Documentation
mikado Find 1.3

Manual Version 1.19, 25.05.2023

Optonic GmbH | Zollhallenstraße 11 | 79106 Freiburg | Germany/Germany
Phone: +49 761-15436-0 | E-mail: info@optonic.com
© Copyright Optonic GmbH , 2023

1 DOCUMENT HISTORY

Date	Vers.	Abbreviation	Comment
10.03.21	1.0	ROP	Initial creation
09.04.21	1.1	ROP	Setting the reference point from the 3D object
03.05.21	1.2	ROP	New paths, software requirements and bundle installers
14.05.21	1.3	ROP	Adjust Axes Menu, Installing the Application
08.06.21	1.4	ROP	Hash, debug mode and log files
22.09.21	1.5	ROP	Add robot calibration, Remove reference point from 3D object
11.01.22	1.6	ROP	2D Refinement, JSON and advanced parameters

03.02.22	1.7	ROP	Mikado Find 1.1
			
			• New PartFinder 3.3:
			
			  o generate new parameters in model, number of templates, model information
			
			  o new parameters for the search, new combined 2D/3D refinement method
			
			• New ROI: 3D manipulation (shift, size, rotation) and 3D display of the ROI, hiding of all points outside the ROI
			
			• Calibration: display of the right camera image for easier positioning of the pattern
			
			• Protocols: Time accuracy in the millisecond range

25.05.22	1.8	ROP	ModelID in Model Info, Updated Licenses and Versions, OPC-UA Server Update Rate
31.05.22	1.9	ROP	OPC-UA Server Rate as parameter, command opcms
20.06.22	1.10	ROP	New OPC-UA interface with static array
05.07.22	1.11	ROP	Add loading/saving of calibration files, TCP/IP Ok bit clarification, new error code
08.07.22	1.12	ROP	Add loading/saving of parameters. Load parameters and calibration also from within a configuration file.
14.08.22	1.13	ROP	Selecting and opening .ply model files
22.08.22	1.14	ROP	Add three new presets for rotation modes
28.08.22	1.15	ROP	Delete files that appear in the lists
01.09.22	1.16	ROP	Add expert level detection parameters

Date	Vers.	Abbreviation	Comment
06.12.22	1.17	ROP	Mikado Find 1.2 • PartFinder and Ensenso SDK 3.4 • Automatic deletion of the log files • new TCP/IP, OPC UA and console commands • Improved camera setting • Improved control and display of ROI • Workspace menu added • Redesigned menus for calibration, search parameters and models • Hand-eye calibration with tip • Loading the calibration from the EEProm • Parameterizable axis alignment • Definition of the gripping point • Sort results • New rotation modes • Multiple instances • --name Start command
17.01.23	1.18	ROP	Search parameters: Add third option for search refinement
25.05.23	1.19	ROP	Mikado Find 1.3 • PartFinder and Ensenso SDK 3.5 • Compatibility with Ensenso SDK 3.5 or higher • Compatibility with the new Ensenso C camera series Application path update Server IP: localhost:8080, Default TCP Port: 20000

2   CONTENT

3   GENERAL

Mikado Find is an industry-standard 3D object recognition software that runs on Microsoft Windows 10. It requires Ensenso SDK 3.5 or higher.

Mikado Find is compatible with the full range of stereo cameras available via USB or GigE ports with Ensenso SDK 3.5.

It uses the "partFinder" algorithm implemented in SDK 3.5 to quickly locate objects defined by a provided 3D model. It returns the 3D positions and orientations of the detected objects in calibrated world coordinates.

The application communicates via an OPC-UA connection. It includes an OPC-UA server that receives commands and transmits the results to an OPC-UA client. The application can also exchange commands and results via a direct TCP/IP connection.

The application runs in console mode on the server and provides a web GUI for remote control. The web GUI modern touch screen interface is designed with the operator in mind and can be customized to meet customer requirements.

4   SYSTEM REQUIREMENTS

4.1   OPERATING SYSTEM AND SOFTWARE

Mikado Find runs on Microsoft Windows 10 operating system, 64 bits.

Ensenso SDK 3.5 or higher is required.

Please visit https://www.ensenso.com/support/sdk-download/ to download the latest version or install it using the Mikado Find Bundle installer.

Mikado Find has the same requirements as Ensenso SDK 3.5, but also requires the ability to use OpenGL and a real camera, so the minimum requirements are more restrictive than Ensenso SDK. Mikado Find only runs on Windows 10.

To use the camera, you also need the uEye driver from IDS, version 4.9 or higher, which is also included in the bundle installer.

Requirements	Minimum:	Recommended:
CPU	2.0 GHz	4 x 3.0 GHz with support for AVX2 instruction set (1)
RAM	8 GB	16 GB in dual-channel or quad-channel configuration
Operating system	Windows 10 64bit	Windows 10 64bit
Graphics	OpenGL 4.0 compatible graphics card and driver	OpenGL 4.0 compatible graphics card and driver
For GigE cameras	GigE port	A dedicated GigE port for each camera on a PCIe adapter based on an Intel chipset
For USB cameras	USB2.0	A motherboard chipset that provides a separate USB2.0 root hub for each camera

For Intel CPUs, you can check the instruction set and the actual number of CPU cores of your model at http://ark.intel.com.

Required software	Recommended:
Ensenso SDK 3.5	https://www.ensenso.com/support/sdk-download/
uEye driver > 4.9	https://en.ids-imaging.com/downloads.html
Codemeter User Runtime For Windows	https://www.wibu.com/support/user/user-software.html
Microsoft Visual C++ Redistributable	https://learn.microsoft.com/en-us/cpp/windows/latest-supported-vc-redist?view=msvc-170

These required software are included in the bundle installer.

4.2   SETTING THE NETWORK CARD PROPERTIES

Check that the network card's receive buffer and MTU settings are as high as possible.

In Device Manager, open the properties dialog of your network card. Select the Advanced tab and try to find the properties for the receive buffers and set them as high as possible. Then find an entry for the MTU (Maximum Transmission Unit) (sometimes Jumbo Frames) and set it as large as possible.

Receive buffer settings

MTU settings (Jumbo Frames)
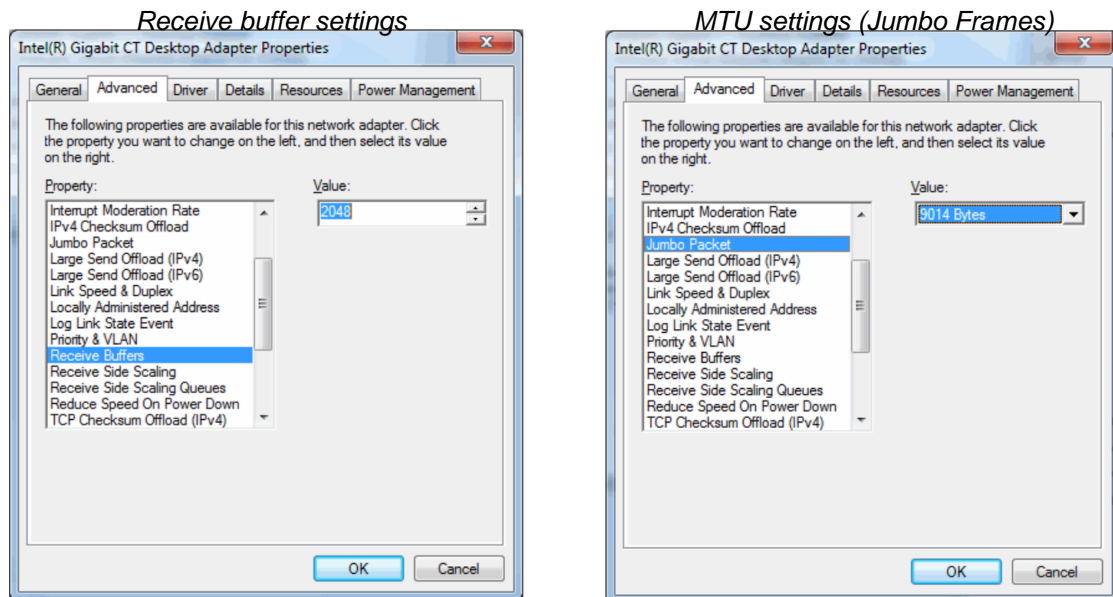
4.3   SOFTWARE PROTECTION

The Mikado Find application and the PartFinder algorithm are protected by a Codemeter license and require a special USB dongle.

The Codemeter User Runtime is also required. You can download it from https://www.wibu.com/support/user/user-software.html or install it via the Mikado Find bundle installer.

The Mikado Find software protection dongle

4.4 STEREO CAMERA

Any stereo camera compatible with ENSENSO SDK 3.5 can be used with Mikado Find connected to USB or GigE Ethernet ports. This includes Ensenso N, S, X, XR and C series cameras.

4.5 OPC-UA COMMUNICATION

The application communicates via an OPC-UA connection. The server "End-Point" is "Mikado Find OPC-UA Server" at port **4840**. This can be changed with the console commands and saved in the configuration. The interface is defined in the appendix (see below. 11.3 OPC-UA interface).

4.6 HTTP COMMUNICATION

The web GUI runs on port **8080 on the** server by default. On the server it can be found at the URL **localhost:8080,** while you have to specify the server's IP in the URL to access it from a remote computer. The port can be changed using the console commands and saved as part of the configuration.

4.7 TCP/IP COMMUNICATION

The application can also exchange commands and results via a direct TCP/IP connection (see below. 12 TCP/IP interface). The default TCP port is **20000**.

4.8 MULTIPLE INSTANCES

It is possible to run multiple instances of the application if all instances are started with different instance names (see below. 6.8 --name). Each instance must be started with different ports, from the command line or via desktop shortcuts.
For example:

```
"C:\Program Files\Optonic GmbH\Mikado_Find\Mikado_Find-
130\Bin\Backend\Mikado_Find.exe" --name instance1 --web
8080 --tcp 20000 --opc 4840
```

```
"C:\Program Files\Optonic GmbH\Mikado_Find\Mikado_Find-
130\Bin\Backend\Mikado_Find.exe" --name instance2 --web
8081 --tcp 20001 --opc 4841
```

In this case, multiple codemeter licenses for Mikado Find and PartFinder are required. Please also note that a camera can only be used by one instance at a time.

5 APPLICATION INSTALLATION

The installer is available in two versions:

- a single installer that installs only Mikado Find

- a bundle installer that also packs the following third-party
  dependencies:

  o Ensendo SDK 3.5
  o uEye driver
  o Codemeter User Runtime
  o Microsoft Visual C++ Redistributable

Since this application is not sold in the Microsoft Store, Microsoft Defender
SmartScreen can mark it as undetected.
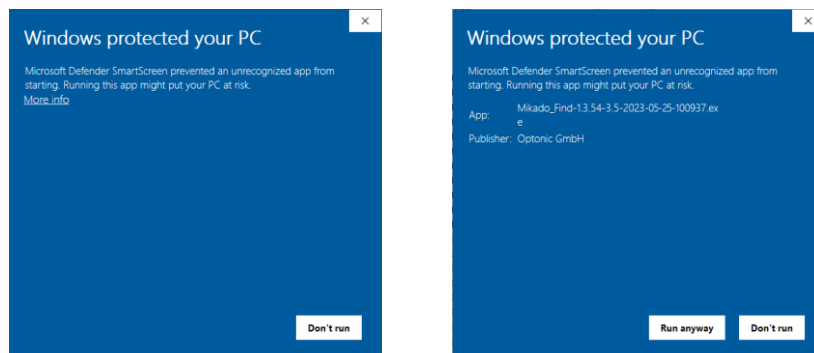
First and second page of Microsoft Defender SmartScreen

In this case, please click on "More Info" (or the local language equivalent), verify
that the publisher is Optonic GmbH, and then click on "Run anyway".

For standalone installation you can specify where the application should be
installed, by default in C:\Program Files\Optonic GmbH.
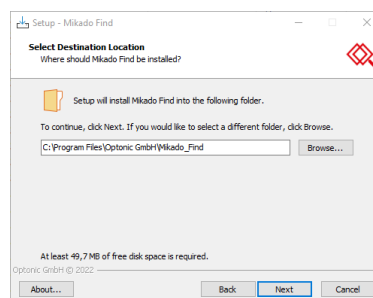
The selection of the installation location in the single installer

The bundle installer provides a selection of components to be installed in their
own location.
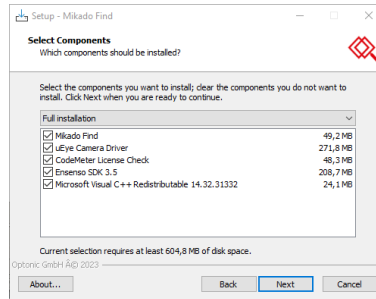
The component selection in the bundle installer

It is possible to set the ports used by Mikado Find: the TCP port for the HTTP web GUI (see below. 4.6 HTTP communication), the TCP port for OPC-UA communication (see below. 4.5 OPC-UA communication) and the TCP port for direct TCP/IP communication (see below).4.7 TCP/IP communication).
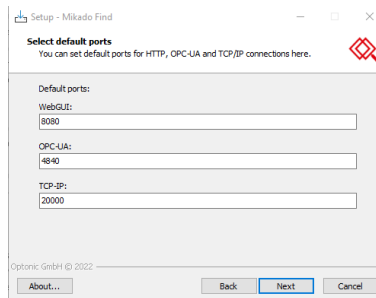
On this page you can change the default ports

Run the installer to the end to successfully install Mikado Find and its dependencies.

6   STARTING THE APPLICATION

You can launch Mikado Find from the desktop shortcut or from your application folder by clicking Mikado_Find.exe.

The application folder is

```
C:\Program Files\Optonic GmbH\Mikado_Find\Mikado_Find-
130\Bin\Backend
```

When starting the application for the first time, the firewall settings must be set so that Mikado Find can communicate on the selected HTTP, OPC-UA and TCP/IP ports.
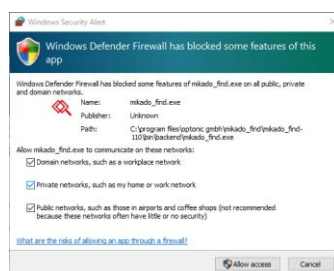
Windows Defender Firewall settings when Mikado Find is run for the first time

You can also start Mikado Find in a Windows command shell. It is possible to specify parameters at startup. This also works when creating shortcuts to Mikado Find.

