INSCOPER

User Guide

Imaging Software

v9.3

Table of contents

1. Imaging Software

1.1 Overview

The **Inscoper Imaging Software** (Inscoper I.S.), powered by a proprietary hardware controller, provides an integrated interface for managing optical imaging workflows.

Inscoper I.S. offers research microscopists a unified interface to manage their optical imaging systems:

- Orchestrate multi-dimensional acquisition sequences.
- Control multiple cameras and manage data streams.
- Monitor hardware status in real time.
- Manage experimental configurations via reusable projects.
- Perform basic image processing and volumetric visualization.

1.1.1 Prerequisites

An Inscoper-certified installer assembled a hardware configuration for your system using the Configurator software. Now, your system is ready to perform your acquisition workflow.

If you want to perform minor hardware modifications (switch or add objectives, cubes), you can proceed with caution following this tutorial (part 3). For any other changes, do not modify the hardware configuration on your own at the risk of breaking image acquisition functions. Please contact us if you need advice.

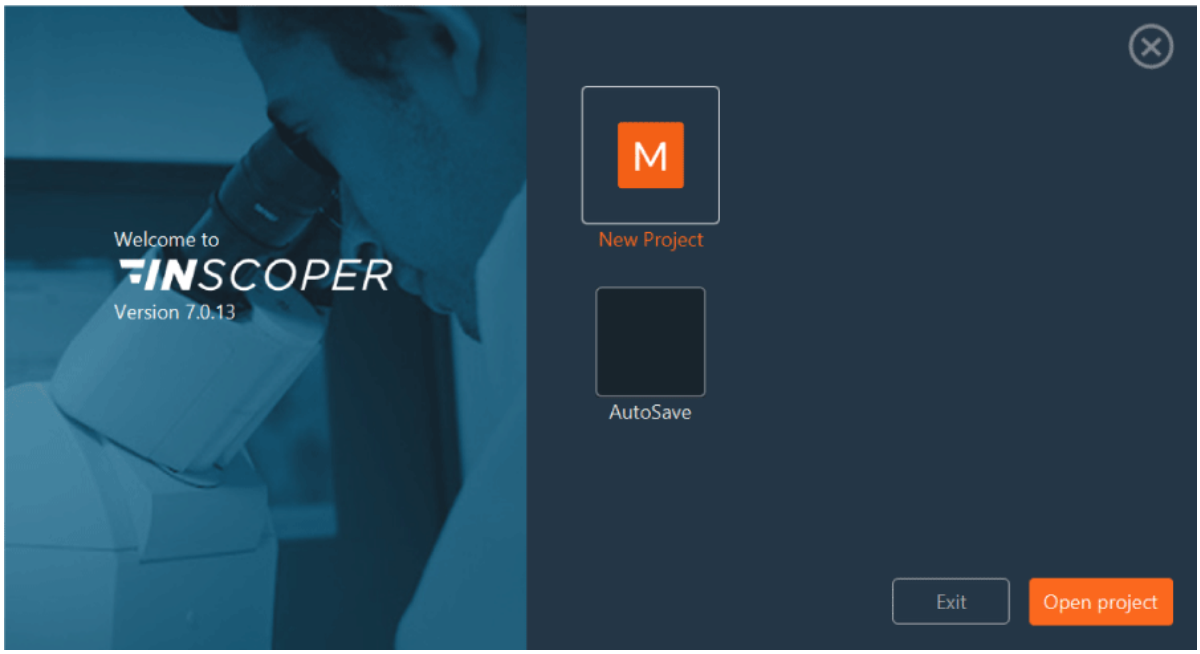
1.1.2 User Journey

The Inscoper I.S. interface follows a guided workflow that helps new users get started quickly, while remaining efficient and enjoyable for experienced users.

