCP displacement values set. Tap **Done** to confirm and proceed Step 2 (*Note:* You cannot modify these settings at this stage).



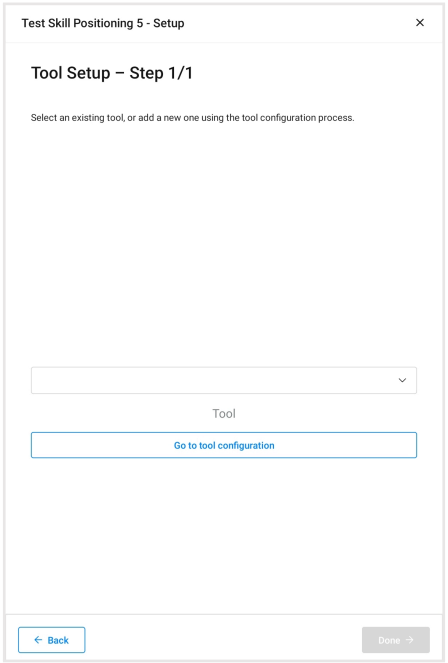






2 **Tool Setup:**

Select the tool from the "Tool" drop-down menu (i.e., Gripper, Picker, or Plug), then tap **Done** to proceed to the next step.

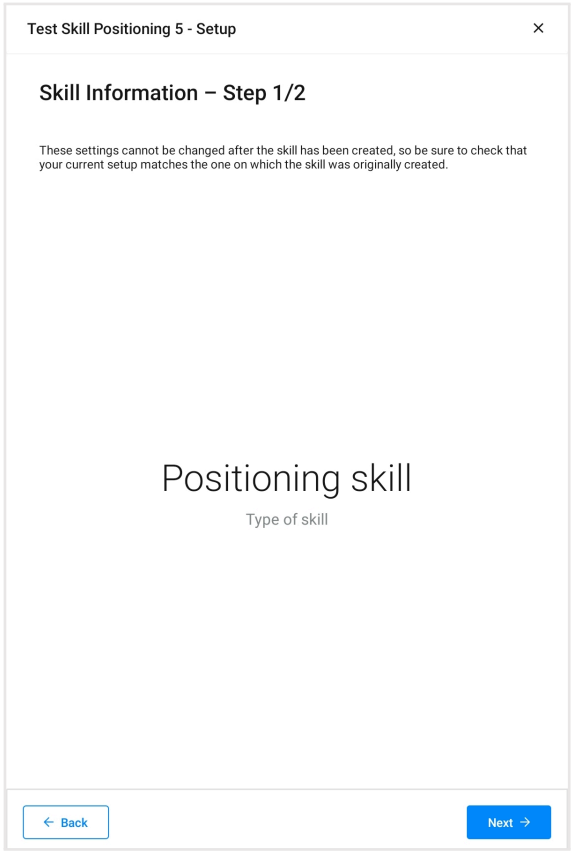


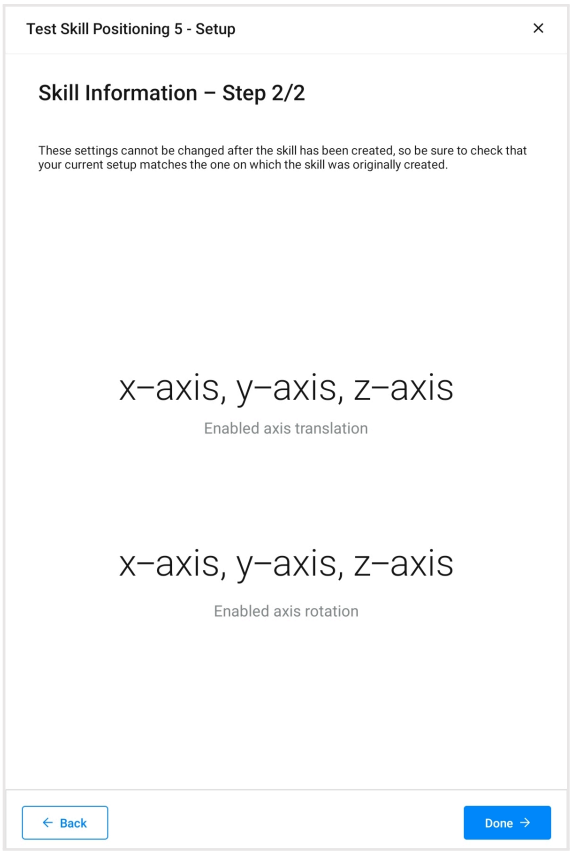
Note

- Ensure that you use **the same end effector**.
- If you did not use a tool previously, it is recommended to go through the **tool configuration process**, found in section 4.3 Configure a tool.
- If the skill you selected to synchronize has no rotations, this step will be hidden.

③ **Skill Information:**

- Confirm the type of skill (i.e., Positioning skill, Motion skill, or Multi-target Positioning skill) by tapping **Next** (*Note:* You cannot modify these settings at this stage).
- Confirm the enabled axis transitions and the enabled axis rotations (i.e., x-axis, y-axis, z-axis) by tapping **Done** (*Note:* You cannot modify these settings at this stage).

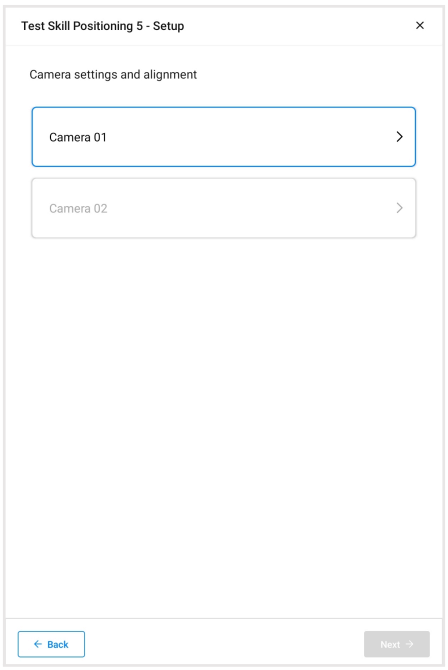




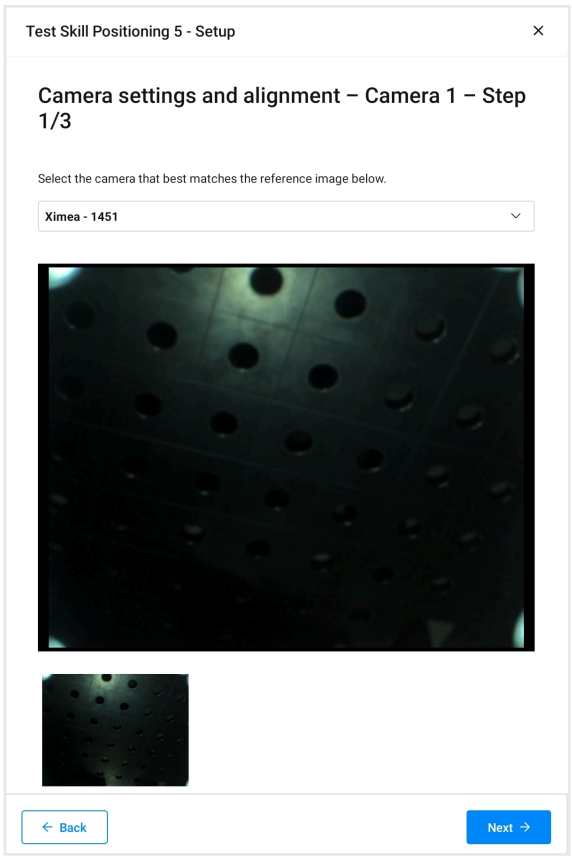
④ **Camera Settings and Alignment:** Depending on how many cameras your setup has, go through the three (3) steps listed below for **each** camera.

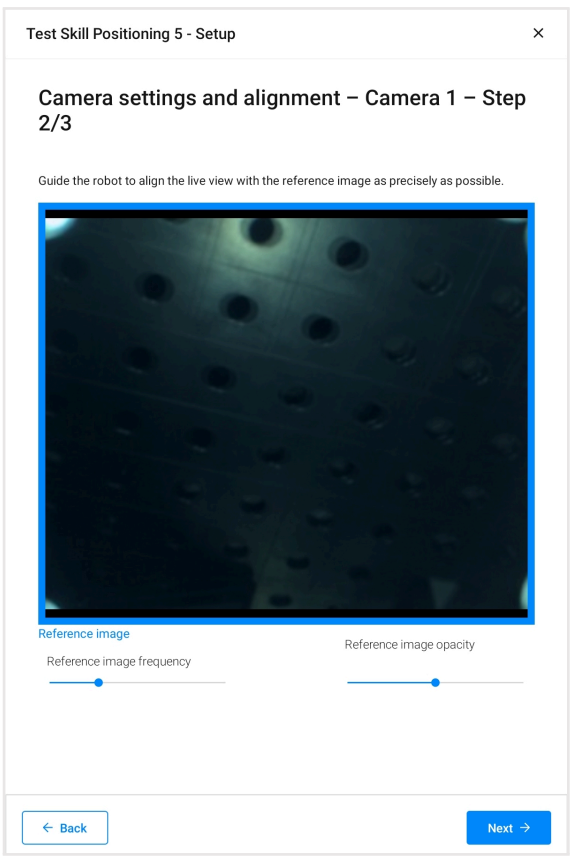
Note

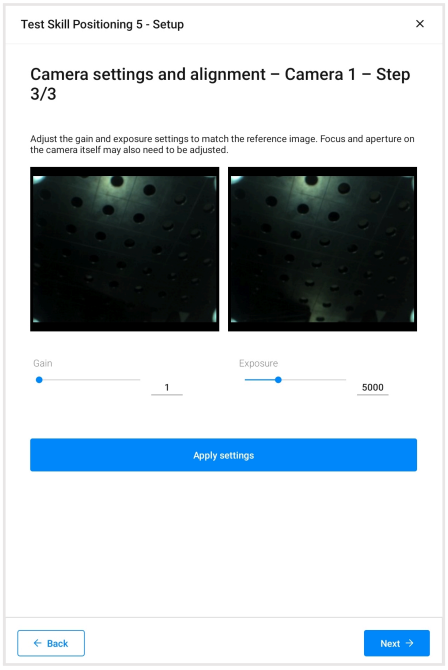
For **GigE cameras** to appear in the drop-down menu, their IP address must be configured to the correct subnet and they must be selected in the network settings of the MIRAI Training App. You can find instructions on how to correctly configure the IP address of GigE cameras in the Robot Integration Guides.



- Using the drop-down menu, select the camera that matches the reference image shown to you on the screen, then tap **Next**. (*Note:* The reference image shown comes from the original skill setup).
- Guide the robot to align the live camera view with the reference image as precisely as possible. The sliders are meant to aid you with ensuring that the reference image and the live camera view image are as aligned as possible.
    * **Reference image frequency** refers to how frequently the reference image would blink over the live camera view image. That said, you should drag the slider to the right side if you want the reference image to blink more frequently.
    * **Reference image opacity** refers to how opaque the reference image overlay would be. Given this, you should drag the slider to the left side if you want the overlay to be minimal (i.e., more transparent).
- Adjust your camera settings to match the reference image shown as closely as possible.
    * Please check the gain and exposure and if needed, adjust these via the sliders. **What matters the most at this point is that the live camera view and reference images shown on the screen look as similar as possible.** That said, if your current gain and exposure settings are not exactly the same as the one used on the original skill, it is recommended that you adjust them so that the images on the screen match as closely as possible.
    * Additionally, check the camera's focus and aperture (*Note:* These are physical settings found on the camera itself; they also affect all other skills, so please be very careful when changing these settings).
    * Tap **Apply settings**, and then **Next** if you are content with the settings you have entered.