6.1 --WEB

One can specify the http port with the --web command. For example:

```
.\Mikado_Find.exe --web 8081
```

6.2 --TCP

One can specify the http port with the --web command. For example:

```
.\Mikado_Find.exe --tcp 20001
```

6.3 --OPC

One can specify the http port with the --web command. For example:

```
.\Mikado_Find.exe --opc 4842
```

6.4 --NO-WEB

Use this command to disable the http server and the user interface. For example:

```
.\Mikado_Find.exe --no-web
```

6.5 --NO-TCP

Use this command to disable the TCP/IP server. For example:

```
.\Mikado_Find.exe --no-tcp
```

6.6 --NO-OPC

Use this command to disable the OPC UA server. For example:

```
.\Mikado_Find.exe --no-opc
```

6.7 --CONFIG

This gives the possibility to load a configuration directly at startup by adding the configuration name after the exe name, For example:

```
.\Mikado_Find.exe --config config1
```

6.8 --NAME

This option gives a name to the current application instance. This allows multiple instances to run with different configuration folders and log files, For example:

```
.\Mikado_Find.exe --name instance1
```

7 DATA DIRECTORY

The applications store their parameters in the folder

```
C:\Users\(User)\AppData\Local\Optonic GmbH\
```
Mikado_Find_Mikado_Find-130\Data

and the error logs in the `\Logs`.

File	Task
Data\xxx.zip	Saved configuration files as .zip
Data__currentConfig[_name of instance]	Current unpacked configuration. [Name of the instance], see below. 6.8 --name.
Data__calibrations	Calibrations are stored in this folder. Only calibration files from this folder can be loaded via the Load file button of the calibration menu.
Data__parameters	The search and model parameters can be saved and loaded individually. The files are saved in this folder.
Logs\Report_[instance name]_xxxxxx.xxx.zip	Log file as zip archive. [Name of the instance], see below. 6.8 --name.

7.1 CONFIGURATION FILE

By default in `C:\Users\(User)\AppData\Local\Optonic GmbH\ Mikado_Find\Mikado_Find-130\Data`

The configuration files are stored as a zip archive. They contain the files described in the following table.

File	Task
model.json	Saved configuration
parameters.json	Log file as zip archive.
<serial>.json	If a camera <serial> is open, the camera parameter file
model.dat	The data of the search model
<name>.stl or <name>.ply	The STL or PLY 3D model file whose name is <name>.

7.2 LOG FILE

By default in `C:\Users\(User)\AppData\Local\Optonic GmbH\Mikado_Find\Mikado_Find-130\Logs` are stored the log files.

- *Report_nnn_xxx.zip* (with the instance name as nnn and the date as xxx timestamp.)

This "Report" zip archive contains the error messages, the actual images and the camera calibration.

File	Task
xxx_nn_left/right.png	Saved camera images
xxx_calib.json	When a camera is open, the camera parameter file
xxx_messages.log	The log messages
xxx_cameras.log	The list of available cameras
xxx_servers.log	The status of the OPC and Web servers

In addition, the following log files are available for the last application run:
- *Mikado_Find_nnn_xxx.log* : Backend debug logs
- *WebServer_nnn_xxx.log* : Web server error logs
- *Mikado_Find_nnn_xxx.nxlog* : NxLib error logs
  (with the instance name as nnn and the date as xxx timestamp).

The NxLib error log is automatically created in case of an NxLib error when the debug level of Mikado Find is above 0.

The log files older than 30 days are automatically deleted when the application is started.

8   FUNCTIONS

The main function of the application is the localization of 3D objects. Before the first object search, the application must be set up.

The preparation steps are:

1) Choose a stereo camera
2) Set the reference point in the 3D coordinate system
3) Select a 3D model file that describes the object you are looking for.
4) Optionally narrow the search to a specific region of interest (ROI).
5) Generate a search model

When all steps are done, or loaded, the search is ready. To perform these steps, Mikado Find offers the following functionalities:

8.1   CAMERA FUNCTIONS

- Open camera

Mikado Find can recognize and open all stereo cameras supported by Ensenso NxView software.

If the camera needs to be set up, a console command opens NxView for this purpose. If the camera's network connectivity needs to be corrected, NxView's Network Wizard is available via the console or the web GUI.

- Camera image

Snap, live image: Mikado Find also offers image capture and live mode.

8.2   CALIBRATION FUNCTION

A reference point for locating the results can be taught in.

There are two options, a single position calibration with a calibration pattern or a hand-eye calibration to define a match between the camera position and a robot coordinate system.

8.3   MODEL FUNCTIONS

- 3D Model Shop

Mikado Find can load 3D models saved in STL or PLY format.

- Generate search model

An internal model is created to represent a 3D model to be searched in a specific 3D area in the field of view of a stereo camera. This model can be parameterized, optimized with symmetric properties of the model and stored for reuse.

8.4 SEARCH FUNCTION

The search model will then be used to detect and accurately locate objects. The results include the location (x, y, z), rotation and quality of detection.

8.5 STORAGE AND LOADING FUNCTIONS

Save and load: All parameters can be saved in a configuration file. This includes the camera parameters, the model parameters, the search and the region of interest.

It is also possible to load and save individual files for the search parameters, model parameters and calibration.

These components can also be loaded selectively from a configuration file.

8.6 KONTROL FUNCTIONS

The functions can be accessed via three different user interfaces: via the console, the web GUI and via the OPC UA Server.

9 CONSOLE

The Mikado Find console on the server

Mikado Find is started in console mode on the server.

The console displays basic information about the server, such as the name of the current configuration, the version of the Ensenso SDK, the ports of the OPC UA server, the TCP/IP server and the web server.

The status of the OPC UA server, the TCP/IP server, the web server, the camera, and the update rate of the OPC UA server is also displayed.

The value of 3D model, reference point, rotation mode and the current debug level is displayed on the right side.

From this console, the administrator can enter various commands related to the setup, but also perform 3D searches.

A prompt (>>) waits for the command. The list of available commands is displayed by entering "help" and pressing the Return key.

9.1 CONFIGURATION

Command	Parameter	Description
load	Name	Load parameters from named configuration file. Example: load project1
save	Name	Save parameters from the named configuration file. Example: save project1
list config		list available configuration file. Example: list config
web	Port	Restart of the web server on port (number). Example: web 8080
opc	Port	Restart the OPC-UA Server on port (number). Example: opc 4841
opcms	Time	Set the update rate of the server, in ms. Example: opcms 50
tcp	Port	Restart the TCP/IP server on port (number). Example: tcp 20001
list ip		List of all ipv4 addresses of the server
list ip6		List of all ipv6 addresses of the server
logs		Show 10 last log messages.
report		Generates a bug report as a zip archive.
help		Display the list of available commands.
debug	Debug level	Set debug level of the web GUI.
hash		The hash to display.

9.2 CAMERA

Command	Parameter	Description
list cam		list available cameras. Example: list cam
open	Serial	Open camera <serial>. Example: open camera17234
open	Index	Opens the camera at the index in the list. Example: open 1
calibrate		Set the reference point with a calibration pattern
net		Opens the Network Wizard. This can be used, for example, to fix the connectivity of unrecognized cameras.
setup	Serial	Opens the camera <serial> in NxView to set it up. After closing NxView, when the camera is opened (with the 'open' command), its parameters are automatically loaded as they are set in NxView. Example: setup camera17234
setup	Index	Opens the camera (at the index in the list) in NxView to set it up. After closing NxView, when the camera is opened (with the 'open' command), its parameters are automatically loaded as they are set in NxView. Example: setup 1

9.3   MODEL

Command	Parameter	Description
List models		List available STL or PLY model file
model	Name	Select 3D model file. Example: model SimpleCylinder Currently, only non-accented characters are allowed in the name.
model	Index	Select 3D model at the index in the list. Example: model 1
generate		Generate search model

9.4   SEARCH

| Command	Parameter	Description
detect		search for objects (without image capture)
snapdetect		Image capture, then search for objects
results		List X, Y, Z Position of the found objects
red	Mode	Set the rotation mode of the results. Example: red rxyx (see below 13.3 Rotation modes)

10 WEB GUI

The web GUI runs on port 8080 by default. The port can be set during installation. This can be changed in the console and saved in configurations.

Chrome, Firefox, and Edge can be used from all operating systems. We recommend Chrome on Windows 10 for an optimal experience. The minimum recommended resolution is 1152 × 864 (XGA+).

In the following sections, the individual operating pages of the user interface are each shown as examples with a screenshot, and the individual elements are described.

10.1 OVERVIEW

The Mikado Find control page contains a button bar on the left, the camera image in the center, and a status bar at the bottom.
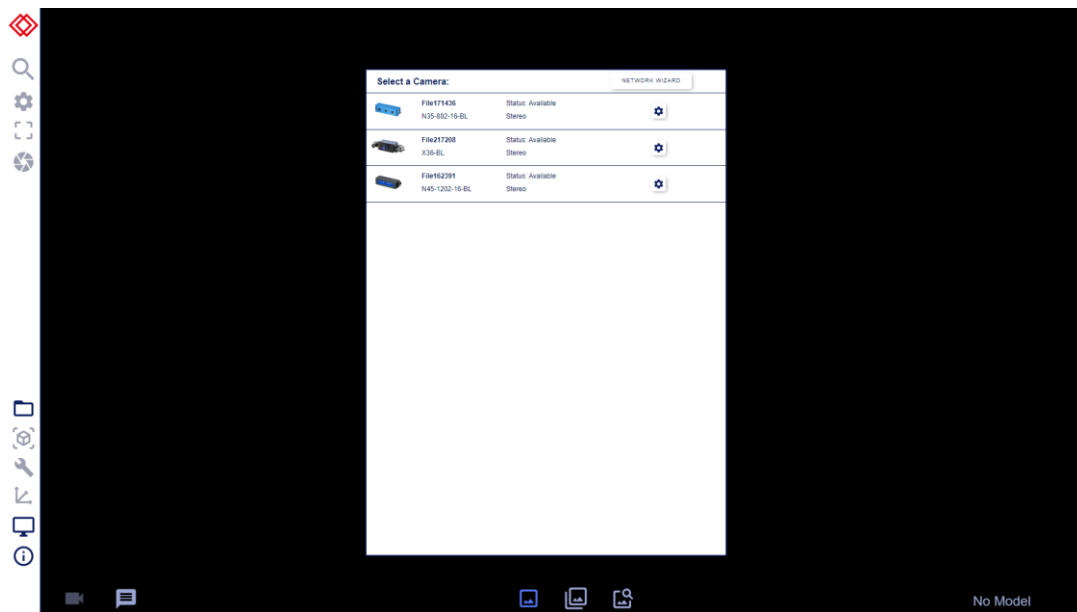
The application -operation page at the beginning

In the upper left bar there is the "Actions" button: the detection and its parameters, as well as a button for image acquisition. In the lower part of the left bar there are the "Setup" buttons that prepare the application before a detection.

The status bar at the bottom has two buttons on the left: Camera Setup and Messages. In its middle part there are the buttons for the display modes. In the right part of the status bar we can find the file name of the current 3D model, if selected.

In the middle part (in black in the figure above) the rectified image of the left stereo camera is displayed.

10.2 CAMERA SELECTOR SWITCH

10.2.1 SELECTING A CAMERA
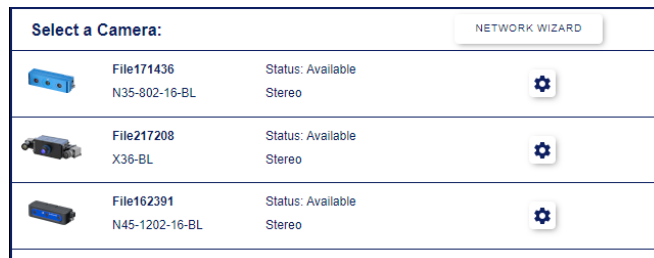
The camera selection window

When the application starts, it opens the connected stereo camera. If there is more than one, the application displays the camera selection window instead, listing the connected cameras with their model and status. If there are none, an empty camera selection window is displayed, which is periodically updated if a camera is connected in the meantime.

It is possible to change the camera at a later time by clicking on the camera menu button in the status bar.

Clicking on the line opens the corresponding camera, if available.

10.2.2 CONFIGURE A CAMERA

With the setup icon (gear) on the right side you can set up the camera. This button opens the camera in NxView, where you can edit the parameters of the camera and then save them in the predefined Json file.
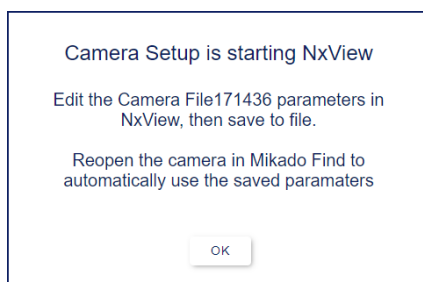
Before NxView starts automatically, an explanatory popup appears.

You can find the camera parameters in NxView by pressing F8 or via the Capture > Parameters menu or the parameter icon.

*The NxView camera parameters have several tabs
and a Save button in the upper right corner.*

When you are done, click Save and overwrite the current json file. Close NxView to complete the process. The saved settings (including exposure, Flexview proportion, stereo algorithm, ...) will be used the next time the camera is opened in Mikado Find.

10.2.3 UPDATING A CAMERA FIRMWARE

If the camera firmware is different from the server firmware, a button appears to update the camera firmware.
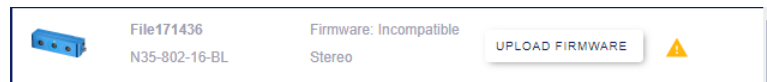
The "Upload Firmware" button for updating the camera firmware

10.2.4 SETTING UP THE IP OF A CAMERA

The Network Wizard button is enabled if the web GUI is running on the same computer as the Mikado Find console. Otherwise, it is disabled.

The Network Wizard button

You can use this network wizard to troubleshoot the connectivity of unrecognized cameras, for example.
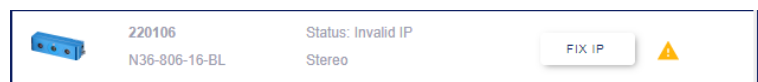
The Network Wizard "Fix IP" button

If the IP address of a camera is not valid, the selector displays the "FIX IP" button, which starts the network wizard specifically for that camera.

10.3 CAMERA MENU
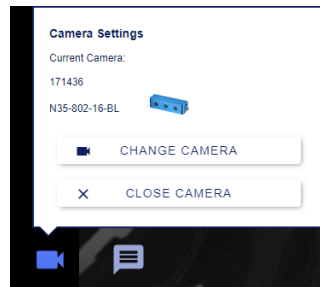
The camera menu

The camera menu shows the name and model of the current camera. The "Change Camera" button opens the camera selection window so that another camera can be selected.

Close Camera" closes the current camera.

10.4 ADVANCED MENUS

Some menus have hidden advanced parameters that are only visible in advanced mode.
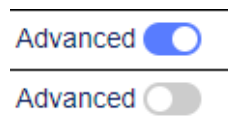
Toggle switch in the upper right corner: examples for extended mode ON and extended mode OFF

In this case, the menus have an "Advanced" button in the upper right corner that can be used to turn the advanced mode on and off. The "Search parameters" menu, the "Roi" menu and the "Model" menu all have their own extended mode.

10.5 SETUP TYPES

📁 Configuration menu

📦 Load 3D model

🔧 Generate search model

📈 Set up workspace

🖥 GUI settings

ⓘ Information

There are 6 buttons in the lower part of the left bar.