This journey follows three distinct phases:

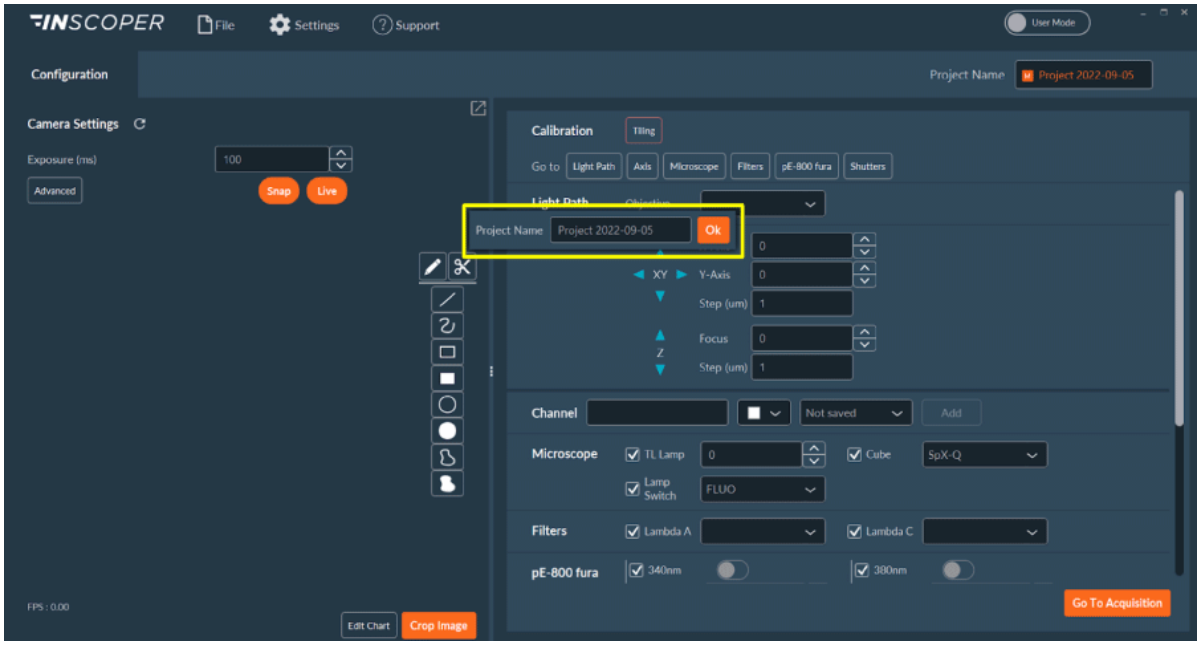
1. Establish optical pathways, configure connected hardware devices, and define parameters (Configuration).
2. Construct an acquisition protocol by configuring programmable dimensions, or design a custom pattern (Multi-dimensional Acquisition / Designer).
3. Explore the resulting images using built-in visualization tools (Visualization).

1.1.3 Getting Started



When you launch Inscoper I.S., the start window appears. The software version number is displayed, and you can choose from the following options to begin:

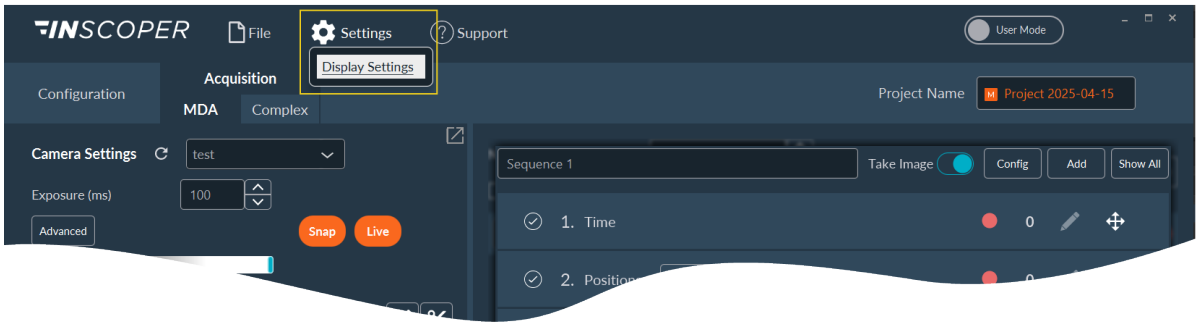
Command Button	Operational Description
New Project	Initializes a new working session.
Auto save	Recovers the most recent session state, including unsaved sequence parameters.
Open Project	Opens an existing project. Select your file (`.cbf`) from the directory tree, then click **Open**.



After you select the appropriate session setup, the software probes and initializes the connected hardware. You are then prompted to name the new project. If the name field is left blank, the system defaults to the current date, as shown in the image above.

1.2 Display Settings

Access interface preferences by navigating to **Settings** > **Display Settings**.