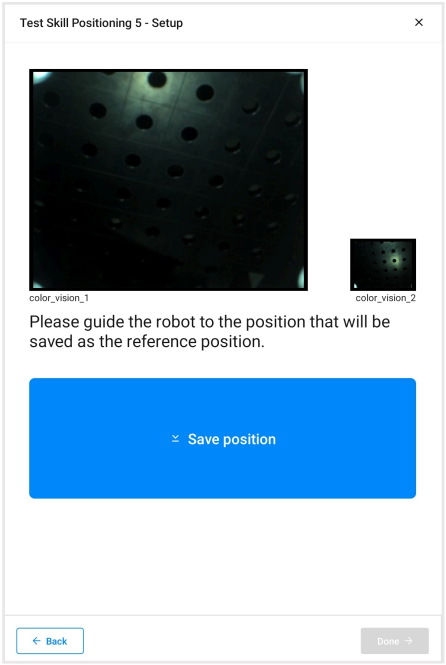




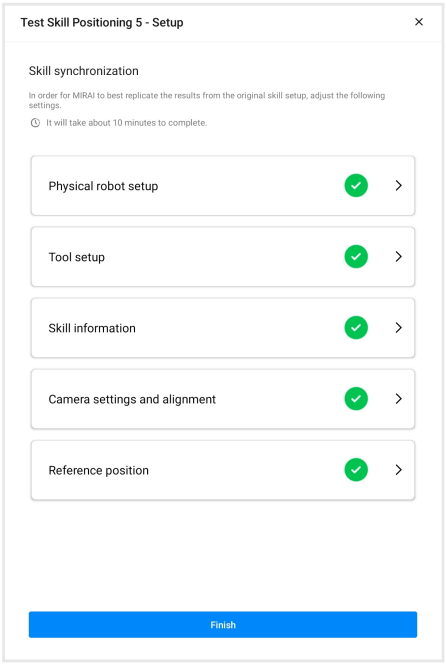


5 **Reference Position**

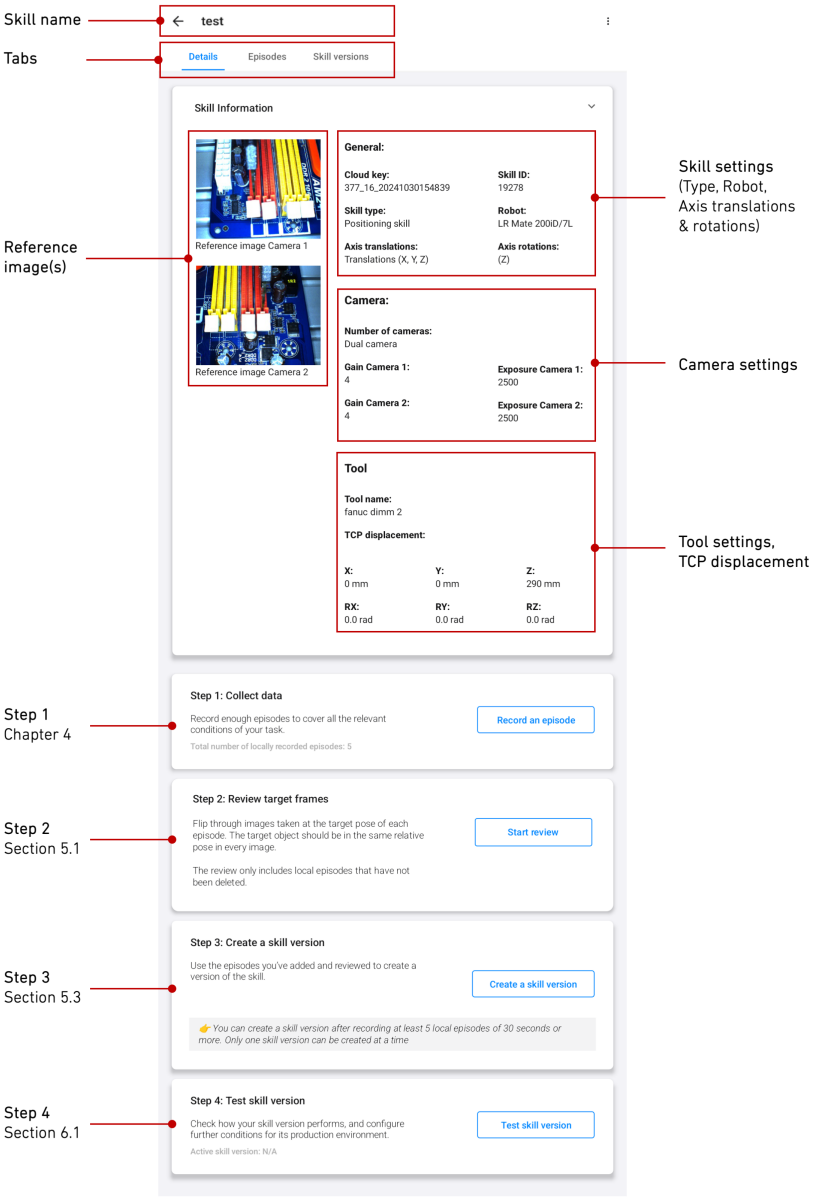
Guide the robot to the position that will be saved as the reference position. Tap **Save position**, and then **Done**.



6 Once you have finished all the Skill synchronization steps, tap **Finish**.



7 The "Details" screen will then show the information of the shared skill. You may select what to do next with the skill (e.g., Record episodes, Start cloud training, or Test & configure).



13.3 Recording Additional Episodes for Your Setup

Recording additional episodes for a shared skill would essentially be the same process as recording episodes for a newly created skill. Nonetheless, below are some tips in making a shared skill more robust:

- **Skill Type:** Ensure that when recording more episodes for the shared skill, it follows the recommended approach for the specific skill type (e.g., if it is a Positioning skill, move the robot in a spiral motion away from the target and scan all the relevant points in the space surrounding it — as described in section **?? ??**).
- **Degrees of Freedom:** Ensure that when recording more episodes, you move in all the originally enabled axis translations and rotations (e.g., if it is a z-axis rotation skill, make sure to record data

showing rotations in the z-axis).

- **Reference Position:** This needs to be configured on every MIRAI Setup. Ensure that you define this when syncing (step 7 in the skill synchronization process).
- **End States:** This, too, needs to be configured on every MIRAI Setup. End states can be set within Execution Settings inside the Testing loop (see section 11 End State Triggers).

Note

In the Episodes tab, only episodes recorded on the MIRAI Controller you are currently working with can be seen. On the other hand, when you request a new skill version, **it will be trained on all the episodes** from all the MIRAI controllers within your group. See section 9.2 MIRAI Cloud Training and Skill Versions for more details on how to request a skill version.

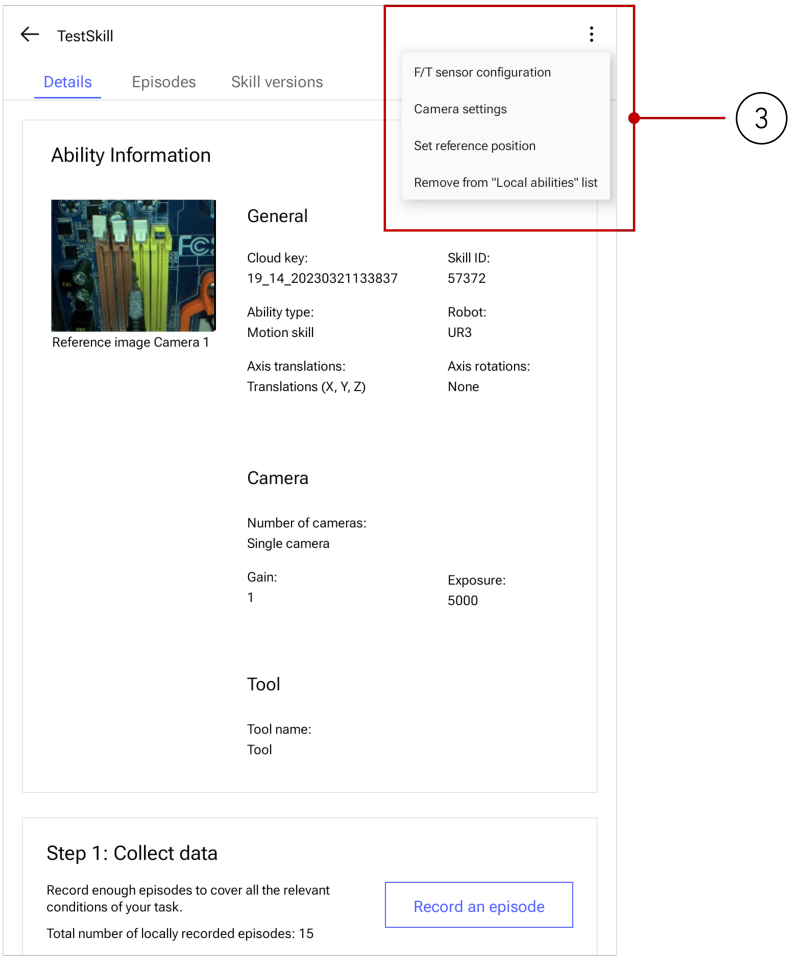
14 Adjusting skill settings after creation or sharing

After a skill is created or shared, most of its configuration (for example, skill type, robot model, rotations) cannot be changed. However, certain settings can be updated after skill creation or sharing to adapt to changes in hardware or the surrounding environment—without starting from scratch. These include:

- **Configuring sensor usage**
- **Adding or changing a tool**
- **Managing camera alignment and focus** (for example, if a camera is replaced or misaligned)
- **Setting a new reference position**
- **Removing a skill from the Local abilities list**

To adjust a skill after creation or sharing:

1. Open the main menu (☰), then tap **Ability overview**.
2. Select the skill you want to adjust, then tap **Open**.
3. On the skill's Details tab, with Ability information displayed, tap **More options** (⋮) in the top-right corner, and select the setting you want to adjust.



The following sections describe how to use each skill setting adjustment feature in detail.

14.1 Changing force/torque sensor usage and tool

A skill can be configured to use a force/torque (F/T) sensor or to run without one, even if this setting differs from how the skill was originally created.