The first four buttons (Configuration Menu, Load 3D Model, Create Search Model and Set Up Workspace) have a vertical status bar on their right side that turns green if they have already been set up and red otherwise. In addition, the status bar of the Configuration menu turns orange if a change has not been saved.

The buttons with their status bar here mean, for example, that the configuration is not saved, the 3D model is loaded, no model has been created yet and the 3D reference point of the workspace is defined.

10.5.1 CONFIGURATION MENU
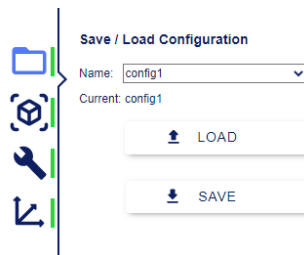
Menu for loading and saving configurations

The parameters of the application can be saved and loaded as configurations. The parameters included in the configurations are the camera parameters, the model parameters, the search and the region of interest (ROI).

The drop-down menu offers a list of existing configurations to load or overwrite. A new name can be entered there to create a new configuration. The current configuration is displayed below this drop-down button.

Clicking Load or Save will attempt to load or save the configuration selected in the drop-down menu.

A confirmation dialog will then appear.

When loading a configuration file, there is the possibility to load the camera parameters from the file or to keep the existing parameters (exposure, used fitting algorithm, ...)

After setting the camera parameters in NxView (see below. 10.2.2 Configure a camera), it is advisable to deactivate the option so that the new parameters are not overwritten by the parameters of an older configuration.
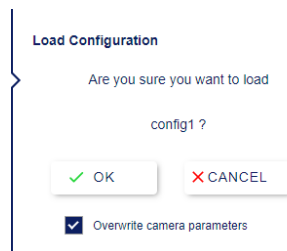
Confirmation dialog for loading, with the option" Overwrite camera parameters".

To remove unused configurations from the list, a supervisor-level user must manually delete the corresponding files in the configuration folder.

These .zip files are located in the folder:
*C:\Users\[User Name]\AppData\Local\Optonic GmbH\
Mikado_FindMikado_Find-130\Data*

10.5.2 3D MODEL SHOP

10.5.2.1   SELECTING OR UPLOADING A CAD FILE

The application detects objects described by a 3D model file in Stereolithography Format (STL) or Polygon File Format (PLY). Press the Load 3D Model button to open a file dialog where you can browse and select a specific file.

Model file selection dialog

You can either click on a file that is already in the list (and on the server) or upload a new file from your local computer.

Currently, only **non-accented characters** are allowed in the model name.

To remove unused CAD models, a supervisor-level user must manually delete the corresponding files in the model folder.

These .stl or .ply files are located in the folder:

C:\Program Files\Optonic GmbH\Mikado_Find\Mikado_Find-130\Bin\Web\models

10.5.2.2   VISUAL CONTROL

The screen is then filled with a 3D view of the model that you can examine and rotate. The 3D axis of the model are displayed in red (X), blue (Y) and green (Z). This is important to determine if the model is symmetrical along certain axes, as this can be used later to optimize detection.
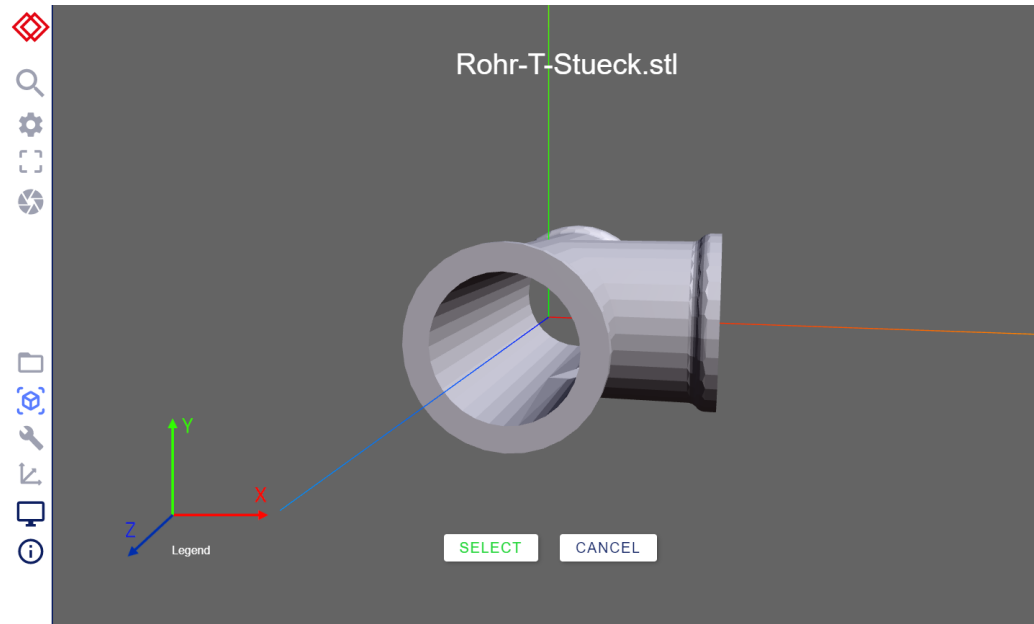
3D model examination screen. X-axis in red,
Y-axis in green (pointing upwards) and Z-axis in blue.

The legend for the axis colors is displayed in the lower left corner

If this is the correct model, click the Select button, otherwise click the Cancel button and restart the procedure to load the 3D model by clicking the Load 3D Model button again.

10.5.2.3 SETTING A GRIPPING POINT

Clicking on the minimized CAD window in the lower right corner of the screen displays the 3D view of the CAD model. It is possible to define a customized grab point by adding translation and rotation to the 3D origin of the CAD model.

The grip point is **defined** in **object coordinates**, not in world coordinates.

This transforms the result coordinates of the detected objects accordingly and can be used to define a grab point and orientation on the CAD model.
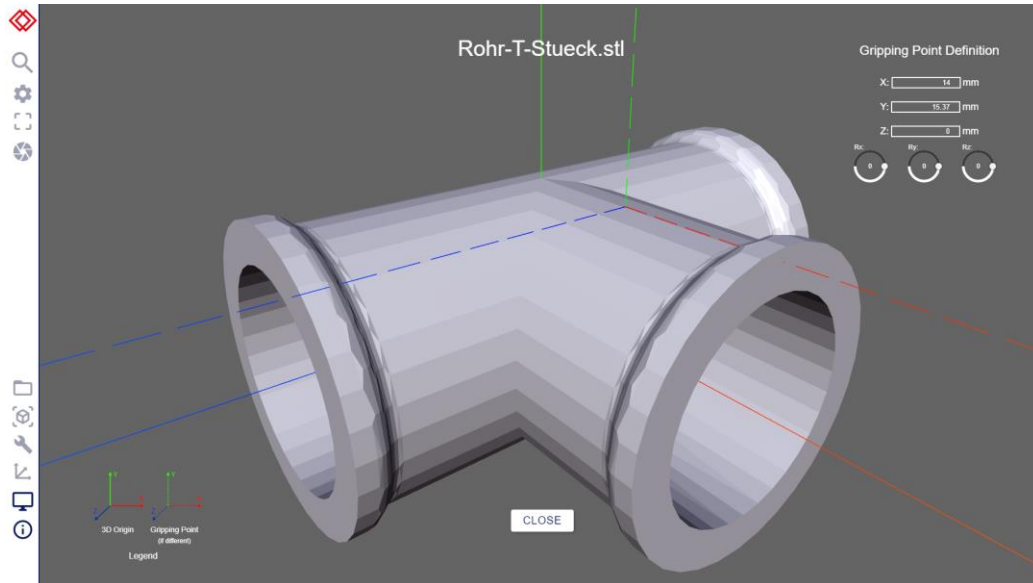
The new 3D origin is shown by dashed lines

The grab point settings are located in the upper right corner of the view.
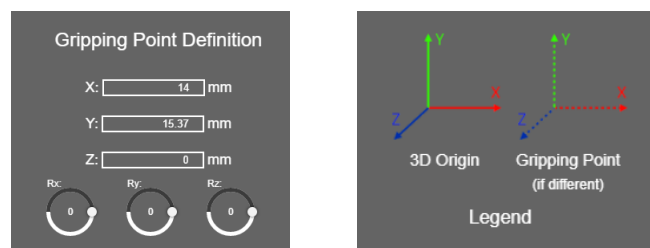
Grip point settings and legend of the axis display.

The legend describes how the new 3D origin is drawn.

The gripping point is saved with the configurations.

⚠ Note that it is used for each model. So be sure to reset it if it no longer fits your model.

10.5.3 GENERATE SEARCH MODEL

Before we can recognize the objects, the application must create a search model of the object. The Model menu gives control over the generation parameters.

There are two versions of this menu: the basic version and the advanced version. The advanced version can be switched on and off with the "Advanced" button in the upper right corner of the menu (see below. 10.4 Advanced menus).
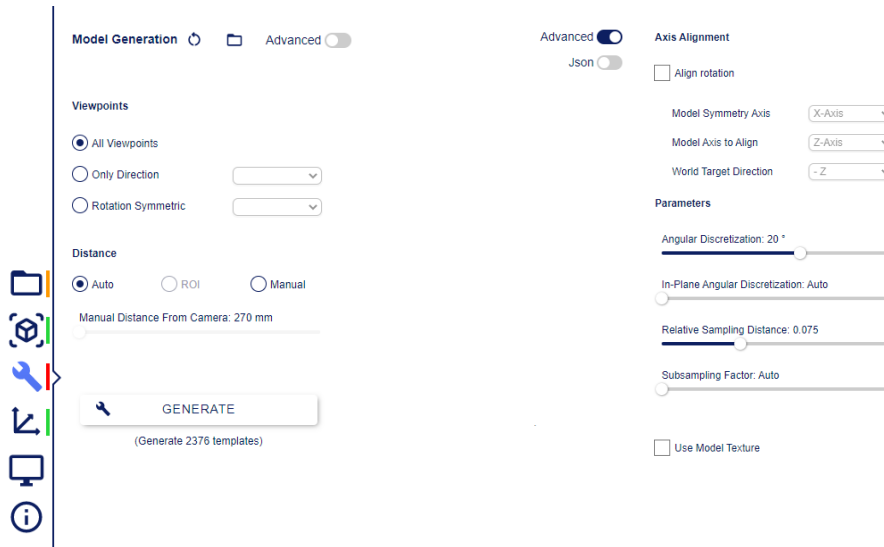
Basic menu for model creation

Extended menu with additional parameters

It also provides the "Generate" button that creates the search model and stores it in memory. Once the search model is generated, the search is ready.

A model contains a certain number of templates. This number is directly influenced by the selected parameter values. Using more templates can lead to higher accuracy at the expense of speed. The estimated number of templates that will be generated is displayed under the Generate button (the number of templates is not displayed if Json parameters are used).
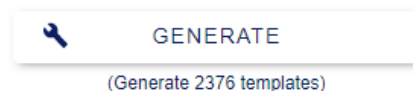
Estimated number of templates under the "Generate" button

The parameters of the model generation can be reset to default values (see below. 10.5.3.11 Reset to default values). They can be saved or loaded from files (see below. 10.5.3.12 Load / Save Model Parameters).

The parameters of the model generation are:

10.5.3.1 VIEWPOINTS

"Viewpoints" defines the points from which the model is generated.

This can be used for (rotationally symmetric) objects or as a preset for the expected pose. Each viewpoint is used to create a template and the visible surface points are saved for refinement.

You can choose:

- "All Viewpoints": All viewpoints.

- "Only Direction": templates are created to cover a 45 degree cone along the desired axis and direction (positive or negative or both).
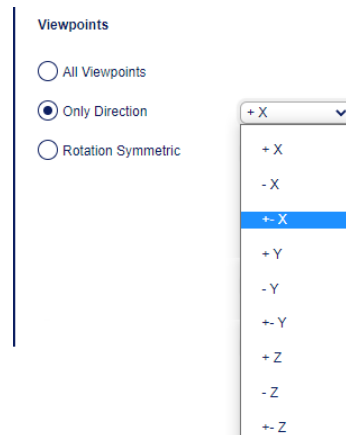
Only the desired axis and direction

- "Rotation Symmetric": Rotation symmetric about the X-axis, Y-axis or Z-axis.
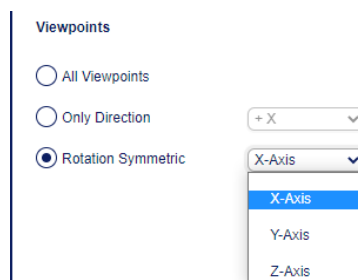
Rotation symmetrical about the X-axis, Y-axis or Z-axis

The taper angle can be set between 1 and 45 degrees in the advanced menu.

Only Direction   45 °   + X ⌄

Selection of the taper angle in the advanced menu

10.5.3.2   DISTANCE

"Distance (distance, in mm) at which the model will be rendered. Should roughly correspond to the distance at which it will be adjusted.

You can let the application calculate it (Auto), or use a defined ROI (ROI) (see below. 10.6.2 Set ROI), or set it manually here (Manual) by using the following slider Manual distance.

The advanced parameters of the model generation are:

10.5.3.3   AXIS ALIGNMENT

"Axis Alignment" is an important feature that eliminates ambiguities in the alignments of the results.

Some symmetrical parts can be detected with an arbitrary rotation. This option allows to return consistent orientations by specifying that an axis of the object should coincide with a world coordinate axis.

"Model Symmetry Axis" is the symmetry axis of the object

"Model Axis to Align" is the axis to be aligned with a word coordinate axis

"World Target Direction" is the world coordinate axis to which the object should be aligned

10.5.3.4   ANGULAR DISCRETIZATION

"Angular discretization" is the average angle between two templates (in degrees).

10.5.3.5   IN PLANE ANGULAR DISCRETIZATION

Approximate angular step between rotated template views.

This parameter controls how many rotated versions should be added to each generated template view. One rotated version is added for each angular step. A low value means more rotated versions of the templates. A value of 0 means that the angle is calculated automatically instead.

10.5.3.6   RELATIVE SAMPLING DISTANCE

Specifies the distance of the sampled point on the model surface relative to the model diameter. The sampled points are used to refine the final pose. Lower settings result in more precise poses, especially for models with fine geometric details.

10.5.3.7 SUBSAMPLING FACTOR

Subsampling selects the resolution of the template search. Subsampling 1 means that the template search is performed at the resolution of the rectified images. If you lower the search resolution below 1, the search process is accelerated considerably. Long thin parts or parts with small features or textures will benefit from better alignment if a higher resolution is used. The special value 0 means that the factor is calculated automatically.

10.5.3.8 USE MODEL TEXTURE

This means that texture information is taken into account when calculating the evaluation of hypotheses and refined parts. The texture information is stored as surface colors in .ply model files. Using texture evaluation can also improve the results when using models without texture.