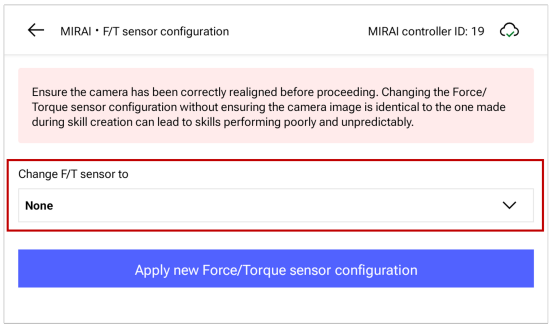
IMPORTANT

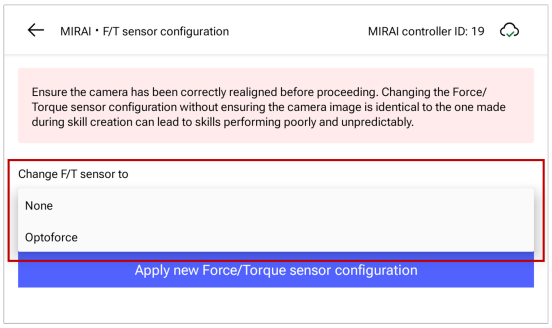
Ensure that the camera settings are correct before proceeding. Changing the force/torque sensor configuration without first verifying that the camera image matches the one used during skill creation can cause skills to perform poorly or unpredictably. Refer to Section 14.2 for instructions on managing camera alignment and focus.

To add or change the F/T sensor and tool:

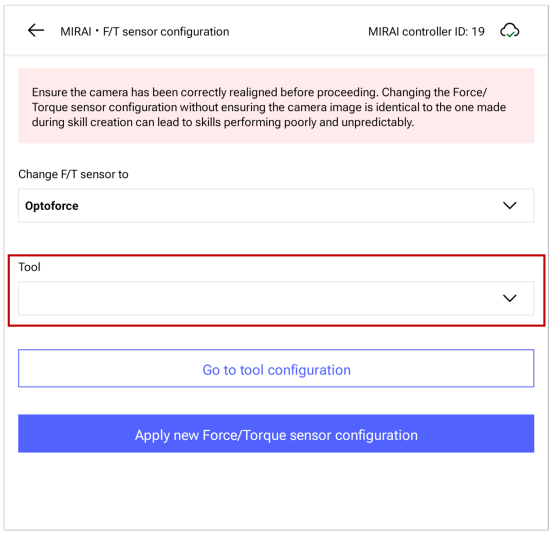
1. Under **More options** (⋮), select **F/T sensor configuration**.

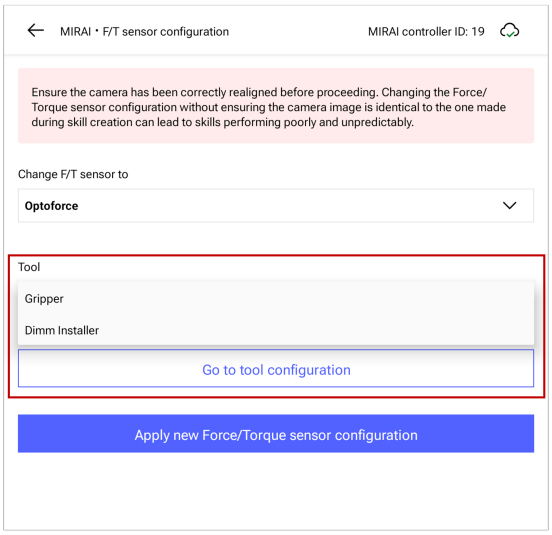
2. Under **Change F/T sensor to, select the sensor** to be used.
NOTE: If no force/torque sensor is connected or detected, only "None" is available.



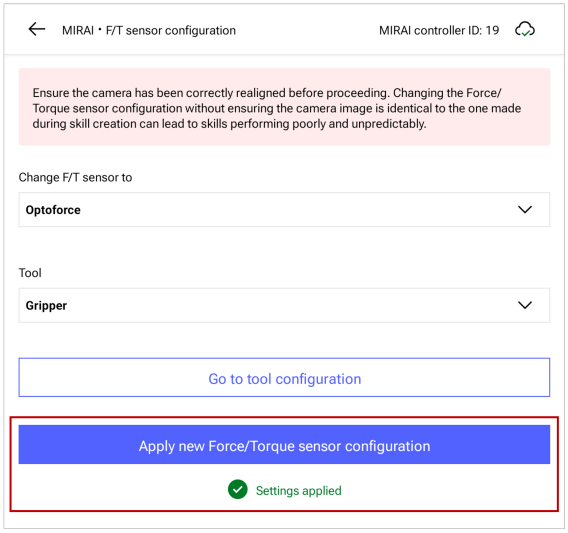


3. If applicable, **select a tool from the Tool drop-down menu**, or tap **Go to tool configuration** to set up a new tool (see Section 4.3 for more details).



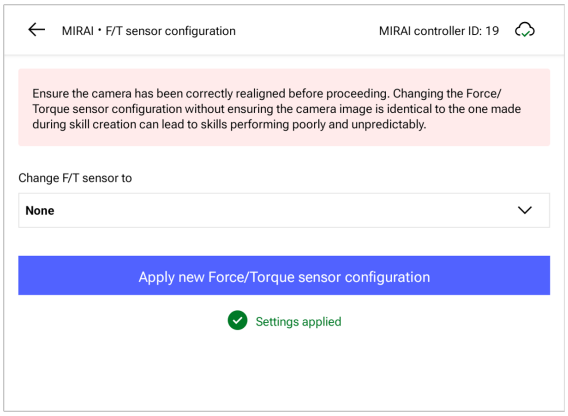


4. Tap **Apply new force/torque sensor configuration** to save the changes. **Settings applied** will appear below.



To remove the F/T sensor:

1. Under **More options** (⋮), select **F/T sensor configuration**.
2. Under **Change F/T sensor to, select "None"**.
3. Tap **Apply new force/torque sensor configuration** to save the changes. **Settings applied** will appear below.



14.2 Managing camera alignment and focus

Camera alignment and focus management allow you to restore the visual conditions under which a skill was originally created, ensuring that the skill continues to execute as intended, and that new recordings have the same visual reference. These settings are useful if a camera has been replaced, misaligned, or repositioned, or if changes in the environment affect image quality.

IMPORTANT

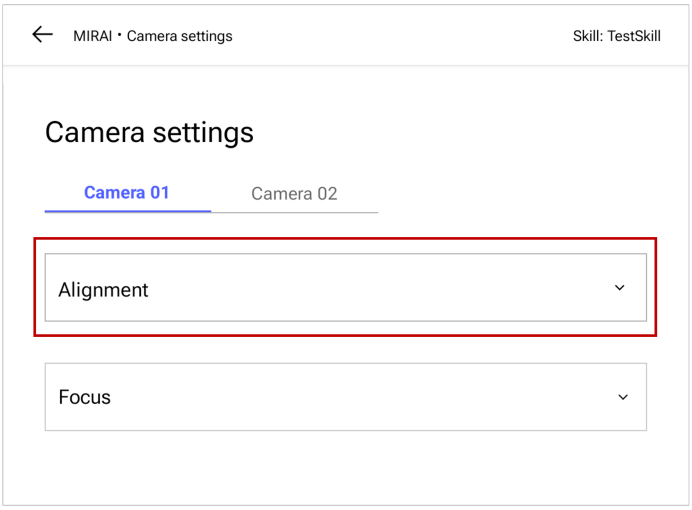
When using **GigE cameras**, ensure that their IP addresses are configured to the correct subnet. Additionally, they should be selected in the Network Configuration settings of the MIRAI app. For detailed instructions, please refer to the corresponding robot integration guides.

14.2.1 Aligning camera view with the reference

Camera alignment ensures that the camera view matches the reference image captured during skill creation and/or training. This allows you to assign the correct camera to the right view, as well as align the live camera image with the stored reference image.

To adjust the camera alignment:

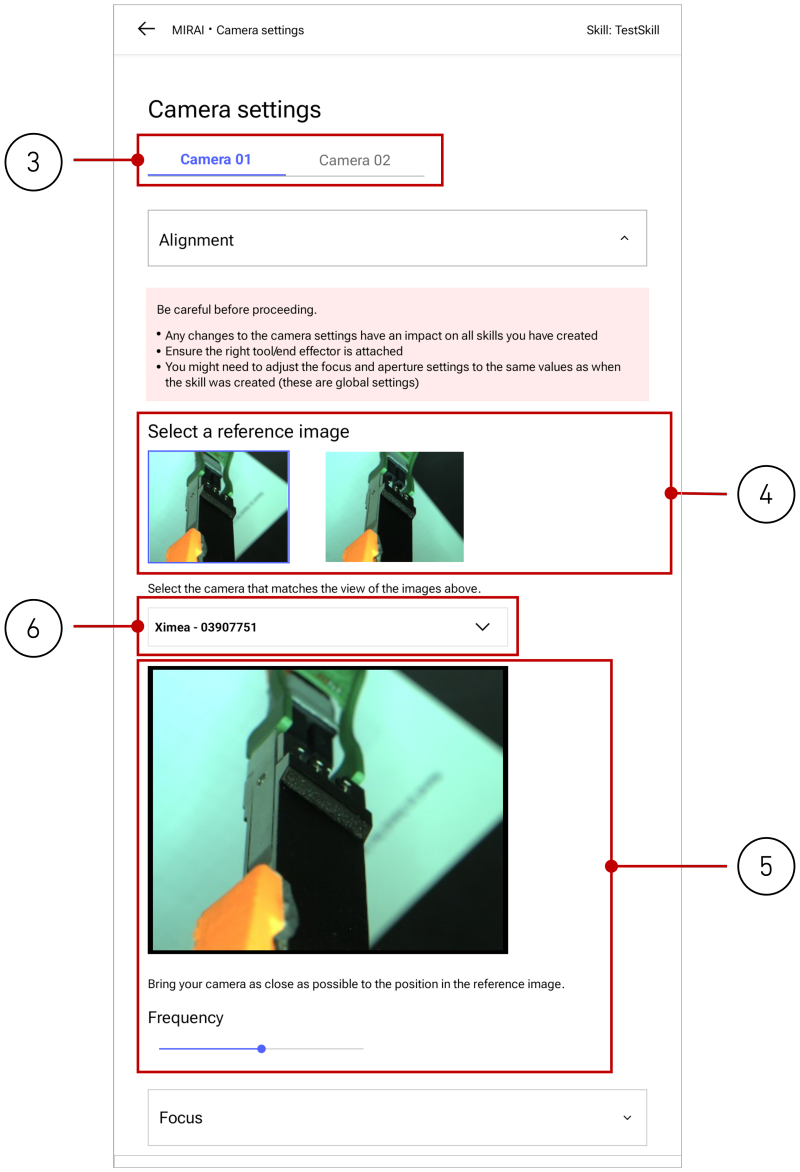
1. Under **More options** (⋮), select **Camera settings**.
2. In the Camera settings screen, tap **Alignment**.



3. For a dual-camera setup, **use the tabs to select the camera** you want to view or configure.
4. Under Select a reference image, **choose the image which the live camera feed should be compared to**.
   TIP: If two images are available, we recommend selecting the image on the right (the most recent reference image based on the last five recorded episodes).
5. Below the camera dropdown, the **camera view and the reference image will alternate** (flashing one after another) on the screen to help with comparison. The **Frequency slider** below adjusts the frequency of image flashing. Use this feed to adjust the camera position and match it with the reference image as closely as possible.

NOTE: Even slight pixel deviations can have an immense impact on the quality of the skill, so make the adjustments as accurate as possible.

6 When necessary, **select a different camera** from the dropdown menu. If a different camera is selected, a pop-up *"Are you sure you want to rematch cameras?"* appears. Tap **Yes, rematch cameras** to proceed.



14.2.2 Reproducing original focus settings

Camera focus recovery enables you to reproduce the visual sharpness of the reference image used during skill creation. The system provides a focus score for both the reference image and the live camera view.

The following information is displayed to help with reproducing the original settings as closely as possible:

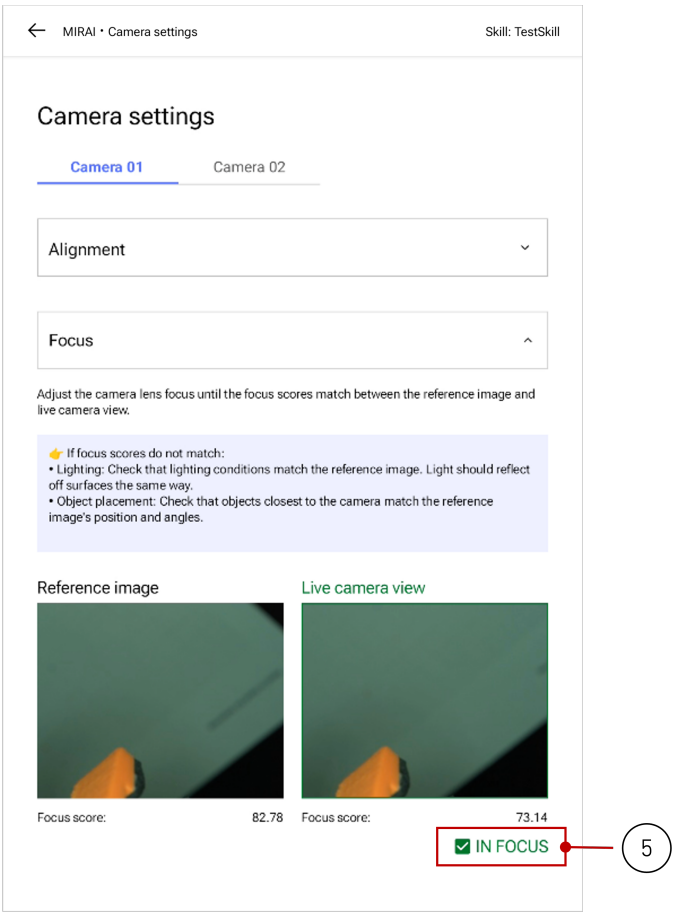
- A focus score for the reference image
- A focus score for the live camera view
- A focus status indicator that updates as conditions change

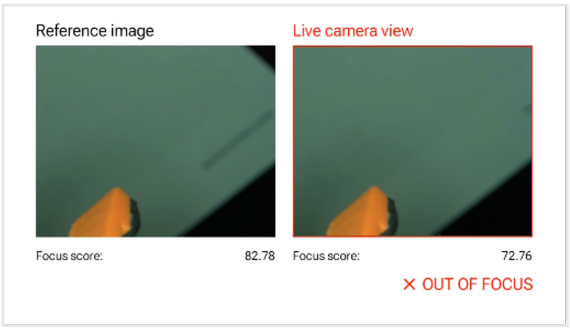
The focus status can have the following meanings:

- ✔ **IN FOCUS**: The focus score of the live image closely matches the reference image. The skill will perform as expected.
- ✖ **OUT OF FOCUS:** The images are similar, but the live image is not sharp enough.
- ✖ **Images too different**: The live camera view differs significantly from the reference image (for example, due to lighting changes, object placement, or camera position).

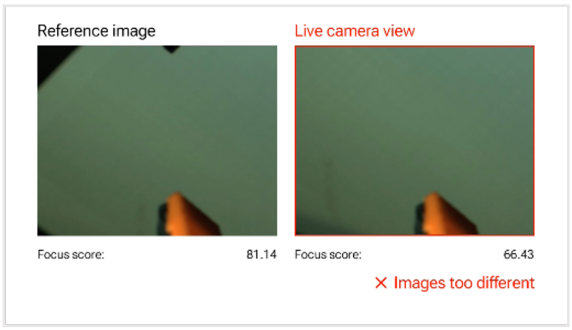
To reproduce the camera's original focus settings:

1. Under **More options** (⋮), select **Camera settings**.
2. For a dual-camera setup, **use the tabs to select the camera** you want to view or configure.
3. In the Camera settings screen, tap **Alignment** first. Under **Select a reference image**, choose the image that the live camera feed's focus setting should be compared to.
   TIP: If two images are available, we recommend selecting the image on the right (the most recent reference image based on the last five recorded episodes).
4. Tap **Focus** and ensure that the selected reference image above appears under **Reference image**.
5. Manually adjust the camera focus (using the focus ring on the lens) and if necessary, the surrounding conditions (such as lighting or relevant objects) until the focus status indicates ✔ **IN FOCUS**.
   NOTE: The arrangement of items in the scene does not need to match the reference image exactly. However, for reflective parts, their distance from the camera can influence the focus score; thus, this should be kept similar to the reference setup.





✕ **OUT OF FOCUS**



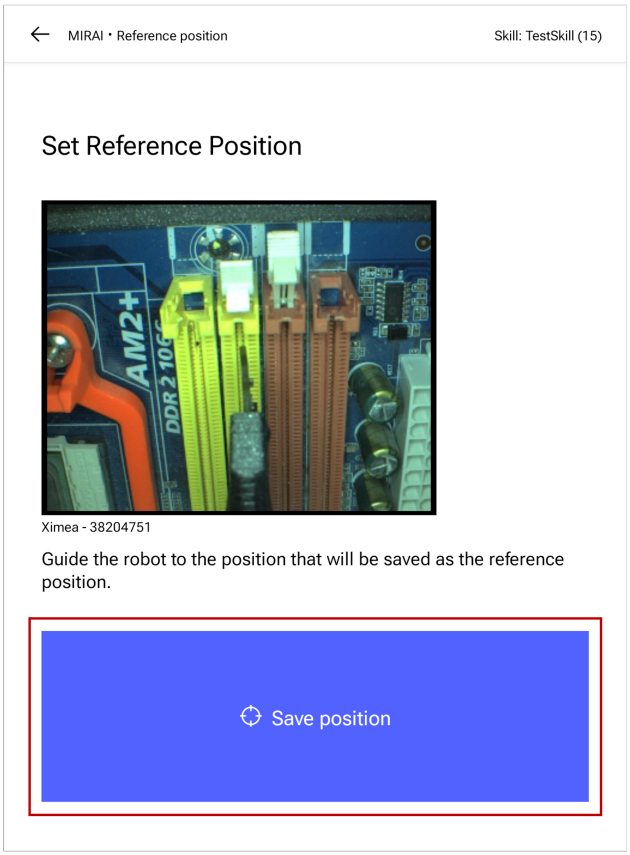
✕ **Images too different**

14.3 Setting a new reference position

You can define a reference position different from the one set during skill creation or sharing.

To set a new reference position:

1. Under **More options** (⋮), select **Set reference position**.
2. **Guide the robot to the desired position**, then tap **Save position**.



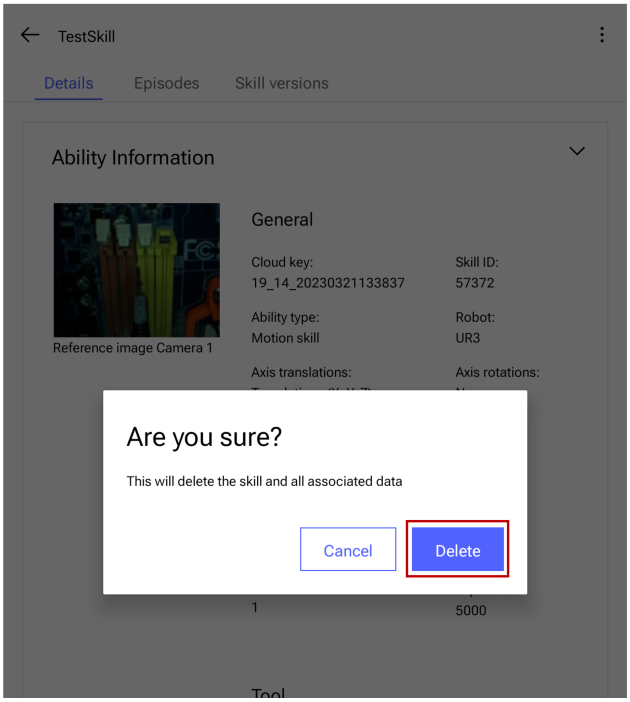
14.4 Removing a skill from Local abilities

Implications

"Removing" a skill deletes it from the Local abilities list, but it is not permanently lost. It can be recovered through skill sharing; however, the skill will then behave according to the rules of skill sharing (For more information, see Section 13)

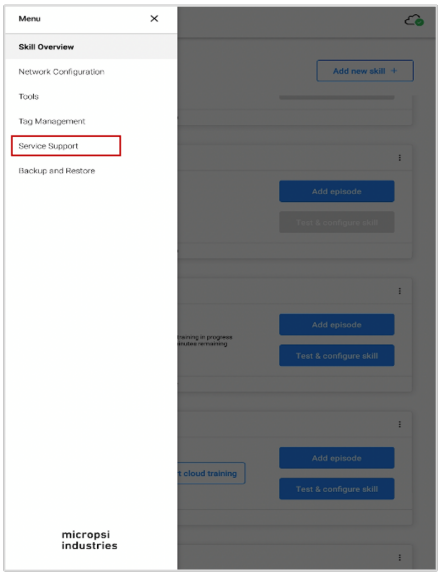
To remove a skill from Local abilities:

1. Under **More options** (⋮), select **Remove from Local abilities**.
2. On the pop-up *"Are you sure?"*, tap **Delete** to proceed.



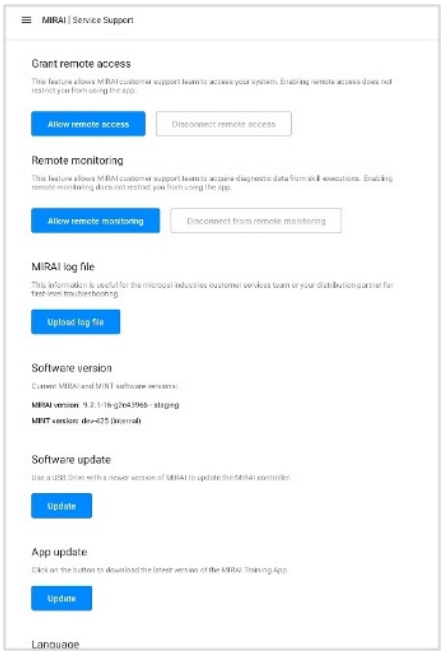
15 Service Support

In the main menu (top left corner), select **Service Support**.



15.1 Enabling Remote Access

The first section, Grant remote access, enables the MIRAI customer support team to access your controller for troubleshooting. **Enable this feature only when asked by our support staff**.