10.5.3.9   JSON PARAMETERS
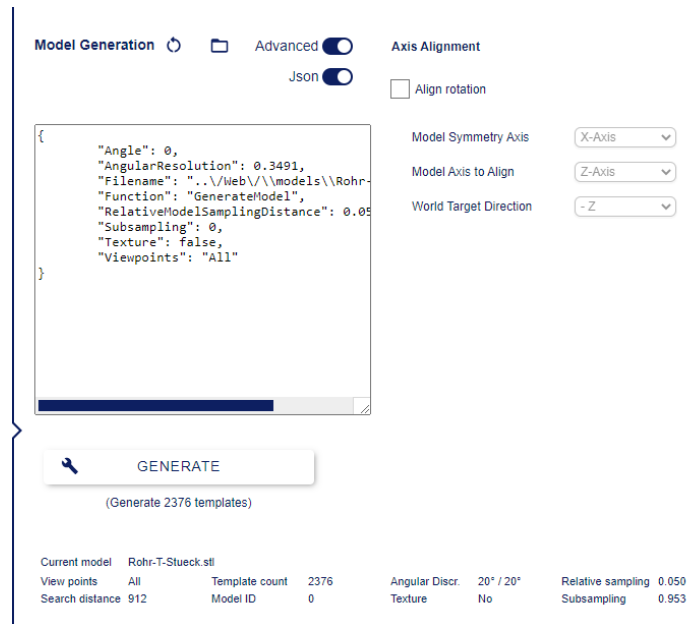
*The parameters can be entered as Json
in a resizable editor window*

The parameters can also be entered as Json (JavaScript Object Notation, a compact text data format) in a text area whose size can be adjusted. This is especially useful for importing/exporting parameters between Mikado Find and NxView or NxTreeEdit. The Json parameters option can be activated with the Json button, which is only visible in the advanced menu ("Advanced").

The json edition window can be enlarged by dragging the resize icon in the lower right corner ( ).

*The model generation icon displays "Json",
when the Json parameters are set*

10.5.3.10 CURRENT MODEL INFORMATION

The information about the current model (loaded or created) is displayed under the Create Model button. The parameter values used for creation, the model ID and the number of templates created are displayed.

GENERATE
(Generate 2376 templates)

Current model	Rohr-T-Stueck.stl						
View points	All	Template count	2376	Angular Discr.	20° / 20°	Relative sampling	0.050
Search distance	912	Model ID	0	Texture	No	Subsampling	0.953

Information about the current model

10.5.3.11 RESET TO DEFAULT VALUES

When you click on ↻ the reset icon, all model parameter values are immediately reset to the default values.

10.5.3.12 LOAD / SAVE MODEL PARAMETERS

The button "Folder 🗀 opens a dialog for loading/saving.
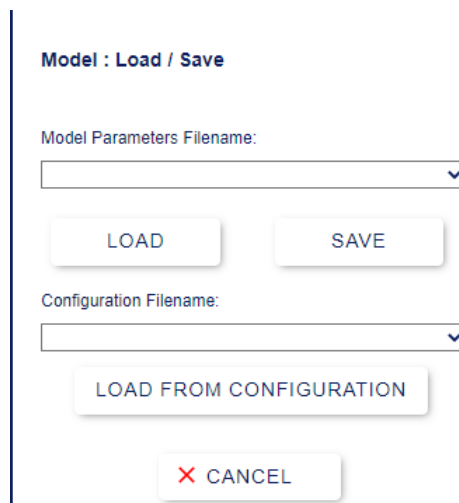
Model : Load / Save

Model Parameters Filename:

	⌄

| LOAD | SAVE |

Configuration Filename:

	⌄

LOAD FROM CONFIGURATION

✕ CANCEL

Load/Save dialog

From this dialog it is possible to save or load parameter files or load from configurations.

To remove unused parameter files, a supervisor-level user must manually delete the corresponding files in the parameter folder. These .json files are located in the folder:

C:\Users\[User Name]\AppData\Local\Optonic GmbH\Mikado_Find\Mikado_Find-130\Data__parameters

Saving individual parameter files

Enter a new name in the Parameter Filename field and click *Save*

If the file already exists, a dialog will ask you if you want to overwrite the file.

Overwrite Model Parameters File

Are you sure you want to **overwrite**

testmodel2 ?

✓ OK   ✗ CANCEL

Override permission dialog

The files are stored on a server in the *Data__parameters* folder.

Loading individual parameter files

Previously saved files are displayed in the drop-down selection. Select the desired file and click "*Load*" to update the parameters with those in the files. The number of updated parameters is displayed in the message window.

2022-07-08 12:22:56.700 : Loaded 8 Model Parameters
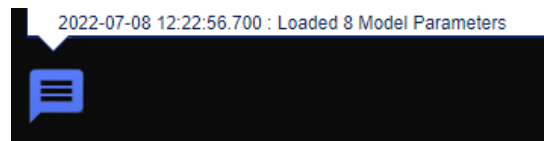
The number of loaded parameters is displayed in the message window

Loading from the configuration

It is also possible to load the model parameters contained in a previously saved configuration. Select the desired configuration from the drop-down list for the name of the configuration file and click "*Load From Configuration*".

The model parameters will now match those saved in the configuration without loading other parts (camera, search parameters, settings) of the configuration. The number of updated parameters is displayed in the message window.

⚠ When loading model parameters, no stl/ply model file is loaded, only the parameters for template creation.

10.5.4 WORKSPACE SETUP
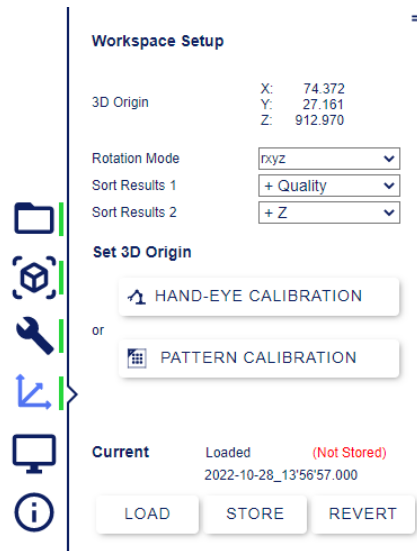
The workspace menu.
This menu can be moved on the screen
(click on the handle icon in the upper right corner).

The Workspace Setup menu (which replaces the former Version 1.1 Calibration menu) allows you to set not only the origin of the acquisition coordinate space, but also the rotation mode of the results and the sorting of the results.

10.5.4.1 THE ROTATION MODES

The rotation mode defines how the orientation of the detected objects is displayed. This rotation in the world coordinate system can be encoded in different formats: Euler angles, quaternions, angular axes and condensed Rodrigues format.

The Euler angles can be any of the 24 combinations. Some industrial robots have known Euler angle formats available under the robot manufacturer's name (see below. 13.3 Rotation modes).

Note that there is no postprocessor that ensures that the angle can be used directly as robot coordinates. Possible collisions and ambiguities in the Euler angles must be additionally considered.

It is also possible to enter the rotation mode manually: It starts with an 'r' (intrinsic) or 's' (extrinsic) prefix, then the three rotation axes follow. For example: *rxyz*, or: *sxyx*

10.5.4.2 SORTING THE RESULTS

By default, the results are sorted by the best quality first. It is possible to specify two sorting levels, the second one being used in case of equality on the first level. Note that this is unlikely, as the values used for sorting have three decimal places.

The possible sorting options are '+' or '-'. Quality and '+' or '-' X, Y or Z positions

'+' means that the highest values come first, '-' that the lowest values come first.

10.5.5 3D REFERENCE POINT CALIBRATION

Set 3D Origin

⚲ HAND-EYE CALIBRATION

or

▦ PATTERN CALIBRATION

There are two ways to set the reference point.

When objects are localized, their coordinates are given in the world reference and in units of measurement (mm). It is therefore important to define a point as a reference point from which measurements are made.

Calibration determines exact transformation of the camera to the reference point.

There are two options:

- *Hand-Eye Calibration*. Finds are then output in the coordinates of the robot

or

- *Pattern Calibration* to define a match between the camera position and a pattern position

10.5.5.1 HAND-EYE CALIBRATION

Mikado Find can provide the detected object positions directly in a third-party robot arm coordinate system.

We use a robot with a mounted tip, calibrate the robot tool coordinate system (TCP, Tool Center Point), and then move the tip in three different positions as described in more detail below.

10.5.5.1.1 TOOL CENTER POINT CALIBRATION

Tool Center Point (TCP) must be calibrated.

We proceed as follows:

1. Attach tip to gripper.
2. Attach second tip to tray.
3. Approach tip and use robot manufacturer's measurement program.

The TCP is correct if the mounted tip is exactly aligned with the base tip, even if the mounted tip rotates around it.

An example of implementation can be found here:

https://www.youtube.com/watch?v=lb3jQwTA9Ag

For example, you can store the tip TCP under "tool1" and use "tool0" for the actual TCP of the gripper.

In this case, you would use "tool1" for the hand-eye calibration. Once the calibration is complete, you only need to switch to "tool0" to use the gripper in the same coordinate system as Mikado Find.

10.5.5.1.2   CREATION OF A 3D MODEL OF THE TIP

The front part of the tip on the gripper of the robot must be modeled as a CAD model. The coordinate system should lie exactly in the tip and the Z-axis should point axially away from the gripper.
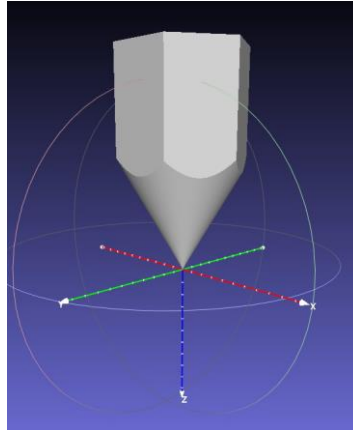
A 3D model of the tip aligned with the Z axis
and the zero point at the end of the tip

Select your lace CAD file in Mikado Find (see below. 10.5.2 3D Model Shop) and create a new model (s.b. 10.5.3 Generate search model).

10.5.5.1.3   TIP MEASUREMENT
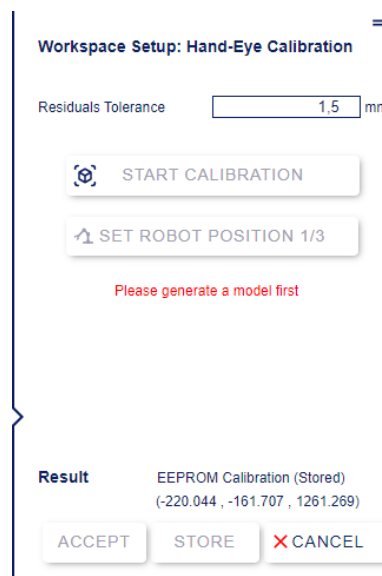
Hand-eye calibration menu, requires the creation of a tip search model.

The robot arm is moved with the mounted tip to three different positions from which the coordinate system connection is derived.

Preparation

a. Select the model of your tip. (see below 10.5.2 3D Model Shop).

b. Create a search model (see below. 10.5.3 Generate search model).

c. Open the ROI to activate an ROI that will be set later by clicking on the screen (see below. 10.6.2 Set ROI).

d. Go to the workspace menu and click on "Hand-Eye Calibration".

e. Set the residual tolerance. The default values are set to 1.5 mm.

With the robot, we will try to cover three corners of the field of view. The positions do not have to be exactly in the corners, but should ideally approximate a right-angled triangle.

First step

1. Move the robot to a first corner position, read the TCP x, y, z positions of the robot. Make sure you are using the correct tool, e.g. "tool1" that was saved in the TCP calibration step (see below. 10.5.5.1.1 Tool Center Point Calibration).

2. Click the tip in the screen to set the ROI for the tip.

3. Press "Start Calibration" to start the search.

The calibration menu when the system is ready,
to start the hand-eye calibration first step

If the detection is not successful, you can change the parameters and try again with the "Find Model" button.

Only one object must be detected. If there is more than one result, a popup with a warning will appear. Close it by clicking OK. You will then have the option to select the correct result on the screen.

⚠ Hand-eye calibration

Find model : too many objects detected 2 > 1
Please select the target object

OK
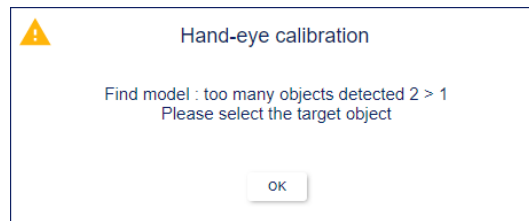
Popup warning when too many peaks are detected

A selection popup appears when you click on a result. Click Select to confirm the selection if this is the expected result.

Detected Object #1   ═

Score: 90 %

X: -662.254 mm
Y: 176.781 mm
Z: 99.545 mm

Select this calibration target ?
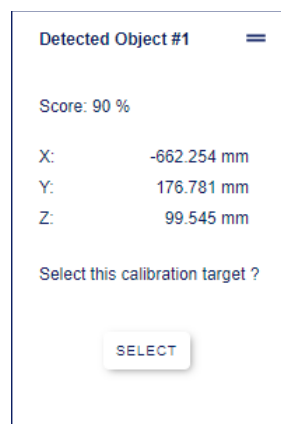
SELECT

Selection popup
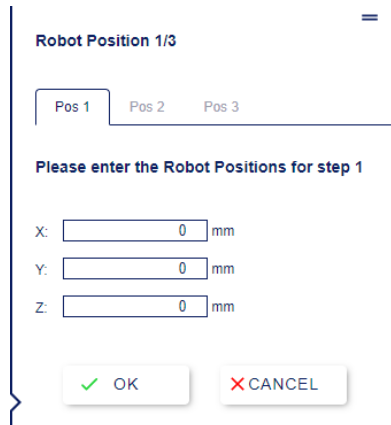
Note that this is unlikely to be the case with exact search parameters and an ROI.

You can repeat the tip detection as many times as necessary by clicking Find Model.

Workspace Setup: Hand-Eye Calibration

Residuals Tolerance   1.5   mm

⦿   FIND MODEL

⚐ SET ROBOT POSITION 1/3

After the first detection, "Start Calibration" becomes "Find Model".
and can be used to perform a recognition again.

4. Click on "Set Robot Position" and enter here the robot position ("tool1") read in substep 1 and click on OK

Robot Position 1/3

| Pos 1 | Pos 2 | Pos 3 |

Please enter the Robot Positions for step 1

X: 0 mm
Y: 0 mm
Z: 0 mm

✓ OK   ✕ CANCEL
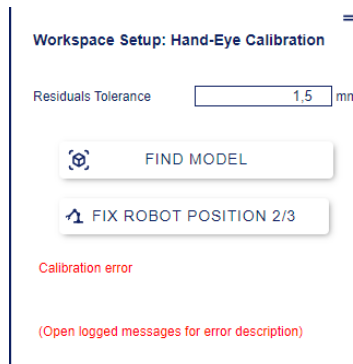
The "Set robot position" window asks for the robot position

Second step

Move the robot to position the tip at the second corner position, read the x, y, z positions of the robot and repeat substeps 2, 3 and 4 of First Step.