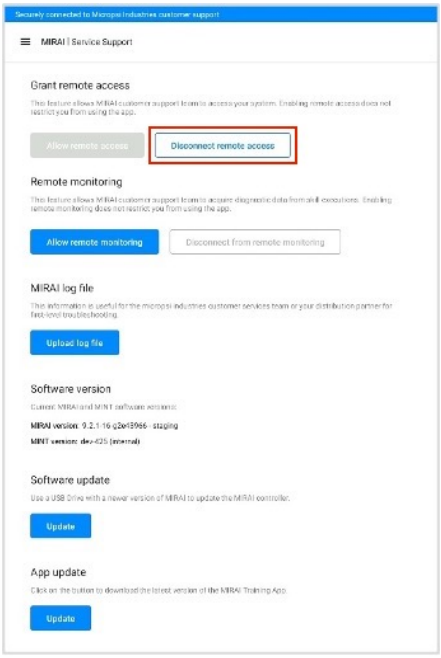


- Once the connection is successfully established, a blue bar appears on the top end of the screen to indicate that the remote access is active. A successful connection is indicated by the Under remote access status bar. You can continue using the app while being connected to remote access.

- To disconnect, select **Disconnect from remote access** in the same screen.

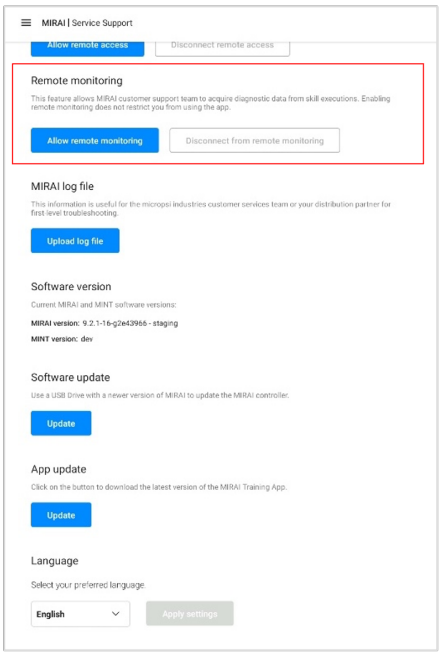
Important

The connection needs to be turned off **manually**! Remote access would usually not be automatically terminated by the MIRAI customer support team.



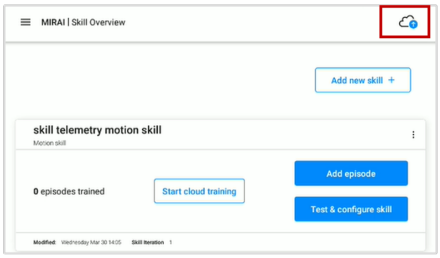
15.2 Enabling Remote Monitoring

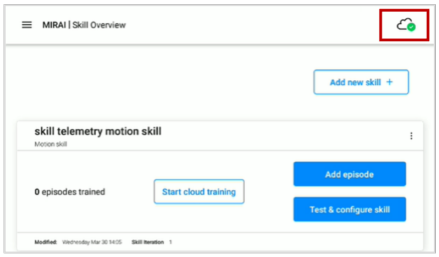
If a MIRAI skill does not execute properly after transferring it to the native controller (PolyScope for UR robots, Teach Pendant for FANUC), remote monitoring enables the Micropsi Industries customer support team to investigate the execution of this skill. **Enable this feature only when asked by our support staff.**



To enable remote monitoring:

1. Tap **Allow remote monitoring**.
2. Run the program on the native controller and wait until the end of the skill execution.
3. Connect the MIRAI controller to the internet (skip this step if you are already connected).
4. Wait for the “Upload in progress” sign to turn green on the MIRAI Training app.



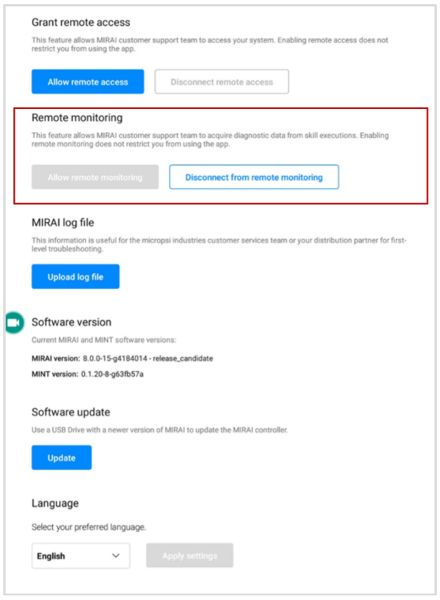


5. You may disconnect from the internet.

Note

For more details on the connection to the Micropsi cloud server and how Micropsi Industries handles recorded data please refer to chapter 17 Data FAQ.

Remote monitoring turns off automatically after 4 hours without further action needed from your side. Alternatively, you can disable it by tapping the **Disconnect from remote monitoring**.



15.3 Updating the MIRAI Controller Software

This section outlines the steps to update the software version on the MIRAI controller. Contact the Micropsi Industries service team to obtain the latest software release.

Important

During the update process, ensure that the MIRAI controller can connect to the following hosts:

- https://apt.tools.micropsi.io (TCP Port 443)
- http://deb.debian.org (TCP Port 80)
- http://security.debian.org (TCP Port 80)

Step 1: Prepare a USB drive

a. **Obtain a USB Drive:**

- Use a USB drive formatted with FAT32 (ideally 16GB or larger).

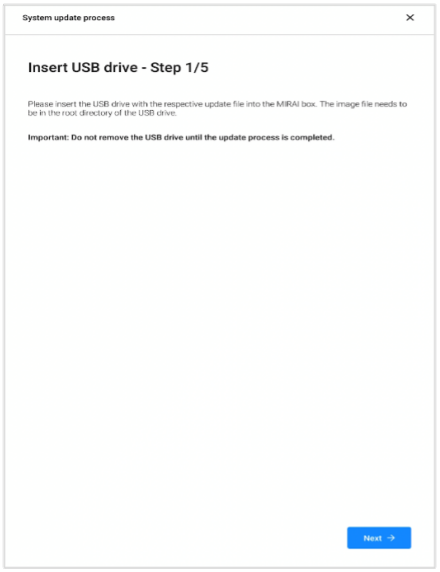
b. **Prepare the software update:**

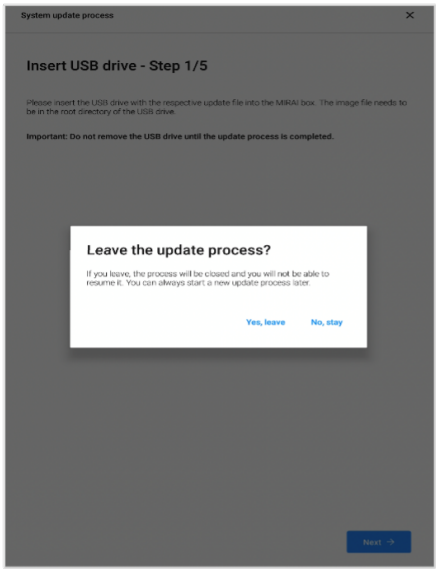
- Download the software file from the link provided by Micropsi Industries.
- Copy the software file to the USB drive, saving it in the root directory.

Step 2: Back up your skills

a. **Enter the update loop:**

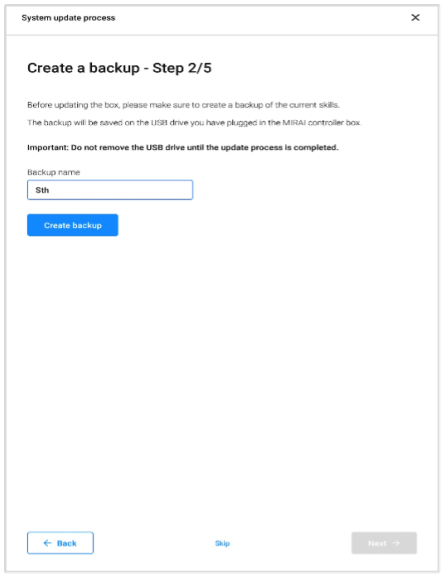
- Turn on the MIRAI controller and wait for the second beep, signaling that the system is ready.
- In the main menu of the MIRAI Training App, tap **Service Support**.
- Under the heading "Software update," tap **Update**.
- When prompted, insert the USB drive with the update file into the USB port of the MIRAI controller.
- Tap **Next** to proceed to the next screen.
  NOTE: To leave the update loop, tap the **x** button in the upper-right corner. When prompted, tap **Yes, leave**.

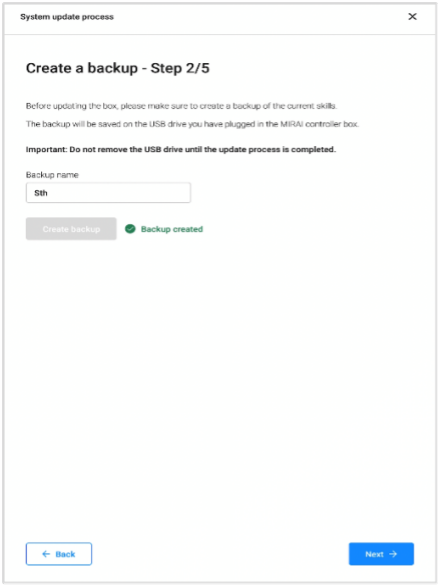




b. Create a backup file

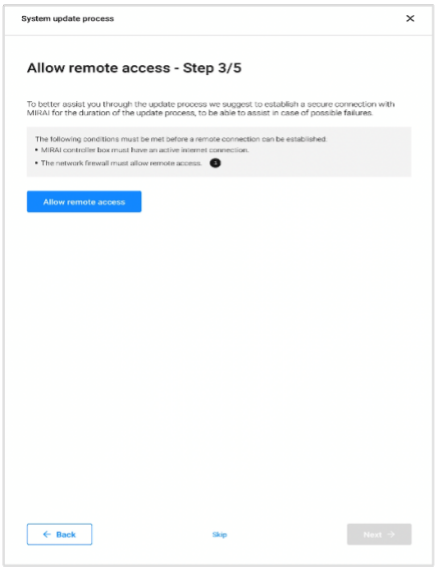
- Enter a name for your skill backup file and tap **Create backup**.
  A green "Backup created" message will appear and the **Next** button will be activated.
- Tap **Next** to proceed.

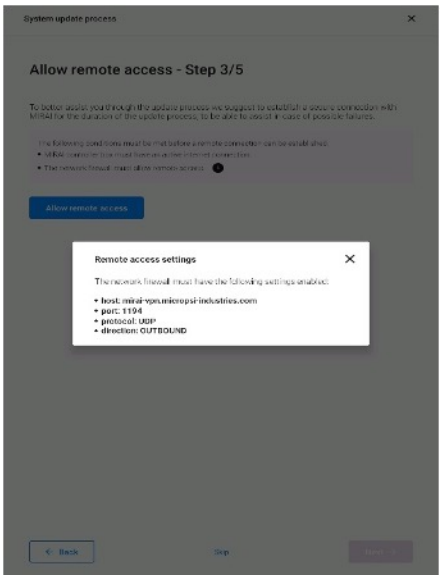




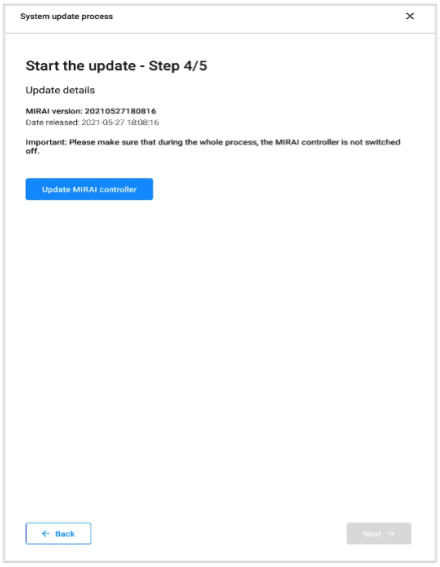
Step 3: Update the MIRAI controller software

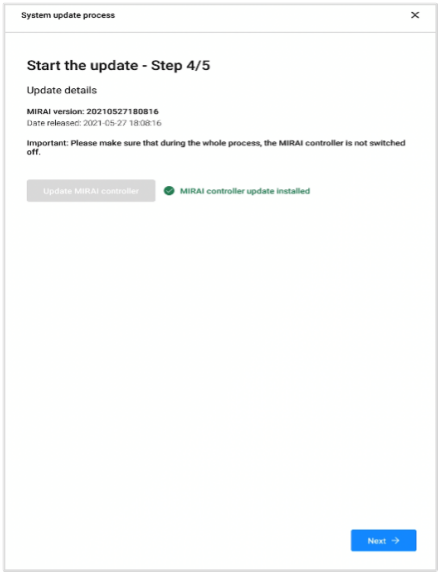
a. To facilitate support, tap **Allow remote access** to grant access to:
mirai-vpn.micropsi-industries.com (UDP Port 1194).





b. On the next screen, tap **Update MIRAI controller**. After installation, a green message will appear saying "MIRAI controller update installed" and the **Next** button will be activated.

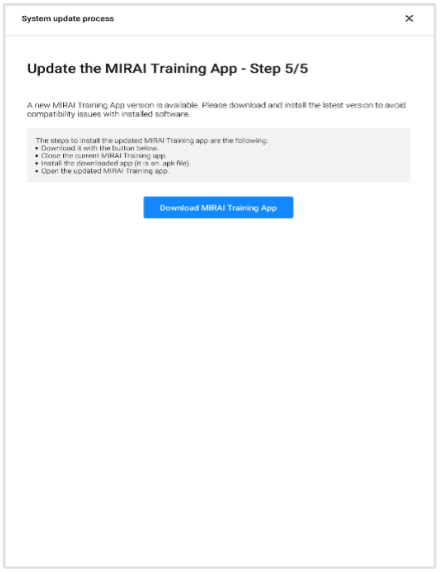




Step 4: Download the latest version of the MIRAI Training App

a. Tap **Download MIRAI Training App** to save the current version of the MIRAI Training App to your tablet.

b. Proceed to the next section, 15.4 Updating the MIRAI Training App.



Note

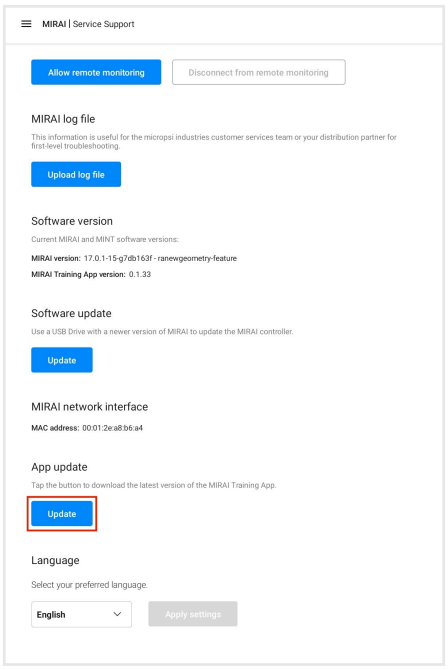
If you need to update the MIRAI software package for your robot platform, an additional link will be included in the email with the controller update.

15.4 Updating the MIRAI Training App

Important

Uninstall the current MIRAI Training App on your tablet *before* reinstalling the latest version from the MIRAI controller.

Download the app:
If you need to download the latest version of the MIRAI Training App, go to the app's main menu and tap **Service Support**. Under the heading "App Update," tap **Update**. This will start the internet browser on the tablet and open http://mirai:6543/mint/apk. Tap to confirm downloading the MIRAI Training App installation file, mint.apk. After downloading, tap the file in the Android file browser to install the app. If a security message requests permission to install the APK file, grant the request.



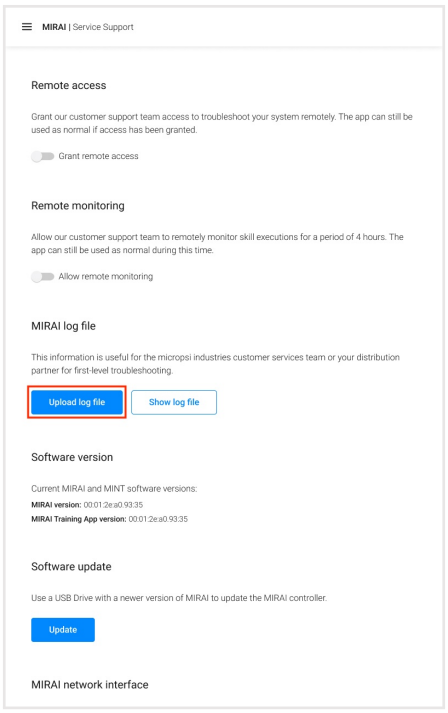
Uninstall the current version of the app:
Refer to the instructions for your tablet. Apps on Android-based tablets can generally be uninstalled by tapping and holding the app until a menu with ”Uninstall” appears.

Install the latest version of the app:
Locate and tap the APK file you previously downloaded to install it. Then open the new version of the app.

15.5 Log Files

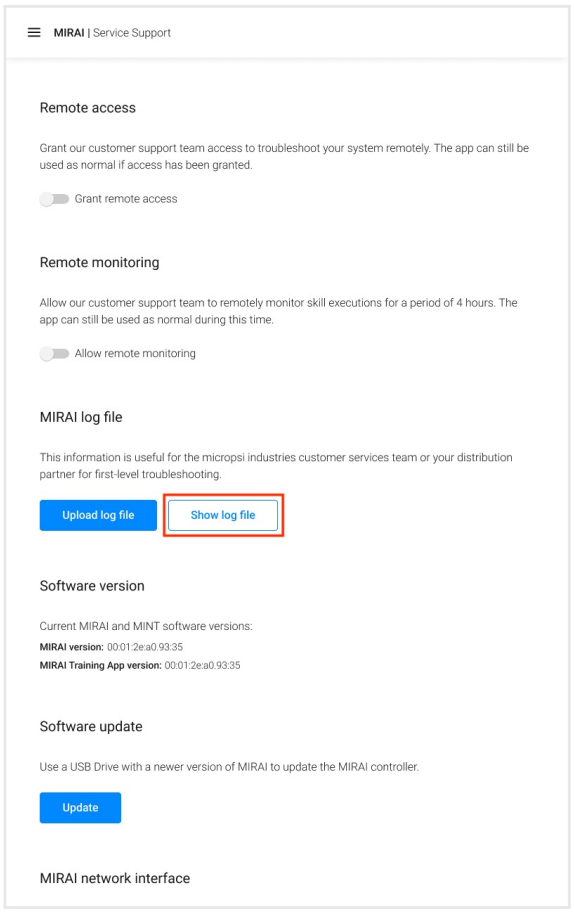
Uploading Log Files

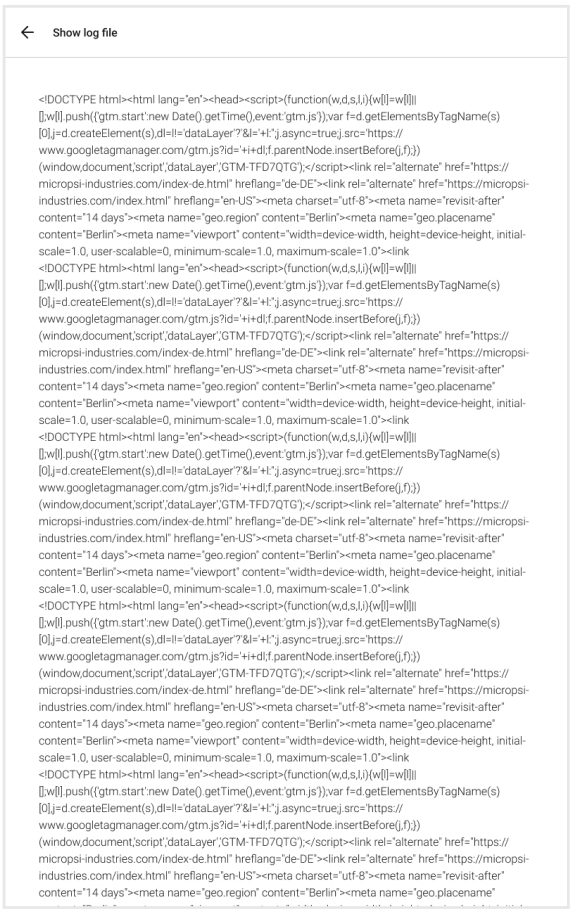
The log files are required in case of first level troubleshooting. Tap the button if this is requested by your distribution partners or the Micropsi Industries customer support team. The file will be uploaded onto the secure premises of the Micropsi cloud server.



Show Log Files

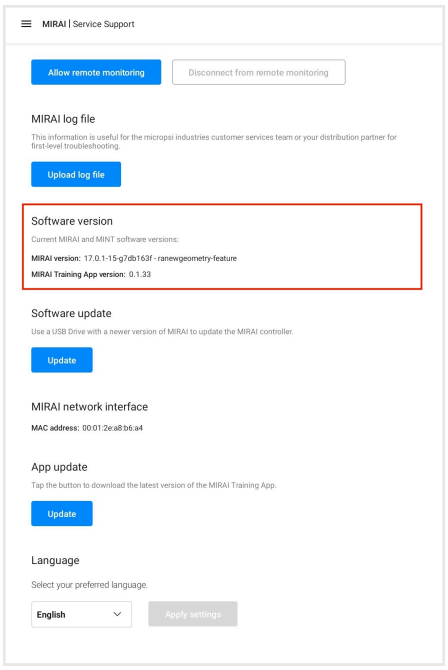
Tap **Show log file** to show the last 1,000 lines of the most recent log file.





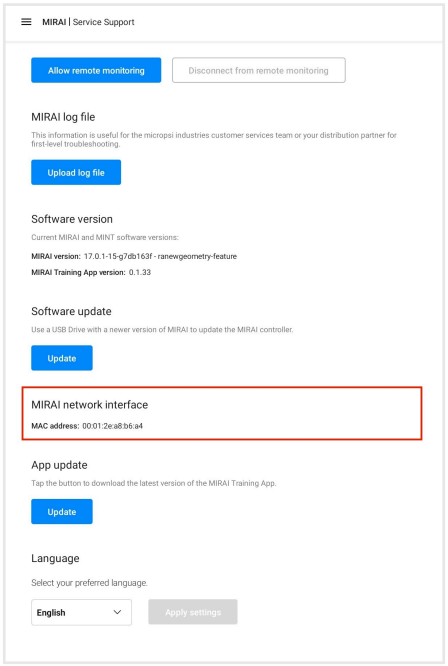
15.6 Software Version

The version number of the MIRAI software and the training app are listed under Software Version. This information can be relevant in case of software updates and when reaching out to Micropsi Industries customer support.