The button for step 2 is now labeled "Find Model ".

When confirming the second robot position, the system checks if the distance between the points is valid. If not, an error is displayed.

Workspace Setup: Hand-Eye Calibration

Residuals Tolerance   1,5   mm

⊙ FIND MODEL

⚲ FIX ROBOT POSITION 2/3

Calibration error

(Open logged messages for error description)

Example of an error that occurs when the entered points do not match the detected points.
For more details, please refer to the logged messages window

The details of the error are displayed in the Logged Messages window (see below). 10.7.1 Messages).

You can then either correct the entered positions (via the "Fix Robot Position" button) or try a new detection with "Find Model".

You can also correct the previously entered positions using the corresponding tabs.

Third step

Move the robot to a third corner position, read the X, X, Z positions of the robot and repeat substeps 3, 4 and 5 as in the previous steps.

When the third position is entered and valid, the link between the two coordinate systems is calculated.

If successful, a success popup is displayed. The 3D axis at the origin is also displayed if it is in the camera's field of view. The calibration is automatically saved in the calibration folder.

✓   **Calibration Success**

Robot Calibration is successful
Mikado Find and Robot coordinates systems are now linked

OK
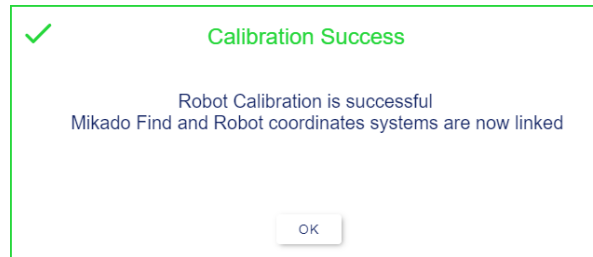
The successful calibration is confirmed by the success popup

⚠️ **The calibration must now be accepted, saved or canceled.**

Result   Calibrated   (Not Stored)
(126.508 , -30.433 , 1796.930)

ACCEPT   STORE   ✕ CANCEL
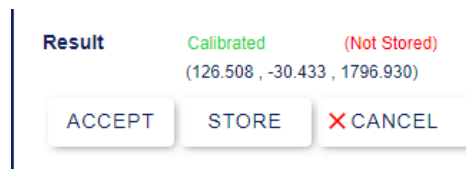
A successful calibration is not automatically saved, the user must accept, save or cancel it.

"Accept" and "Store" mean that the new calibration becomes the current one. A json file is also saved, the calibration can be loaded in the future.

"**Store**" additionally saves the calibration in the internal memory of the camera (EEProm), so that the camera uses this calibration automatically.

"Cancel" discards the new calibration and returns to the previous one.

You should always click "**Store**" a successful calibration. "**Accept**" is only useful if you want to perform a test before saving. **"Cancel**" if the calibration was not satisfactory, and then start again. The had-eye calibration can be reset at any time with the "Cancel" button.

After saving the calibration, you can remove the tip from the robot arm and switch back to the TCP gripper (e.g. tool0) to use it.

10.5.5.2   CALIBRATING PATTERNS
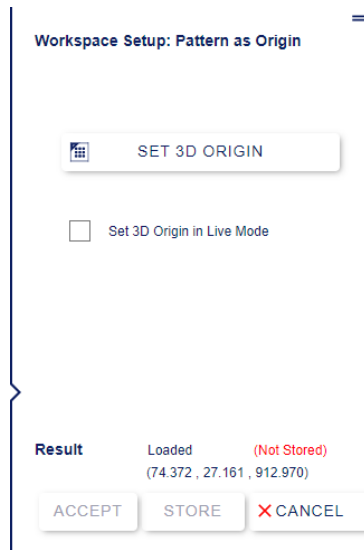
The pattern calibration menu.
This menu can be moved on the screen
(click on the handle icon in the upper right corner).

The 3D reference point can be set by placing a calibration pattern in the stereo camera's field of view. The center of the pattern will be the reference point.

More information about the samples is available online here: https://www.ensenso.com/manual/3.3/guides/calibration/patterns.html
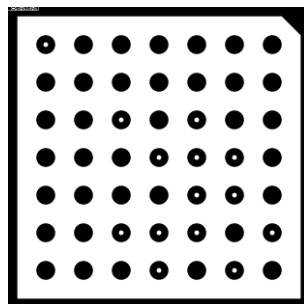
Example of a calibration pattern

Then either press the "Set 3D Origin" button or check the "Set 3D Origin in Live Mode" checkbox to start a live calibration mode. The application will attempt to detect the pattern. If the pattern is detected, it will be highlighted and its 3D axis will be displayed.

The pattern must be visible in both the left and right images of the stereo camera, but the main image in Mikado Find shows only the left image. Therefore, when the calibration menu is open, the right image is displayed in a pop-up window in the lower right corner of the screen.
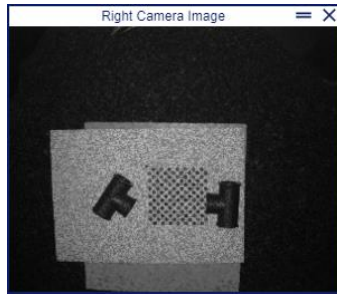
The popup shows a reduced right image

If the pattern is only visible in one camera, the calibration fails with an error.
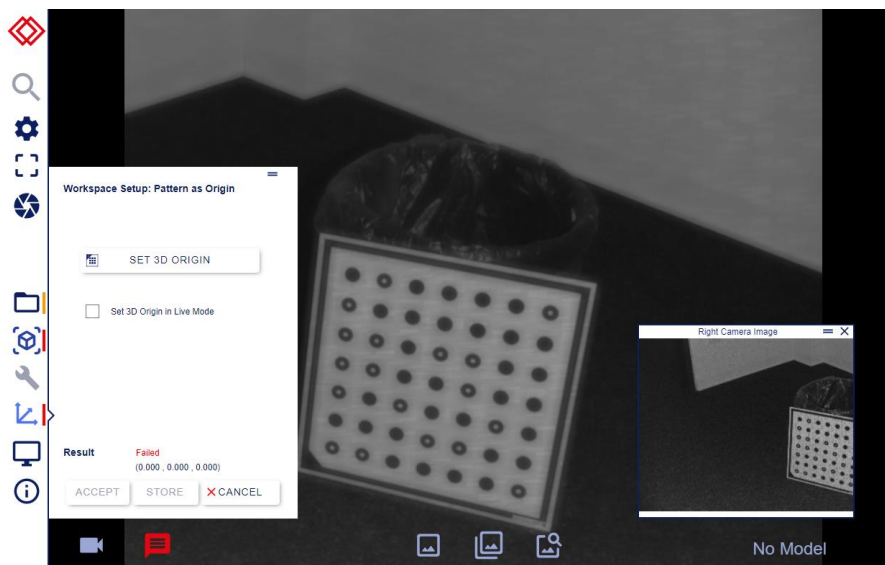
*Calibration fails here because the
pattern is not visible on the right image*

⚠️ **A successful calibration must then be accepted, saved or canceled.**

"Accept" and "Store" mean that the new calibration becomes the current one. A Json file is also saved, the calibration can be loaded in the future.

"**Store**" additionally saves the calibration in the internal memory of the camera (EEProm), so that the camera uses this calibration automatically.

"Cancel" discards the new calibration and returns to the previous one.

10.5.5.3   LOAD CALIBRATION, SAVE, RETURN

The "Load" and "Save" and "Return" buttons are located at the bottom of the workspace menu.

10.5.5.3.1 CALIBRATION LOADING

The reference point can also be loaded from a previous calibration.

The accepted or saved calibrations are stored in the "*Data__calibrations*" folder. In addition, when a configuration is loaded, the included calibration file is copied from the configuration to the calibration folder for easy access.

The "Load" buttons open a dialog with a list of available calibrations. Only the calibrations performed for the current camera are displayed. The file name of the calibration is composed of the camera serial number and the date on which the calibration was performed.
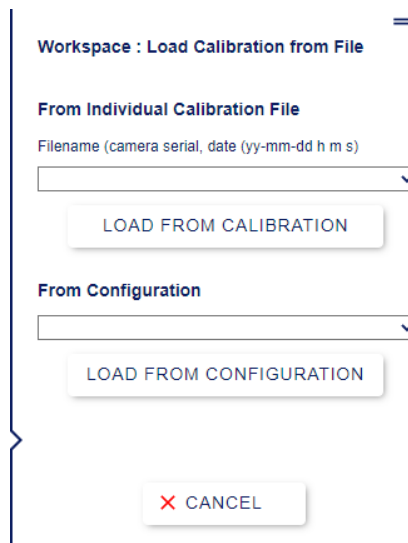
*Load the calibration either from json files
or from whole configuration files.*

It is therefore easy to select the desired calibration file and then click Load to set the reference point defined in the file.
Mikado Find then automatically captures an image to update the reference frame and set the reference point.

To remove unused calibration files, a supervisor-level user must manually delete the corresponding files in the calibration folder. These .json files are located in the folder:

C:\Users\[User Name]\AppData\Local\Optonic GmbH\Mikado_Find\Mikado_Find-130\Data__calibrations

10.5.5.3.2 SAVE CALIBRATION

"Store" additionally saves the calibration in the internal memory of the camera (EEProm), so that the camera uses this calibration automatically. It also stores a json file in

C:\Users\[User Name]\AppData\Local\Optonic GmbH\Mikado_Find\Mikado_Find-130\Data__calibrations

so that the calibration can be loaded individually.

Current   EEPROM Calibration (Stored)

The calibration status is then displayed as "stored in EEProm".

10.5.5.3.3   RETURN TO EEPROM CALIBRATION

If a calibration is accepted, but after some tests it is not considered good enough, it is possible to reset the calibration to what was previously stored in the camera's EEProm.

When you click Revert, the current calibration is replaced by the calibration stored in the camera.

10.5.6 GUI SETTINGS
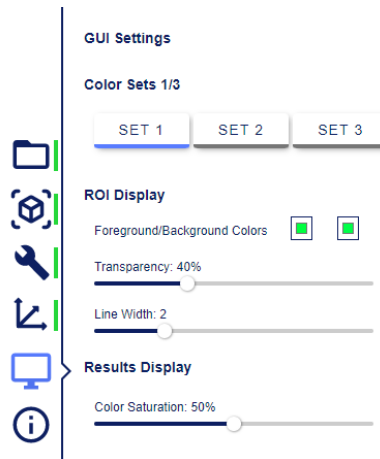
GUI options

The GUI menu contains options to change the appearance of the application.

10.5.6.1 COLOR SETS

Nine different color sets are available to show the various possibilities, with a custom color scheme available upon request. There are three pages, each with three color sets. Click on the page number to go to the next page.

Three different color sets

10.5.6.2 ROI DISPLAY

Here you can select the colors of the lines indicating the area of interest (ROI). There are two colors, one for ROI lines in front of the object (foreground lines) and one for lines behind the object (background lines).

The transparency slider controls an additional transparent effect for the lines behind the objects (background lines).

The line width defines how thick the ROI lines should be drawn.

10.5.6.3 RESULTS DISPLAY

Color Saturation: The detected objects are highlighted in a color that goes from green to yellow depending on the quality of the detection. The saturation of these colors can be adjusted with the last slider.

10.5.6.4 SWITCH MENUS BACK ON

Some actions disable the GUI menus while they are being processed.

If the server has a problem, it may happen that the signal for re-enabling does not reach the GUI. In this case it is possible to re-enable the GUI manually here. The "*Re-enable Menu*" button appears when all menus are disabled. Only use this button if you suspect an error.
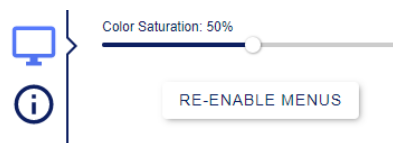
The Re-enable Menu button

10.5.7 INFORMATION

The last button in the left bar shows the information menu.
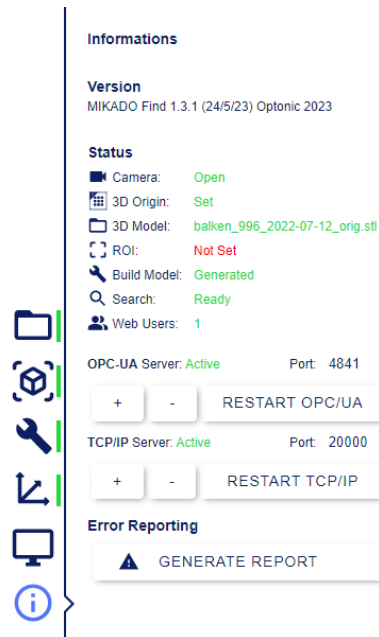
Information menu

10.5.7.1 VERSION OF MIKADO FIND SOFTWARE

To further verify the version number, the application hash is hidden under the version number. This hash is very useful for debugging as it allows accurate identification of the code used.
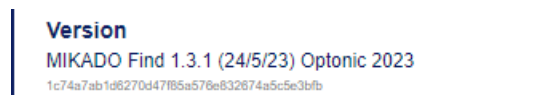
The hash is hidden under the version number

Move the mouse under the number to see the hash and copy it.

10.5.7.2 APPLICATION STATUS

You can check here whether the camera is open (see below. 10.2.1 Selecting a camera), whether the 3D reference point is set (s.b. 10.5.5 3D reference point calibration), whether the STL model is loaded (s.b. 10.5.2 3D Model Shop), whether the ROI is set and in use (s.b. 10.6.2 Set ROI), whether the search model has been generated (see below. 10.5.3 Generate search model) and whether the search is ready.

10.5.7.3   OPC UA AND TCP/IP SERVER

The states of the OPC UA and TCP servers with the respective port numbers follow. Here you can change the port number by pressing the + and - keys. Long pressing of a key increases or decreases the port number by 100. You can then press the "Restart Server" button to change the current port number. The "Restart Server" button is disabled unless it is needed.

If you close this menu without restarting the server, the port number displayed will be reset to the current number.

10.5.7.4   GENERATE ERROR REPORT

Lastly comes the Generate Report button, which helps with troubleshooting.

Error Reporting

⚠ GENERATE REPORT

Last report: Report_1641905973.zip

Once the report is created, it will be linked under the button.
If you click on this link, you can download the report

When you press this button, a report is created as a zip archive in Logs (see below. 7.2 Log file) is generated. You can download the report by clicking on the link that appears below the button. Please send us at Optonic this report for further analysis.

10.6 SEARCH BUTTONS

Search

Search parameters

Set ROI

Recording a new image

In the upper part of the left bar there are 4 buttons for search and related actions. In the next section we will describe them in the following order: Capture a new image, Set ROI, Search parameters and Search.

10.6.1 RECORDING A NEW IMAGE

This function is not needed to perform a search, but is useful to set up the camera.

10.6.1.1 *SINGLE FRAME MODE*

Click the Capture button to capture and display an image.