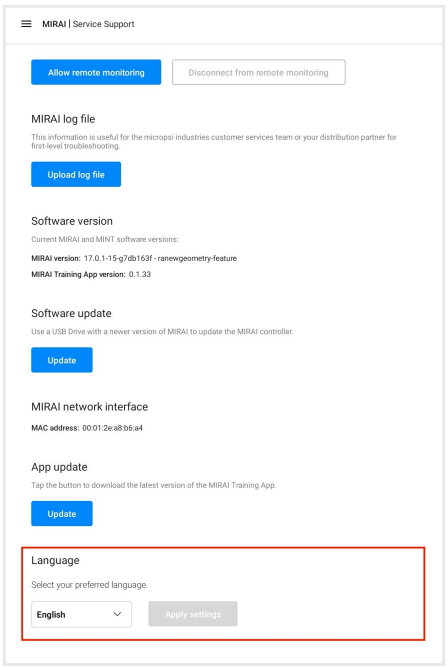
15.7 MAC Address

MIRAI's MAC Address can be found under MIRAI Network Interface.



15.8 Language Settings

Tap the language drop-down menu to choose your preferred language setting. You can choose between English and German.


MIRAI | Service Support
Allow remote monitoring
Disconnect from remote monitoring
MIRAI log file
This information is useful for the micropsi industries customer services team or your distribution partner for first-level troubleshooting.
Upload log file
Software version
Current MIRAI and MINT software versions:
MIRAI version: 17.0.1-15-g7db163f - ranewgeometry-feature
MIRAI Training App version: 0.1.33
Software update
Use a USB Drive with a newer version of MIRAI to update the MIRAI controller.
Update
MIRAI network interface
MAC address: 00:01:2e:a8:b6:a4
App update
Tap the button to download the latest version of the MIRAI Training App.
Update
Language
Select your preferred language.
English
Apply settings

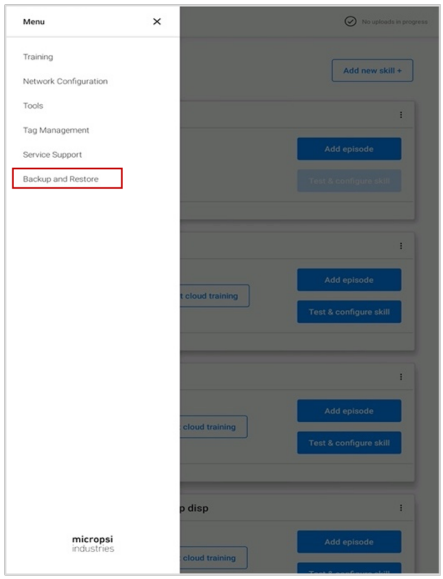
16 Backup and Restore

16.1 Creating a Backup of MIRAI Skills

Note

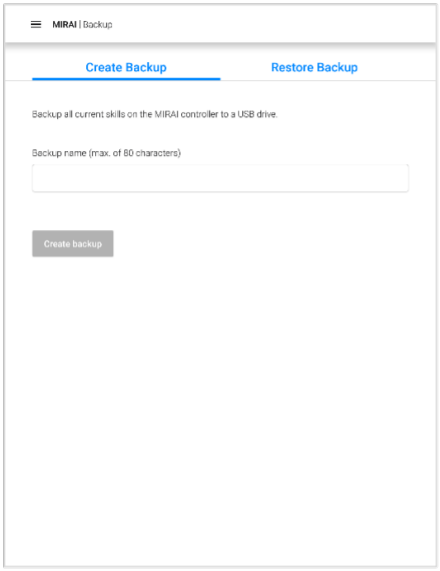
Please use a USB drive and format it with FAT32 (16GB and larger).

1. To create a backup of your existing MIRAI skills, on the main menu (top left corner) select **Backup**.



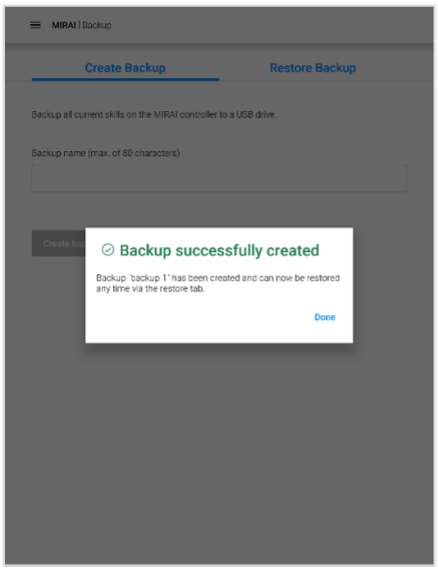
2. Insert a USB stick (minimum 16GB) into one of the USB ports located on the MIRAI controller.

3. Enter a name for the backup file and then select **Create backup**.



4. This process will take a few minutes. Do not switch off the MIRAI controller or the tablet during this process. Also refrain from removing the USB stick as long as the backup is ongoing. Once the backup

is successfully created, the message below will be displayed. Tap **Done**.



16.2 Restoring MIRAI Skills into the MIRAI Controller

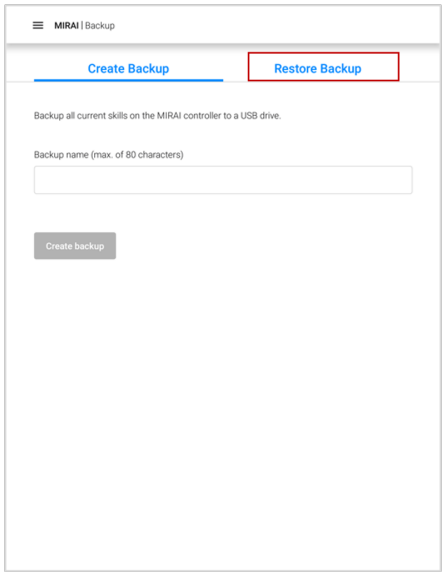
This feature allows you to restore the saved skills back to on the MIRAI controller.

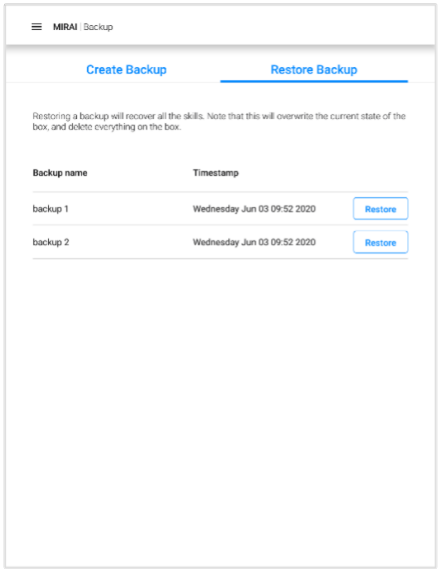
Important

This process **will replace the current skills** with the state of the backup and with that delete all changes done since the last backup. Please make sure to back up your skills again after a MIRAI software update.

1. To restore a backup of skills, insert a USB stick into one of the USB ports located on the MIRAI controller. From the main menu, select **Backup**, and then select the **Restore Backup** tab.

2. From the displayed list of backups, tap **Restore** to restore the relevant backup.

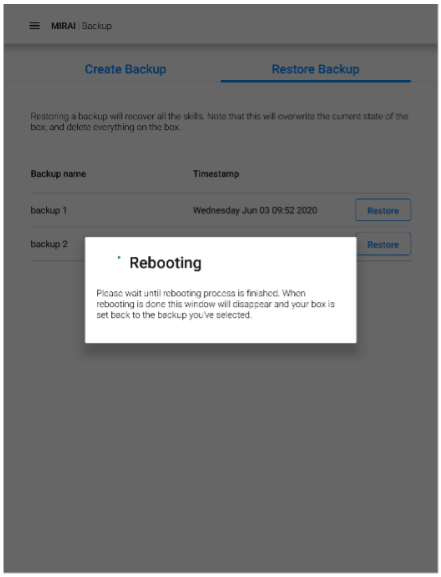


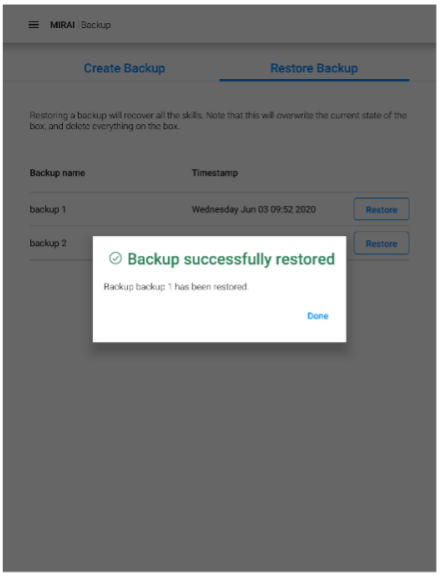


③ Once the process is complete, the MIRAI controller will reboot after which the restored skills will be seen on the main screen.

Note

Do not switch off the MIRAI controller or the tablet during this process. Also refrain from removing the USB stick until the restore is successfully completed.





17 Data FAQ

This section explains how Micropsi Industries handles MIRAI data.

What types of data does Micropsi Industries collect and store?

The MIRAI robot control software uses sensor data to enable robots to act in settings with high production variances. When a user creates a robot skill with MIRAI, the initial skill configuration entered into the MIRAI Training App is stored.

Further data is collected when a user starts recording episodes during the MIRAI training phase. The MIRAI system collects and stores the following information:

- camera image data (camera typically attached to the robot's wrist)
- robot motion data
- force/torque sensor signals
- tool center point (TCP) positions
- software version
- skill meta data (skill type, skill name, degrees of freedom, action frame)
- log files / diagnostic data

No data is collected and stored during skill execution / unsupervised production operation, with the exception of rotating logs - which means older logs are overwritten by newer logs.

Note that the videos provided typically do not show human workers or even complete wokspaces. These are very zoomed-in sequences taken by cameras typically moving with the robot, showing the robot's end-effector and/or an individual part.

Data is only ever stored on the MIRAI controller (indefinetly) and on the Micropsi cloud server (indefinity). The training app is a pure user interface to the MIRAI controller and does not persist any data except the IP address of the controller to talk to.

What does Micropsi Industries use that data for?

Micropsi Industries uses data described above to prepare and/or train skills – mathematical models that generate actions the robot executes. A skill connects the raw sensor information captured by the camera and the force/torque sensor to the actions performed by the robot.

Beyond creating skills, Micropsi Industries does back-testing on existing data when new features are developed. This involves generating data sets, mixing anonymized data from multiple sources, and other machine-learning research in order to further improve models and algorithms. Skills will not be shared with third parties.

Furthermore, we store log file/diagnostic data to be able to help users in case of support requests. In order to create an initial password and Controller ID, the MIRAI controller will be provisioned with the customer's company name. The resulting Controller ID is needed to be able to provide remote support.

Why is the data not deleted after skill creation?

Whenever skills are modified with additional data, i.e. to make a skill more robust against a rare type of variance that wasn't encountered in initial training, the new skill version will be trained on all data available for that skill. That means that there is a functional requirement to keep training data available for future use in the cloud application.

Additionally, Micropsi Industries reserves the right to use the data for the purposes defined in Section 2. Requirements to delete data after a fixed period of time or on request can be discussed as part of a commercial discussion between Micropsi Industries and customers.

Where is collected data stored and secured?

For data storage, Micropsi Industries uses AWS. The stored data is geographically located in the AWS eu-central-1 region, in Frankfurt, Germany. AWS uses triple redundancy, with no redundancy of data outside of eu-central-1. The data that MIRAI collects is S3 encrypted in transit and at rest (prior to January 5, 2023 HTTPS transport encrypted). A further encryption on storage level is not necessary, because read and decryption permissions would be in the same security realm. Whoever has reading access to the data, would also have access to the decryption key. Addendum All AWS users globally can rely on the terms of the AWS GDPR DPA which will apply automatically from May 25, 2018, whenever AWS services are used to process personal data under the GDPR. Please find the complete Data Processing (DPA) here: AWS Data Processing Addendum

Note however that MIRAI data as described above does not fall under the provisions of the GDPR, as it is not personal data as defined by the GDPR.

Who has access to data on the Micropsi cloud server?

There are three groups of Micropsi Industries employees that need access to user data in the context of their work. The three roles are developers, reviewers & admins.

- developers can list files (while not seeing the content) when debugging or updating the server-side cloud / data- processing application
- reviewers can read files incl. content in order to debug and optimize skill performance. This role is intended for support staff, mainly machine learning and application engineers
- admins can write & delete files incl. content in order to ensure overall functionality

Access will be granted for valid reasons and on individual level. Only admins can grant access and individual access keys can be revoked anytime. Permissions & access rights are documented and regularly reviewed.

No customer users can access the data stored on the Micropsi cloud server, it's used solely by the cloud application to create skills that will be downloaded to the MIRAI controller.

Note

Above mentioned access refers only to data on the Micropsi cloud server. Micropsi Industries does not have access to a MIRAI system in operational mode.

Why does MIRAI need a cloud?

The heavy lifting for creating a MIRAI skill from recorded data requires very powerful hardware. Delivering this hardware in the MIRAI controller would make the controller extremely expensive, especially considering that 99% of the time, MIRAI controllers do something else. It's just reasonable to share the resources for these brief bursts of computation between MIRAI installations.

In addition, access to the data recorded by MIRAI installations (see the other answers for what we do with the data) is important for Micropsi Industries' ability to deliver a superior product. We're a specialist for turning this data into value for our customers, and access to data enables us to do this.

Why is the cloud not on Micropsi Industries' own hardware?

There are no advantages of operating our own data centers, or operating our own server hardware in a data center that is not AWS'es. AWS has excellent service levels, superior security to anything we could provide at comparable cost, and enough capacity reserves to allow us to service the compute needs of our MIRAI installations even if they all decide to create a skill at the same time.

Renting compute capacity and services around security and availability while retaining full control of the data used for the computation is the de-facto default model for all sensibly run IT today.

Can I operate MIRAI on my own corporate cloud?

In general, no. Micropsi Industries would consider such a scenario only for deployments of >100 MIRAI units. The Micropsi Industries server-side infrastructure is a complex application, requiring multiple specialized containers to run, as well as GPU-enabled resources ready to spin up on request. Operating, updating and maintaining such an application in a shared responsibility with a corporate IT unit introduces several severe friction lines and requires large operational efforts on Micropsi Industries's side. Engaging in any form of shared-responsibility application hosting (containerized or not) requires a serious-long-term commitment of both parties. Usually, when such long-term commitments are commercially and strategically an option at all, the matter of hardware operation and security/availability service outsourcing is not a relevant factor and a commercially equivalent contractual alternative can be found to moving applications around physically.

What networks are included in the standard MIRAI setup?

Most customers implement a variant of this network architecture:

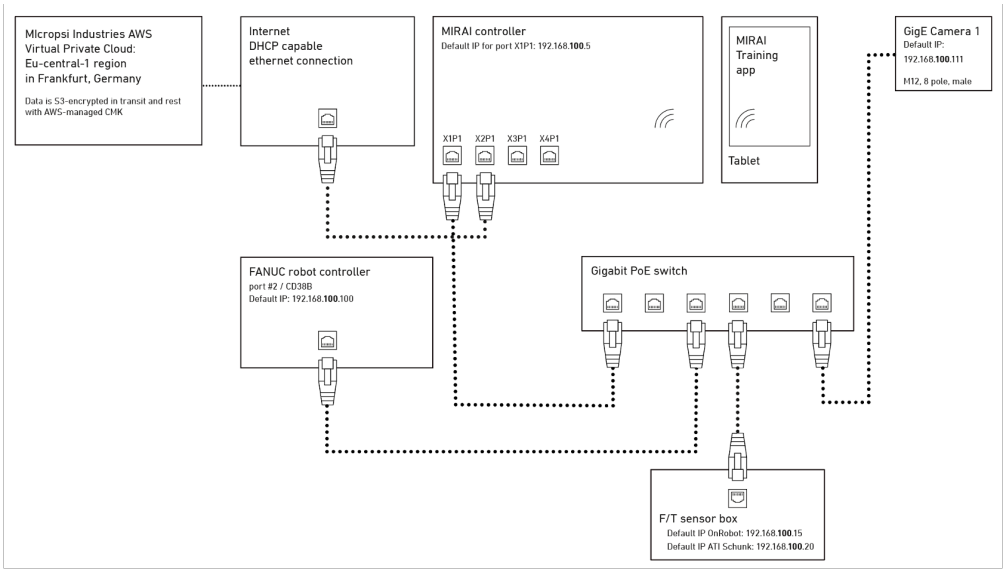


Figure 55: GigE single camera setup

There are three networks in the standard MIRAI setup: the robot network, the WAN, and the Wi-Fi network. The MIRAI controller uses different ports for each network, ensuring they remain separated and cannot communicate with each other.

1. **Robot Network:** This network includes the MIRAI controller, the robot controller, a force/torque sensor (if used), and the GigE camera or cameras (if used). Each device in this network has a static IP address. The devices are connected through specific ports on the controller, ensuring they operate within an isolated network that cannot communicate with the other networks.

2. **WAN Network:** This network also includes the controller, but it is connected through a different port preconfigured to expect a DHCP-assigned IP address. The WAN typically includes an HTTP/HTTPS proxy that facilitates communication between the controller and the Micropsi cloud, making it reachable via the internet. This network is logically separated from the robot network.
3. **Wi-Fi Network:**This network is created by the controller and includes a single member, an Android-based tablet. The Wi-Fi network is isolated from both the robot network and the WAN, ensuring that the tablet can only communicate with the controller through this dedicated wireless connection.

Can the connection to the Micropsi cloud server go through an HTTP proxy?

Yes. A proxy for the connection to the Micropsi cloud server can be configured via the tablet app's "Network configuration" section. The proxy needs to allow connections to the following hosts:

> *For training:*
> https://crunch.micropsi-industries.com (TCP Port 443)
> https://crunch.micropsi.io (TCP Port 443)
> *For support:*
> mirai-vpn.micropsi-industries.com (UDP Port 1194)
> *For software updates:*
> https://apt.tools.micropsi.io (TCP Port 443)
> http://deb.debian.org (TCP Port 80)
> http://security.debian.org (TCP Port 80)

Does Micropsi Industries require any form of remote access into their customer's networks?

No. There is a VPN-based remote-access facility available, but its use is not mandatory. It is disabled by default, defaults to disabled on boot, and should only ever be turned on during a remote support scenario and discussed with Micropsi Industries support. Customers that prefer not to allow VPN connections into their networks for remote service of an edge device can provide alternative ways of giving remote access, i.e. SSH sessions from Citrix terminals.

How is access control handled for the Micropsi cloud server, MIRAI controller, and training app?

The cloud application has no user-facing interfaces. Its only interfaces are machine-to-machine APIs that are being used by the MIRAI controller.

MIRAI controllers authenticate against the cloud services with their ID and a secret token generated at provisioning time. Customers cannot access these tokens (seeing them requires SSH access to the MIRAI controller, which customers don't typically need/have).

The training app and the MIRAI controller's APIs that it uses are assumed to be in a safe network (i.e. part of a machine and connected to a robot) and do not require authentication.

18 Declaration of Incorporation

MICROPSI INDUSTRIES

EU Declaration of Incorporation *(in accordance with Machinery Directive 2006/42/EC Annex II, part B)*

Manufacturer:	Person Authorized to Compile the Technical File:
micropsi industries GmbH Möckernstrasse 120, 10963 Berlin, GERMANY	Naaimah Saghir VP Product micropsi industries
Description and Identification of the Partly Completed Machine(s)	
Product and Function:	Vision-based motion control system for industrial robot systems that enables such robot systems to solve automation problems with high variance in position, shape, or background and lighting conditions. The final function is determined by the completed machine (i.e., robot system, robot cell or robot application with intended use).
Model:	MIRAI Software version 14.0.0 onwards
Incorporation:	The MIRAI vision-based motion control system shall only be put into operation upon being integrated into a final completed machine, which conforms with the provisions of the Machinery Directive and other applicable directives.
It is declared that the above product, for what is supplied, fulfils the directives as detailed below: When this partly completed machinery is integrated and becomes a final machinery, the integrator is responsible for determining that the final machinery fulfils all applicable Directives and providing the Declaration of Conformity.	
(I) Machinery Directive 2006/42/EC	The following essential health and safety requirements were fulfilled: Annex IV, clause 1.1.2, 1.1.3, 1.1.5, 1.2.2, 1.2.3, 1.5.1, 1.5.2, 1.7.1, 1.7.2, 1.7.4 It is declared that the relevant technical documentation was compiled in accordance with Annex VII, Part B The following Harmonized Standard were used (where applicable): EN 60204-1:2019
(II) Low-Voltage Directive 2014/35/EU	The following Harmonized Standard were used (where applicable): EN IEC 60320-1:2023
(III) Radio Equipment Directive 2014/53/EU	The following standards were used: Article 3.1a): EN60950-1:2006+A11:2009+A1:2010 +A12:2011+A2:2013; EN 50663: 2017 Article 3.1b): Draft EN 301 489-1 V2.2.0; final draft EN 301 489-3 V2.1.1; Draft EN 301 489-17 V3.2.0 Article 3.2): EN 300 328 V2.1.1; ETSI EN 301 893 V2.1.1; ETSI EN 300 440 V2.2.1
(IV) Other directives	All applicable directives and harmonized standards have been followed by the component suppliers and can be provided upon request.
Reference to Other Technical Standards and Specifications Used:	
EN ISO 10218-2:2011, ISO/TS 15066:2016	
The manufacturer, or his authorized representative, shall transmit relevant information about the partly completed machinery in response to a reasoned request by the national authorities.	

Berlin, Germany, 15 February 2024

Ronnie Vuine, Chief Product Officer

https://www.micropsi-industries.com/
micropsi industries GmbH Möckernstrasse 120, 10963 Berlin, Germany +49 30 555 71 929
Micropsi Industries USA, Inc, 300 Brannan St. Suite #101, San Francisco, CA 94107, USA +1 718 440 7353