10.6.1.2 *LIVE MODE*

Pressing the record key for a long time (a few seconds) starts a live mode instead of single image recording. The record button turns red when the live mode is switched on. Click the record key again to end the live mode.

10.6.2 SET ROI
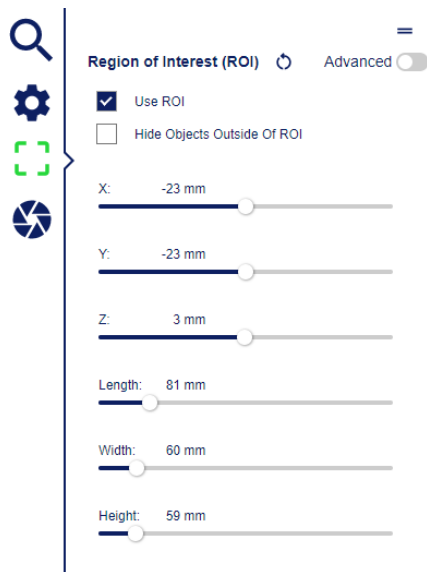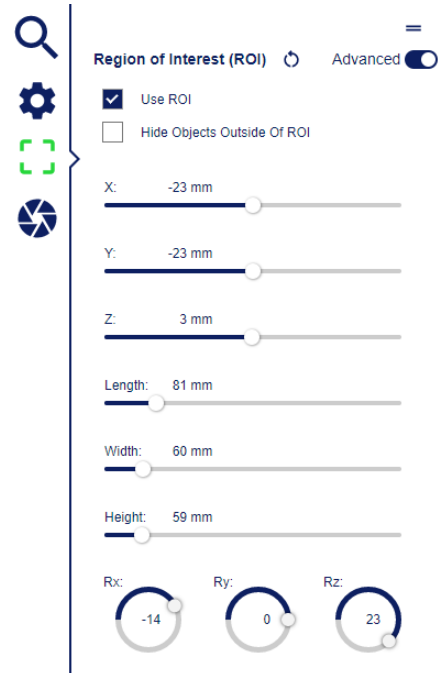
The ROI basic menu   The ROI Advanced Menu

A region of interest (ROI) can be defined to search only in a specific 3D volume rather than the entire volume covered by the stereo camera.

Click on the "Use ROI" checkbox to activate the ROI. This will automatically take an image to calculate the minimum and maximum ROI range.

To delete the ROI, there is a reset button. It removes the ROI points and allows a new selection with the mouse or the sliders.

The menu has a basic and an advanced mode where additional rotation controls are displayed.

The menu can be moved so that it does not obscure the possible position of the ROI on the left side.

All displayed values are input fields that can be changed using the keyboard or the mouse wheel. The sliders can also be controlled with the mouse wheel.

10.6.2.1 ROI DEPTH RANGE HIDE

"Hide Objects outside Of ROI": Hide objects outside of the ROI. If this option is enabled, only the details inside the ROI are visible. This is very useful to confirm the presence of the target object in the ROI.

10.6.2.2 ROI SLIDER

The ROI is defined by its position in relation to the reference point and its dimension. In the extended menu it is also adjustable to define the rotation of the ROI.

Position: The X, Y and Z values are the displacement of the ROI center to the reference point. The reference point is the camera zero point in the uncalibrated system and the calibrated zero point in the calibrated system, see under "Calibration".

Dimension: The length, width and height of the ROI are its dimensions along the X, Y and Z axes as calibrated, otherwise along the camera axes. By default, the dimensions are set to 100 mm.

Rotation: In the Advanced menu, circular sliders are available to adjust the rotation of the ROI along the three axes X, Y, and Z.

Reset: The button ↻ "Reset" does not reinitialize the values for position, dimension and rotation, but clears the display and the ROI becomes unusable until the values are actively changed again (via the sliders or by clicking on the screen).

A note "Please set the ROI values" is displayed in red. The values will also be displayed in red until an active change is made to re-enable the ROI.
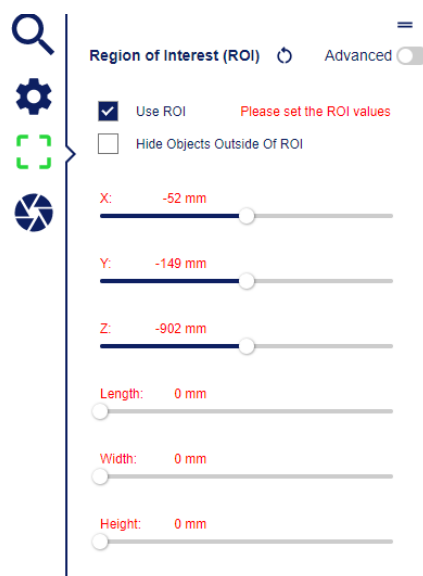
After resetting, the ROI values are displayed in red and the ROI is not active until a change is made.

10.6.2.3 DIRECT SELECTION ON THE IMAGE

When the ROI menu is open, you can also set the ROI points with the mouse.

If no ROI points are displayed (at the beginning or after using the Reset ROI button), you can click on a location to define the central position of the ROI.

This creates an ROI with a standard size of 100 x 100 x 100 mm at the corresponding 3D position.

You can then move the ROI with the middle handle (cross in the circle) and resize it with the side handles (circles)
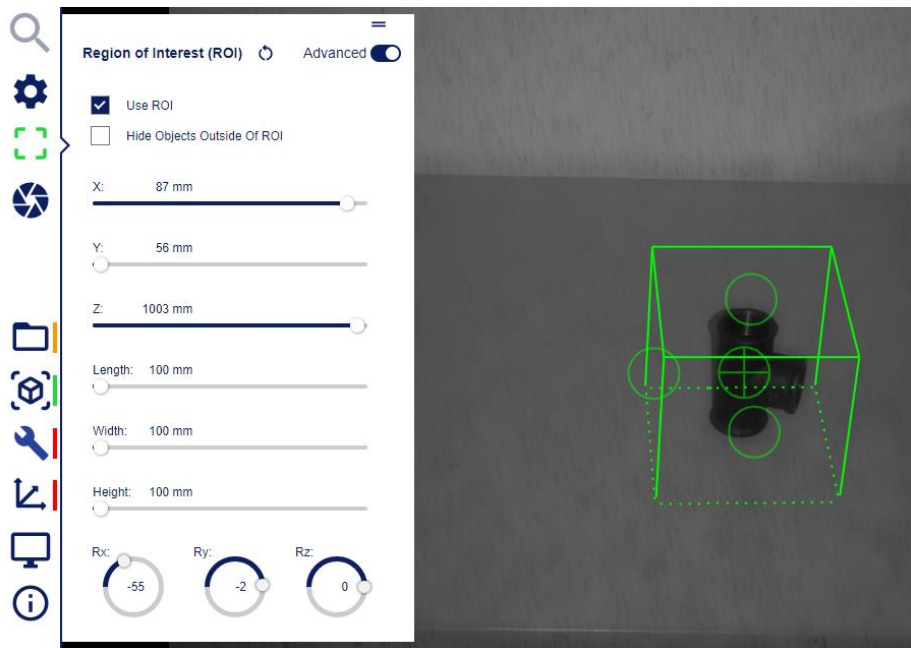
A selected ROI is displayed.
The cross inscribed in a circle is the handle for moving the ROI position.

The three simple circles on the sides allow you to change the length, width and height of the ROI accordingly.

⚠ When setting the ROI with the mouse, the position of the ROI cannot be set at a point for which there is no 3D data.

10.6.3 SEARCH PARAMETERS

The search parameters are set in the search parameters menu.

There are two versions of this menu: the basic and the advanced version. The advanced version can be switched on and off with the "Advanced" button in the upper right corner of the menu (see below. 10.4 Advanced menus).

The parameters can be reset to the default values with the reset key (see below. 10.6.3.17 Reset to default values), and can be loaded or saved with the folder key (see below 10.6.3.18 Load / Save parameters).
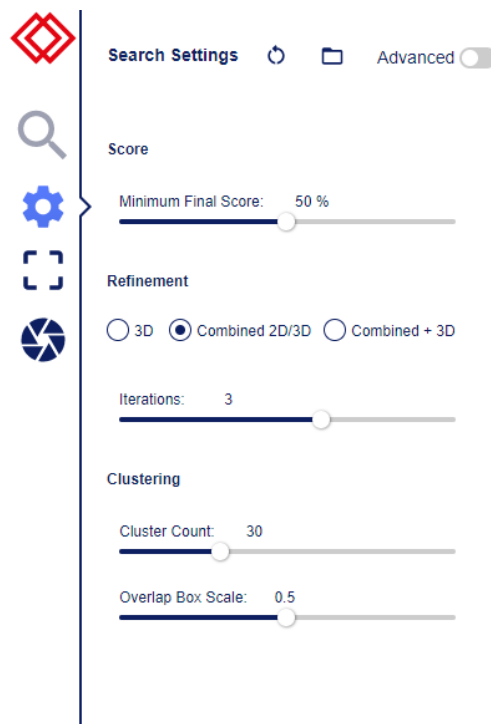
Search parameters Basic menu

The basic menu search parameters are:

10.6.3.1 MINIMUM FINAL SCORE

Threshold the results to minimum 3D point coverage after refinement. 2D refinement combines the evaluation from 2D features and 3D data coverage evaluations.

10.6.3.2 REFINEMENT

There are three options: 3D, 2D/3D combined, or Combined + 3D

- With the "3D" option, a 3D refinement of the results is performed.
- The Combined 2D/3D option performs a series of 2D/3D refinement iterations
- The Combined + 3D option performs a series of 2D/3D refinement iterations followed by an additional 3D refinement.

10.6.3.3 ITERATIONS

Number of iterations of 2d/3d combined refinement steps.

10.6.3.4 CLUSTER COUNT

Limits the number of parts found by refining only a limited number of hypothesis clusters with the best results.

10.6.3.5 OVERLAP BOX SCALE

After the combined refinement, overlapping detections are removed by applying an additional clustering to the results of the combined refinement. The density of these clusters is determined by the parameter "Overlap Box Scale". The smaller the size of the box, the more overlaps between the returned parts are allowed.
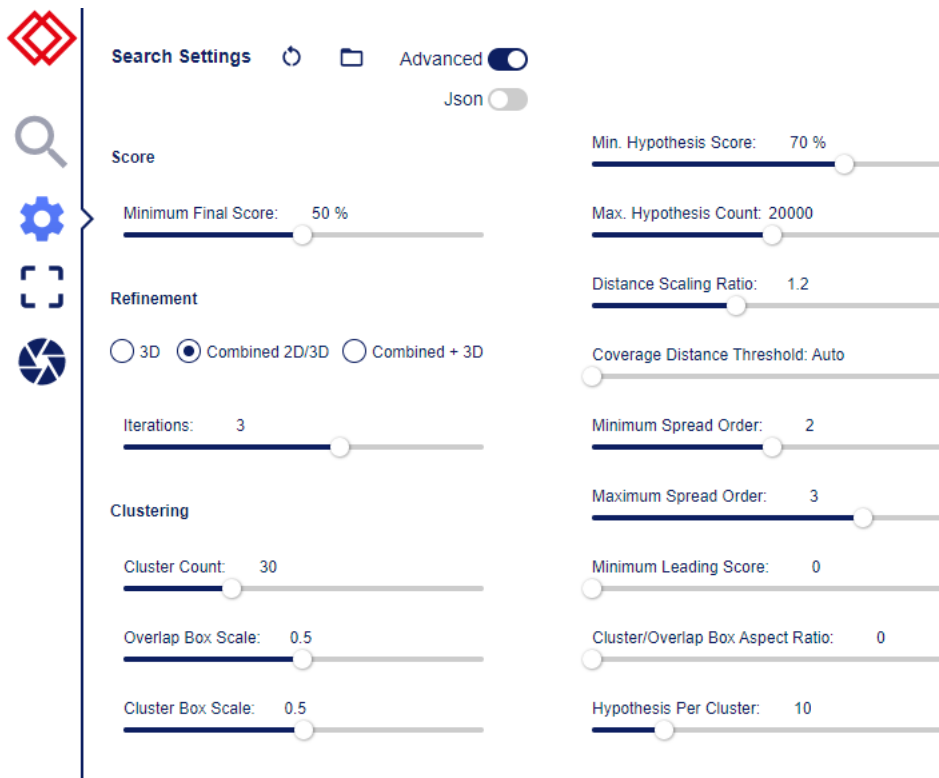
Advanced menu with additional search parameters

The additional extended parameter in the **left** column is:

10.6.3.6   CLUSTER BOX SCALE

The hypotheses are clustered so that their distribution over the 3D scene becomes more uniform. The density of the clusters is controlled by the parameter "*Cluster Box Scale*".

The extended parameters in the **right** column are:

10.6.3.7   MIN. HYPOTHESIS SCORE

Hypothesis Score. Specifies the minimum score of a pose hypothesis to be used for refinement.

10.6.3.8   MAX. HYPOTHESIS COUNT

Hypothesis Count. Limits the number of initial pose hypotheses considered for refinement to the n best hypotheses found. Limiting the hypothesis count reduces the runtime of the search. If you allow more hypotheses to be tested, parts will be found at more locations in the point cloud.

10.6.3.9 DISTANCE SCALING RATIO

Distance scaling ratio. Specifies a scaling factor between different search distances for the search.

Using a smaller scaling factor can increase the detection rate when searching over a large range of distances at the expense of runtime.

10.6.3.10 COVERAGE DISTANCE THRESHOLD

Threshold for the coverage score. The largest distance between an object point and a measurement point so that the object points are considered to match (cover).

10.6.3.11 MINIMUM SPREAD ORDER

Final block size of the 2D refinement. The underlying LineMod algorithm refines its feature search with pixel blocks whose size is 2 to the power of x, where x is the spread order defined here. The LineMod algorithm performs a multi-level refinement starting with a block of size 2^max and ending with a block of size 2^min, where max and min are the maximum and minimum spread orders.

10.6.3.12 MAXIMUM SPREAD ORDER

Start block size of the LineMod 2D refinement.

10.6.3.13 MINIMUM LEADING SCORE

Specifies the minimum value of the maximum score of a hypothesis in a cluster. A cluster is accepted if the maximum hypothesis/score value in the cluster exceeds this value.

10.6.3.14 CLUSTER/OVERLA ASPECT RATIO

The hypotheses are clustered so that their distribution over the 3D scene becomes more uniform. Removing overlapping hypotheses is similar to clustering hypotheses. This parameter limits the aspect ratio of the surrounding area of a cluster in a spatial region to a minimum value.

10.6.3.15 HYPOTHESIS PER CLUSTER

Specifies how many hypotheses in each cluster should be refined separately. If you choose a larger number for the cluster size, more different part poses will be evaluated at each location, which helps to correctly distinguish similar part poses.

10.6.3.16   JSON SEARCH PARAMETERS

Parameters can also be entered as json text

In advanced mode ("Advanced" set to ON), the search parameters can also be entered as Json (JavaScript Object Notation, a compact text data format) in a text area that can be resized. This is especially useful for importing/exporting parameters between Mikado Find and NxView or NxTreeEdit. The Json parameters option can be activated with the Json button, which is only visible in advanced parameter mode.

If you turn on the Json mode, the current parameters will be converted as Json text so that you can edit them in the text field.

*The search icon displays "Json",
if the json parameters are set*

⚠ If you copy Json parameters into the input field, make sure that the ModelID matches the currently generated model ID (see below. 10.5.3.10 Current model information).

⚠ If an ROI is defined in Json, it will be used instead of the ROI defined in the ROI menu. The displayed ROI lines correspond to the Json ROI, and the sliders in the ROI menu are disabled with the indication "Json ROI in use".
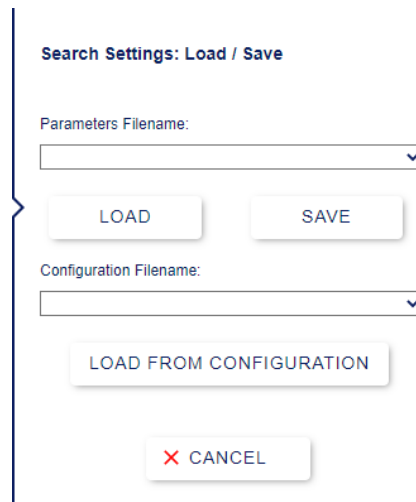
ROI menu is disabled when Json ROI is used

10.6.3.17   RESET TO DEFAULT VALUES

When you click the ↻ reset icon, all search parameter values are immediately reset to the default values.

10.6.3.18   LOAD / SAVE PARAMETERS

The button "Folder 🗀 opens a dialog for loading/saving.

Search Settings: Load / Save

Parameters Filename:

[▾]

[LOAD]   [SAVE]

Configuration Filename:

[▾]

[LOAD FROM CONFIGURATION]

[✕ CANCEL]

Load/Save dialog

From this dialog it is possible to save or load parameter files or load from configurations.

Saving individual parameter files

Enter a new name in the Parameter Filename field and click *Save*

If the file already exists, a dialog will ask you if you want to overwrite the file.

Overwrite Parameters File

Are you sure you want to **overwrite**

testparams2 ?

✓ OK   ✕ CANCEL

Override permission dialog
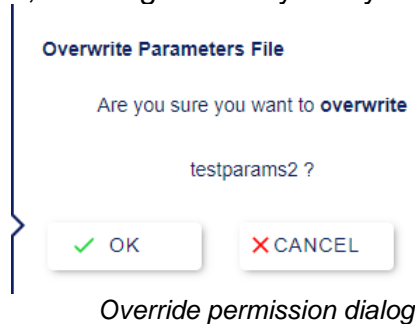
The files are stored on a server in the *Data__parameters* folder.

Loading individual parameter files

Previously saved files are displayed in the drop-down selection. Select the desired file and click "*Load*" to update the search parameters with those in the files. The number of updated parameters is displayed in the message window.

2022-07-08 15:33:49.543 : Loaded 13 Search Parameters
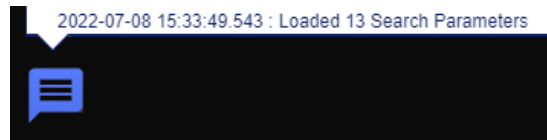
The number of loaded parameters is displayed in the message window

Loading from the configuration

It is also possible to load the search parameters contained in a previously saved configuration. Select the desired configuration from the drop-down list for the name of the configuration file and click "*Load From Configuration*".

The search parameters will now match those saved in the configuration without loading other parts (camera, model parameters, settings) of the configuration. The number of updated parameters is displayed in the message window.

10.6.4 SEARCH

Once the search is ready and its parameters are set, clicking the search button starts the 3D object detection. A pop-up window shows the progress of the search.

10.6.5 SEARCH RESULTS

The search results are highlighted in the image in a color from yellow to green relative to the quality of detection (yellow for low quality, green for high quality).

The result of the search is also shown in detail in the logged messages (see below. 10.7.1 Messages).

If you click on one of the selected objects on the screen, a pull-up popup appears with the specific detection results of the object: detection quality, position relative to the reference point in mm and orientation, depending on the selected rotation mode (see below. 13.3 Rotation modes).
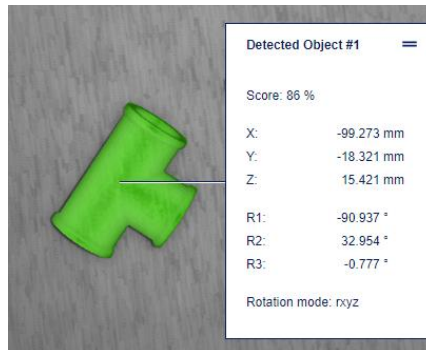
Specific detection results of the selected object
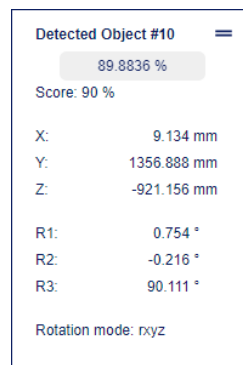
If you hold the mouse pointer over the quality score of the result, the non-rounded value is displayed

The results are also available via the OPC UA server.

10.7 STATUS BAR

The status bar consists of three parts: the left, middle and right part. The left part contains the Camera menu button (see below. 10.3 Camera menu) and the Messages button.

10.7.1 MESSAGES

Messages, such as which commands are started, their result or errors, are logged in the message window. You can open the message window to browse through the logged messages by clicking on the message icon.
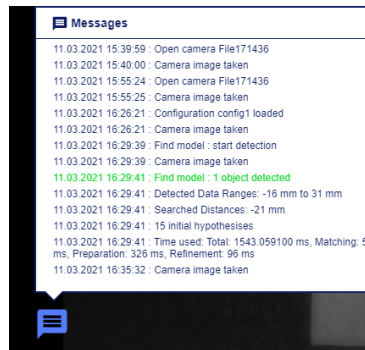
The message window

10.7.2 DISPLAY MODES

Normal, texture and result modes

The middle part has three buttons for selecting different display modes:

- The Normal mode shows the rectified left stereo image that is most suitable for the human eye.
- Texture mode shows the calculated textured left stereo image used for the search.
- The Results mode shows or hides the results as a highlight above the image.

10.7.3 SELECTED MODEL

The right part of the status bar shows which 3D model is selected.
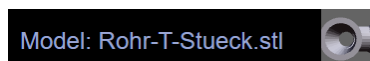
Name and mini-view of the 3D model

Clicking on the mini 3D view of the model opens a larger 3D examination window.

11 OPC-UA

The Mikado Find application and the programmable logic controller (PLC) communicate via OPC UA variables. Mikado Find acts as OPC UA server for this and follows the scheme as defined in the interface description file *Mikado_Find_OPCUA_v1_3.xml.* The file is located in the folder "Opc-ua-xml" in the application folder.

The respective object information such as NodeIds can be found there. By default, the server accepts on port 4840 under the IP of the Mikado Find application.

11.1 OPC-UA UPDATE RATE

The server has a minimum refresh rate of 50 ms. This value can be changed via the console with the command "opcms" (see below. 9.1 Configuration).

11.2 OPC-UA COMMANDS

11.2.1 COMMAND 1, LOAD CONFIGURATION

The command 1 (in the "CommandNo" field) loads all parameters (camera name and parameters, model name and parameters, ROI) previously saved in the configuration file named by the "ConfigName" field.

11.2.2 COMMAND 2, SAVE CONFIGURATION

The command 2 saves all parameters (camera name and parameters, model name and parameters, ROI) in the configuration file named by the "ConfigName" field.

11.2.3 COMMAND 3, MODEL L GENERATE

The command 3 creates a model to be used in the recognition. Please select a 3D model beforehand with the command 7 "Open Model".

11.2.4 COMMAND 4, IMAGE CAPTURE AND RECOGNITION

Command 4 captures an image and then performs a detection. This command sets the value of ResultNo and fills the result array.

The results of this command are

- ResultNo

- Results.50 [ ] : A static array with 50 results.

11.2.5 COMMAND 5, ACHIEVING RESULTS

Command 5 can be used to retrieve all found results from the Mikado Find one after the other.

First the index to be queried must be entered in "GetResultIndex". With command 5 ("GetResult") Mikado Find describes the fields:

- ResultQuality (Result Quality [%])

- Result.X (position values [mm])

- Result.Y (position values [mm])

- Result.Z (position values [mm])

- Result.R1 (rotation values [°])

- Result.R2 (rotation values [°])

- Result.R3 (rotation values [°])

- Result.R4 (rotation values [°])

11.2.6 COMMAND 6, OPEN CAMERA

Command 6 opens the target camera. The name of the target camera must be entered in the "CameraName" field.

11.2.7 COMMAND 7, MODEL LOADING

Command 7 is used to select the 3D model file. The "STLName" field specifies the name of the target file, with extension (.stl or .ply). Currently, only **non-accented characters** are allowed in the name.

11.2.8 COMMAND 8, CALIBRATION EN FROM PATTERN

The PLC sends this command 8 to start the calibration of the origin point. A calibration plate must be placed at the position of the target origin.

11.2.9 COMMAND 9, IMAGE CAPTURE

Command 9 is used to capture an image.

11.2.10   COMMAND 10, IMAGE CAPTURE AND RECOGNITION

This command is used to perform a detection. This is an alternative version of command 4, where no images are taken. Instead, the current image is used, which can be updated with command 9. This command sets the value of ResultNo and fills the result array.

The results of this command are

- ResultNo

- Results.50 [ ]

11.3 OPC-UA INTERFACE

11.3.1 OPC-UA OUTPUTS (MIKADO FIND TO PLC)

Outputs	Type	Description
ReadyBit	Boolean	Mikado Find ready? Condition for commands
BusyBit	Boolean	true = busy false = ready/ready New order can be accepted
ErrorNo	Int	Error messages
ResultNo	Int	Number of found parts is returned [0 = no part found]
ResultQuality	Double	Measurement quality of the current result
Result.X	Double	X of the current result [mm]
Result.Y	Double	Y of the current result [mm]
Result.Z	Double	Z of the current result [mm]
Result.R1	Double	Rotation value 1 (according to specified RotationMode) [°].
Result.R2	Double	Rotation value 2 (according to specified RotationMode) [°].
Result.R3	Double	Rotation value 3 (according to specified RotationMode) [°].
Result.R4	Double	Rotation value 4 (according to specified RotationMode) [°].
Results.50 [ ]	[50] [7] Double	Static 2D array with 50 first results: | N finds | x 8 entries Q - result quality [%] X, Y, Z - position values [mm] R1, R2, R3, R4 - rotation values [°] (see below). 13.3 Rotation modes)

11.3.1 OPC-UA INPUTS (PLC TO MIKADO FIND)

Inputs	Type	Description
ConfigName	String	Configuration name, used to save and load the application parameters
STLName	String	Name of the 3D model (.stl or .ply) to be used for generation. Used for command 7. Currently only non-accented characters are allowed in the name
CameraName	String	Name of the camera to be used. Used for command 6
RotationMode	String	Rotation convention to use for pose and result. See Table 1, Available modes (see below). 13.3 Rotation modes)
GetResultIndex	Int	Index of the result to be read
CommandNo	Int	0 = Inactivity 1 = Load configuration ConfigName 2 = Save configuration ConfigName 3 = Generate model 4 = Image acquisition and recognition 5 = Get result at GetResultIndex 6 = Open camera CameraName 7 = Open STL STLName 8 = Calibrate origin with pattern 9 = Image acquisition 10 = Detection (Without image acquisition)

12 TCP/IP

With MIKADO Find the customer has the possibility to pass information of the system to external systems via a TCP/IP. By default, the server accepts on port 20000 under the IP of Mikado Find application

Received commands must follow a format separated by _. Replies are encoded in a format with a fixed data length.

12.1 COMMANDS FORMATTING

The commands consist of a command name in capital letters and optionally a parameter. The command name and the parameter are separated by the _ character.

Example: SAVE_myconfig

12.2 RESPONSE FORMATTING

The responses are encoded in a format with a fixed data length:

- No separation between variables

- Ok Bit is only one digit: 0 or 1

- All numbers are integer numbers with exactly 8 digits, padded with 0 if necessary. Floating numbers are multiplied by 1000 and converted into integer numbers with exactly 8 digits.

- No termination mark.

In this document, we write down the 8-character integer variables as [name], where the name describes the use of the variable.

Example:

A response consisting of three variables: OK bit, result number and result quality.

we note it here as [Ok][Outcome number][Outcome quality].

is coded as

10000000200000874

This would mean: OK, 2 results, first result quality = 87.4%.

12.3 COMMAND LIST

The following commands respond with [Ok][Error code].

The list of error codes can be found in 13.2 Error codes.

Command:	Parameters:	Description:
LOAD	Configuration file name	Load parameters from named configuration file. Example: LOAD_project1 Response: [Ok][Error code]
SAVE	Configuration file name	Save parameters from the named configuration file. Example: SAVE_project1 Response: [Ok][Error code]
CAM	Camera material	Open camera <serial>. Example: CAM_17234 Response: [Ok][Error code]
MODEL	Model file	Select STL or PLY model file (with .stl or .ply extension). Currently, only **non-accented characters** are allowed in the name. Example: MODEL_SimpleCylinder Response: [Ok][Error code]
RED	Rotation mode	Set the rotation mode of the results. See Table 1: Available modes (see below). 13.3 Rotation modes): Example: ROT_rxyx Response: [Ok][Error Code]
SNAP		Recording a new image Response: [Ok][Error code]

The following commands respond with:

[Ok][Fehlercode][Ergebnisse-Anzahl][Ergebnisqualität][X][Y][Z][R1][R2][R3]

where [X], [Y], [Z] are the coordinates of the object and [R1], [R2], [R3] its rotation according to the selected rotation mode.

[Results count] indicates the total number of results found.

Command:	Parameters:	Description:
DETECT		This command performs a **detection** (**without recording**). Returns the first result directly. Example: DETECT Response: [Ok][Error number][Results count] [Ergebnisqualität][X][Y][Z][R1][R2][R3][R4]]
SNAPDETECT		This command performs a **capture** and a **detection.** Returns directly the first result. Example: SNAPDETECT Response: [Ok][Error number][Results count] [Ergebnisqualität][X][Y][Z][R1][R2][R3][R4]
GETRESULT	Index n of the result	Returns the results for the nth object found Example: GETRESULT_3 Response: [Ok][Fehlercode][Ergebnisse-Anzahl][Ergebnisqualität][X][Y][Z][R1][R2][R3][R4]

13 APPENDIX

13.1 PARAMETERS.JSON VALUE LIST

List of values in the parameters.json file (saved as part of the configuration zip) and their GUI field names.

Parameters of the application:

GUI field name	Type	Menu	parameters.json
Selected model	Text	Load 3D model	stlmodel
Saturation	Double	GUI settings	saturation
Camera Serial	Text	Camera settings	camera

ROI parameters:

GUI field name	Type	Menu	parameters .json
ROI Advanced switch	Boolean	ROI	advancedRoiMode
Use Roi	Boolean	ROI	roiSet
X	Double	ROI	x
Y	Double	ROI	y
Z	Double	ROI	z
Length	Double	ROI	length
Width	Double	ROI	width
Height	Double	ROI	height
R1	Double	ROI	r1
R2	Double	ROI	r2
R3	Double	ROI	r3
R4	Double	ROI	r4
	Boolean	ROI	show1
	Boolean	ROI	show2

Workspace parameters:

GUI field name	Type	Menu	parameters .json
Rotation Mode	Text	Workspace	rotationMode
Sort Results 1	Text	Workspace	resultsSort1
Sort Results 2	Text	Workspace	resultsSort2
Residuals Tolerance	Double	Workspace	rtolerance

Search parameters:

GUI field name	Type	Menu	parameters .json
Search Advanced switch	Boolean	Search parameters	advancedSearchMode
Search Json switch	Boolean	Search parameters	useJsonDetectParameters
Search Json parameters	Boolean	Search parameters	detectParametersAsJson
Min. hypothesis score	Double	Search parameters	hypothesisScore
Max. Hypothesis Count	Integer	Search parameters	hypothesisCount
Distance Scaling Ratio	Double	Search parameters	distanceRatio
Coverage Distance Threshold	Double	Search parameters	scoreDistanceThreshold
Cluster Count	Integer	Search parameters	clusterCount
Cluster/Overlap Aspect Ratio	Double	Search parameters	minAspectRatio
Cluster Box Scale	Double	Search parameters	clusterBoxSize
Overlap Box Scale	Double	Search parameters	overlapBoundingBoxScale
Hypothesis Per Cluster	Double	Search parameters	maxClusterSize
Minimum spread order	Integer	Search parameters	minimumSpreadOrder
Maximum spread order	Integer	Search parameters	maximumSpreadOrder
Minimum Leading Score	Double	Search parameters	minimumLeadingScore
Refinement	Boolean	Search parameters	refinement
2D/3D Iterations	Integer	Search parameters	ilmIterations
Min. Final Score	Double	Search parameters	minFinalCoverage

Model parameters:

GUI field name	Type	Menu	parameters .json
Selected model	Text	Load 3D model	stlmodel
Model Advanced switch	Boolean	Generate search model	advancedModelMode
Model Json switch	Boolean	Generate search model	useJsonModelParameters
Model Json parameters	Boolean	Generate search model	modelParametersAsJson
Viewpoints all	Boolean	Generate search model	viewpoint_all
Viewpoints Only Selected	Boolean	Generate search model	viewpoint_select
	Integer	Generate search model	viewpoint_select_index
Viewpoints Rotation Symetric	Boolean	Generate search model	viewpoint_sym
	Integer	Generate search model	viewpoint_sym_index
Only Selected Angle	Integer	Generate search model	onlyViewpointAngle
Angular Discretization	Integer	Generate search model	angle
In-Plane Angular Discretization	Integer	Generate search model	inPlanarAngle
Roi Distance	Boolean	Generate search model	useRoi
Manual Distance	Boolean	Generate search model	useManualDistance
Manual Distance slider	Double	Generate search model	manualDistance
Relative Sampling Distance	Integer	Generate search model	relSamplingDist
Subsampling Factor	Double	Generate search model	subsampling
Use Model Texture	Boolean	Generate search model	useTexture
Align Rotation	Boolean	Generate search model	useAlignment
Model Symmetry Axis	Text	Generate search model	modelSymmetryAxis
Model Axis to Align	Text	Generate search model	modelAxisToAlign
World Target Direction	Text	Generate search model	worldAxisToAlignTo
x	Double	Model view	grippingPointX
y	Double	Model view	grippingPointY
z	Double	Model view	grippingPointZ
Rx	Double	Model view	grippingPointR1
Ry	Double	Model view	grippingPointR2
Rz	Double	Model view	grippingPointR3

13.2 ERROR CODES

List of error codes sent to the PLC.

Error codes	Description
0	No error, successful
1	Unknown system error
2	Camera cannot be opened
3	No stereo camera connected
4	Camera error
5	Search failed
6	Search has no associated model
7	Generate model failed
8	Set 3D reference point failed
9	Configuration could not be loaded
10	Configuration could not be saved
11	Command not implemented
12	Wrong index to retrieve the results
13	STL file not found
14	Incompatible model version
15	Invalid TCP/IP command or parameter
16	Codemeter license not found
17	Wrong Ensenso SDK version
18	Non-ascii characters in the name

13.3 ROTATION MODES

The rotation of the detected objects is given in the form of three or four values.The available rotation modes can be selected as *Euler Angle* (3 values), *AngleAxis* (4 values: angle and x,y,z axis values), *Quaternion* (4 values) or *Rodrigues* (3 values).

The rotation mode as Euler angles defines the order of the axes (x, y or z) around which the three returned Euler angles are to be applied. The rotation mode is thus defined with 4 characters.
The first can be either **s** (extrinsic, global system) or **r** (intrinsic, rotated system). The next three define the order of the axes used, with 12 possibilities to choose from.

The following table lists the preset rotation modes, named after the robot manufacturer with which they are compatible.

Rotation mode:	R1:	R2:	R3:	R4:
(Euler, 24 modes)				
'rxyz (default mode) *Staeubli Mecademic*	X	Y`	Z``	
'sxyz *Fanuc Fruitcore Motoman Mitsubishi Techman*	X	Y	Z	
'rzyx *ABB Epson Nachi Kuka*	Z	Y`	X``	
'rzyz *Adept Comau Kawasaki Omron*	Z	Y`	Z``	
... (20 more)				
angleaxis	Angle	X	Y	Z
quaternion	Q0	Q1	Q2	Q3
rodrigues	V1	V2	V3	

Table 1: Available modes

13.4 DEBUG LEVEL

The debug level controls whether additional debug traces are added to the log messages and log file Logs/Mikado_Find-xxx.log.

There are three debug levels:

1: Action tracking (detect, generate, calibrate) and camera recording.

2: Tracking actions that call the camera recording also in sequence loops.

3: Trace flow loops, step by step

Communication between the application and the web GUI is also logged for all debug levels above 0. The logged messages appear in the local browser console.

Additionally, it is possible to enable the debug trace of the web GUI by adding '*?debug=1*' to the browser URL.

If the debug level is above 0, the NxLib exceptions are also logged in a separate .nxlog file.

13.5 THIRD PARTY LICENSES

C++ foreign component	Version	License
Asio	1.12.1	Boost Software License
Own	3.3.9	Mozilla Public License v2.0 (MPLv2)
JSON for Modern C++	3.9.1	MIT License
miniz.c	1.15	Public domain
miniz-cpp	18 Apr 2017	MIT License
open62541	1.3	Mozilla Public License v2.0 (MPLv2)
opencv	4.4.0	3-clause BSD License
openSSL	1.0.2l	OpenSSL License and SSLeay license
Simple Web Server	2.1.1	MIT License
Simple Websocket Server	2.0.0-RC3	MIT License

JS foreign component	Version	License
eslint	8.10.0	MIT License
three.js	0.138.3	MIT License
typescript	4.6.2	Apache License Version 2.0
vue.js	2.6.14	MIT License
vue-property-decorator	9.1.2	MIT License
vue-round-slider	1.0.1	MIT License
vue-slider-component	3.2.15	MIT License
vue-3d-model	1.3.1	MIT License
vuex	3.6.2	MIT License
vuex-class	0.3.2	MIT License
vuex-module-decorators	1.2.0	MIT License

JS foreign component	Version	License	JS foreign component	Version	License	JS foreign component	Version	License
sindresorhus/is	4.6.0	WITH	got	12.0.1	WITH	process-nextick-args	2.0.1	WITH
szmarczak/http-timer	5.0.1	WITH	got	5.6.0	WITH	protocols	1.4.8	WITH
types/cacheable-request	6.0.2	WITH	graceful-fs	4.2.9	ISC	pump	3.0.0	WITH
types/eslint-scope	3.7.3	WITH	gry	5.0.8	WITH	punycode	2.1.1	WITH
types/eslint	8.4.1	WITH	has-flag	4.0.0	WITH	quick-lru	5.1.1	WITH
types/estree	0.0.51	WITH	has	1.0.3	WITH	r-json	1.2.10	WITH
types/http-cache-semantics	4.0.1	WITH	hosted-git-info	2.8.9	ISC	r-package-json	1.0.9	WITH
types/json-schema	7.0.9	WITH	http-cache-semantics	4.1.0	BSD-2-Clause	randombytes	2.1.0	WITH
types/keyv	3.1.2	WITH	http2-wrapper	2.1.10	WITH	rc	1.2.8	WITH
types/node	15.0.3	WITH	inherits	2.0.4	ISC	read-all-stream	3.1.0	WITH
types/responselike	1.0.0	WITH	ini	1.3.8	ISC	readable-stream	2.3.7	WITH
webassemblyjs	1.11.1	WITH	is-arrayish	0.2.1	WITH	registry-auth-token	3.4.0	WITH
xtuc/ieee754	1.2.0	BSD-3-Clause	is-core modules	2.4.0	WITH	registry-url	3.1.0	WITH
xtuc/long	4.2.2	Apache 2.0	is-plain-obj	1.1.0	WITH	resolve-alpn	1.2.1	WITH
abs	1.3.14	WITH	is-redirect	1.0.0	WITH	resolve	1.20.0	WITH
acorn-import-assertions	1.8.0	WITH	is-retry-allowed	1.2.0	WITH	responselike	2.0.0	WITH
acorn	8.7.0	WITH	is-ssh	1.3.3	WITH	round-slider	1.6.1	WITH
ajv-keywords	3.5.2	WITH	is-stream	1.1.0	WITH	safe-buffer	5.1.2	WITH
ajv	6.12.6	WITH	isarray	1.0.0	WITH	schema-utils	3.1.1	WITH
browserslist	4.20.0	WITH	iterate-object	1.3.4	WITH	semver	5.7.1	ISC
buffer-from	1.1.1	WITH	jest-worker	27.5.1	WITH	serialize-javascript	6.0.0	BSD-3-Clause
cacheable-lookup	6.0.4	WITH	jquery	3.6.0	WITH	sliced	1.0.1	WITH
cacheable-request	7.0.2	WITH	json-buffer	3.0.1	WITH	source-map-support	0.5.21	WITH
capture-stack-trace	1.0.1	WITH	json-parse-better-errors	1.0.2	WITH	source-map	0.6.1	BSD-3-Clause
chrome-trace-event	1.0.3	WITH	json-schema-traverse	0.4.1	WITH	source-map	0.7.3	BSD-3-Clause
clone-response	1.0.2	WITH	keyv	4.0.3	WITH	string_decoder	1.1.1	WITH
commander	2.20.3	WITH	limit-it	3.2.10	WITH	strip-json-comments	2.0.1	WITH
core-js	3.21.1	WITH	loader-runner	4.2.0	WITH	supports-color	8.1.1	WITH
core-util-is	1.0.2	WITH	lowercase-keys	1.0.1	WITH	tapable	2.2.1	WITH
create-error-class	3.0.2	WITH	lowercase-keys	2.0.0	WITH	terser-webpack-plugin	5.3.1	WITH
decompress-response	6.0.0	WITH	lowercase-keys	3.0.0	WITH	electron-to-chromium	1.4.77	ISC
deep-extend	0.6.0	WITH	merge-stream	2.0.0	WITH	end-of-stream	1.4.4	WITH
defer-to-connect	2.0.1	WITH	mime-db	1.51.0	WITH	enhanced-resolve	5.9.2	WITH
glob-to-regexp	0.4.1	BSD-2-Clause	prepend-http	1.0.4	WITH	err	1.1.1	WITH
deffy	2.2.4	WITH	mime-types	2.1.34	WITH	error-ex	1.3.2	WITH

duplexer2	0.1.4	BSD-3-Clause	mimic-response	1.0.1	WITH	terser	5.12.0	BSD-2-Clause
es-module-lexer	0.9.3	WITH	neo-async	2.6.2	WITH	pkg.json	2.0.8	WITH
escalade	3.1.1	WITH	node-releases	2.0.2	WITH	three	0.138.3	WITH
eslint-scope	5.1.1	BSD-2-Clause	node-status-codes	1.0.0	WITH	timed-out	2.0.0	WITH
esrecurse	4.3.0	BSD-2-Clause	noop6	1.0.9	WITH	tmp	0.0.28	WITH
estraverse	4.3.0	BSD-2-Clause	normalize-package-data	2.5.0	BSD-2-Clause	typpy	2.3.13	WITH
estraverse	5.2.0	BSD-2-Clause	normalize-url	6.1.0	WITH	ul	5.2.15	WITH
events	3.3.0	WITH	oargv	3.4.10	WITH	unzip-response	1.0.2	WITH
exec-limiter	3.2.13	WITH	obj-def	1.0.9	WITH	uri-js	4.4.1	BSD-2-Clause
fast-deep-equal	3.1.3	WITH	object-assign	4.1.1	WITH	url-parse-lax	1.0.0	WITH
fast-json-stable-stringify	2.1.0	WITH	once	1.4.0	ISC	util-deprecate	1.0.2	WITH
form-data-encoder	1.7.1	WITH	one-by-one	3.2.8	WITH	validate-npm-package-license	3.0.4	Apache 2.0
function-bind	1.1.1	WITH	os-tmpdir	1.0.2	WITH	vue-class-component	7.2.6	WITH
function.name	1.0.13	WITH	p-cancelable	3.0.0	WITH	vue-i18n	8.27.1	WITH
get-stream	5.2.0	WITH	package-json-path	1.0.9	WITH	vue-property-decorator	8.5.1	WITH
get-stream	6.0.1	WITH	package-json	2.4.0	WITH	vuex-module-decorators	1.2.0	WITH
git-package-json	1.4.10	WITH	parse-json	2.2.0	WITH	watchpack	2.3.1	WITH
git-source	1.1.10	WITH	parse-url	1.3.11	WITH	webpack-sources	3.2.3	WITH
git-up	1.2.1	WITH	path-parse	1.0.7	WITH	webpack	5.70.0	WITH
git-url-parse	5.0.1	WITH	picocolors	1.0.0	ISC	wrappy	1.0.2	WITH
mimic-response	3.1.0	WITH	pinkie-promise	2.0.1	WITH			
minimist	1.2.6	WITH	pinkie	2.0.4	WITH			

The corresponding license text files are located in Mikado-Find-FOSS-Licenses.txt, in the "Licenses" folder in the application folder.

Modified open source code from open62541 1.3 and vue-3d-model is located in the "OpenSource-modifications" subfolder.

13.6 COPYRIGHT NOTICE