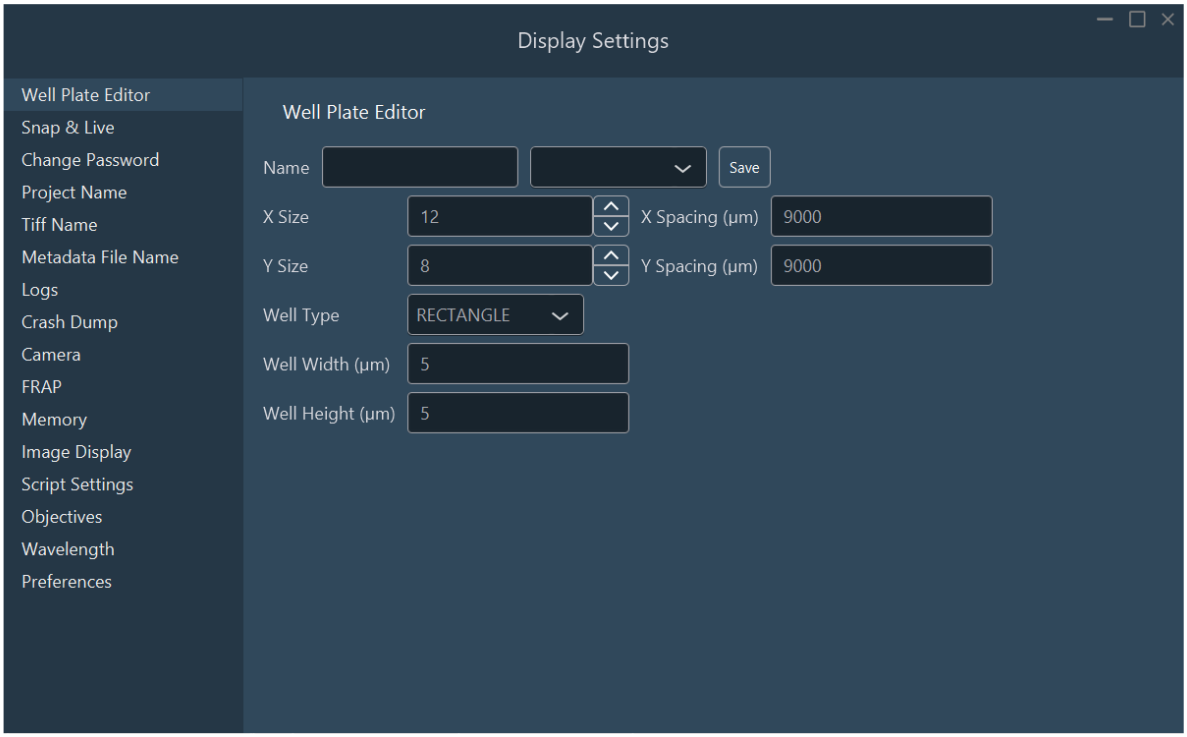


This configuration menu allows you to customize software behavior, storage paths, and interface elements.

- Well Plate Template Editor
- Snap & Live Defaults
- System Security: Change Password
- Default Project Naming
- Tiff Export Naming Conventions
- Metadata Export
- System Logs
- Diagnostic Crash Dump
- Multi-Camera Management
- FRAP Sensitivity Parameterization
- System Memory Allocation Engine
- Primary Image Display Engine
- Objectives Database
- Source Wavelength Database
- Global System Preferences

1.2.1 Well Plate Template Editor

Construct and edit multi-well plate configurations used during acquisition:

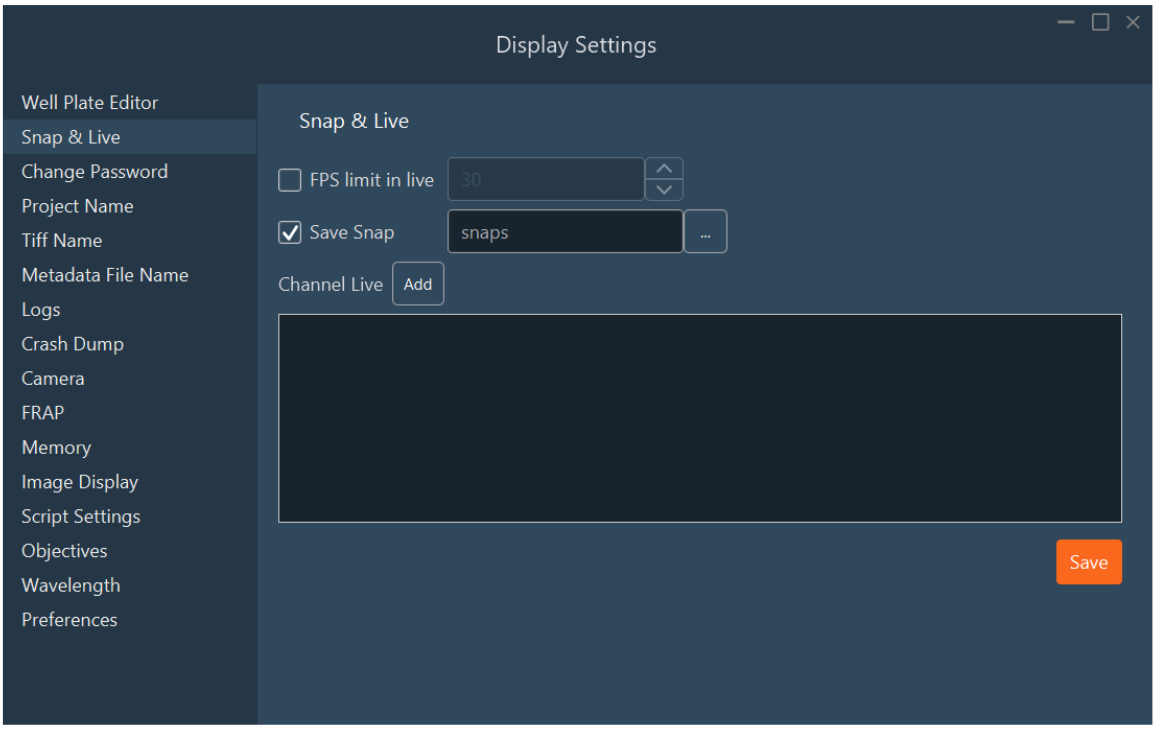


- **Name**: Assign a name to the template.
- **Edit**: Select an existing well plate configuration from the drop-down menu to modify and save its parameters.
- **Grid**: Define the number of horizontal (columns) and vertical (rows) wells, labeled X Size and Y Size respectively.
- **Spacing**: Define the distance (pitch) between the wells (in µm).
- **Type**: Select the well shape from the drop-down menu (e.g., circle, square, or rectangle).
- **Well Width/Height**: Define the internal dimensions of the individual wells (in µm).
- **Save**: Save this well plate configuration to the database.

> **Multi-Slide Engineering**
>
> Define a custom multi-slide holder by configuring individual glass slides as large rectangular "wells" within a 1×4 or 1×8 grid format.

1.2.2 Snap & Live Defaults

Configure default parameters for the Snap and Live imaging modes:

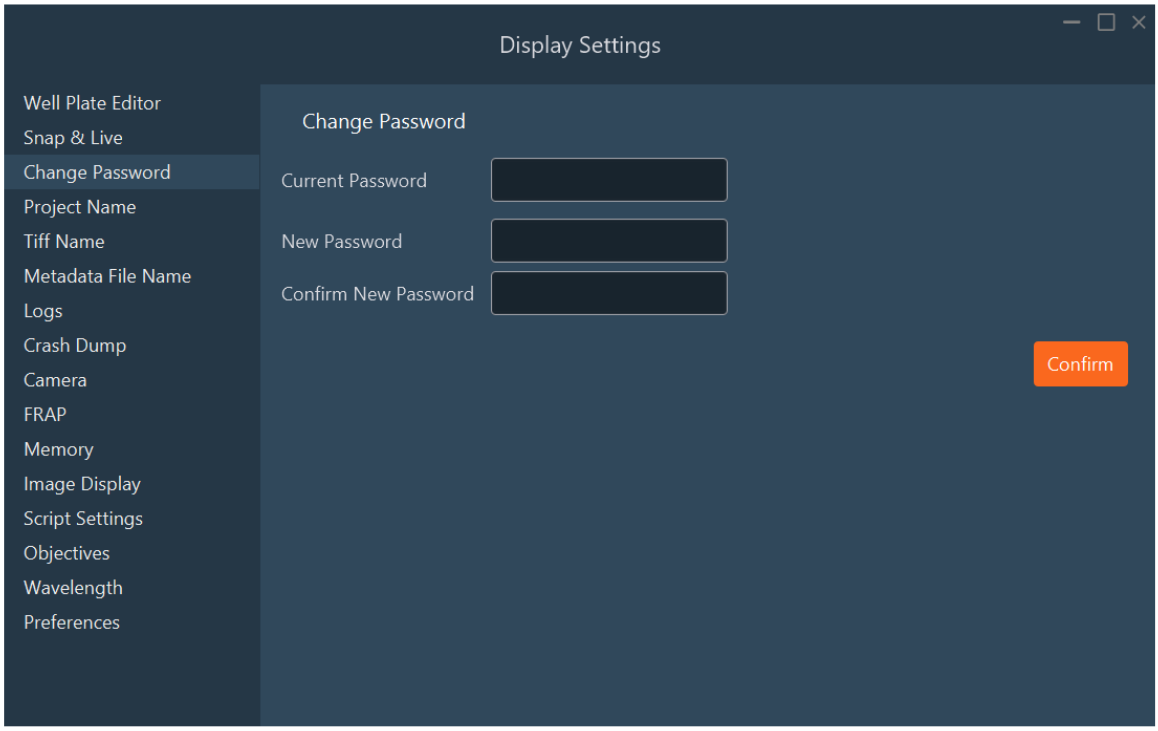


Available parameters:

- **Maximum frame rate (FPS)**: Sets an upper limit on the frame rate during Live streaming.
- **Default save directory**: Specifies the output directory for Snapshots.
- **Default startup channel**: Selects the channel that automatically engages when Live mode is toggled.

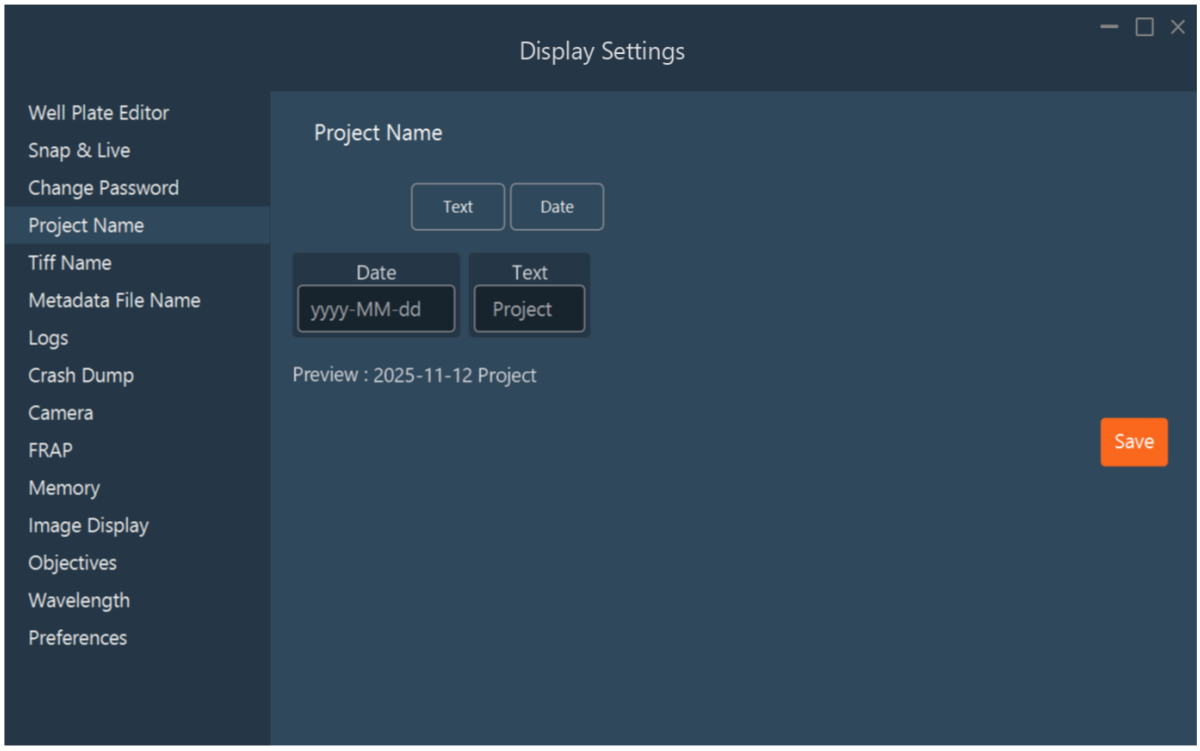
1.2.3 System Security: Change Password

Define or update the password required to access **Expert Mode**.



1.2.4 Default Project Naming

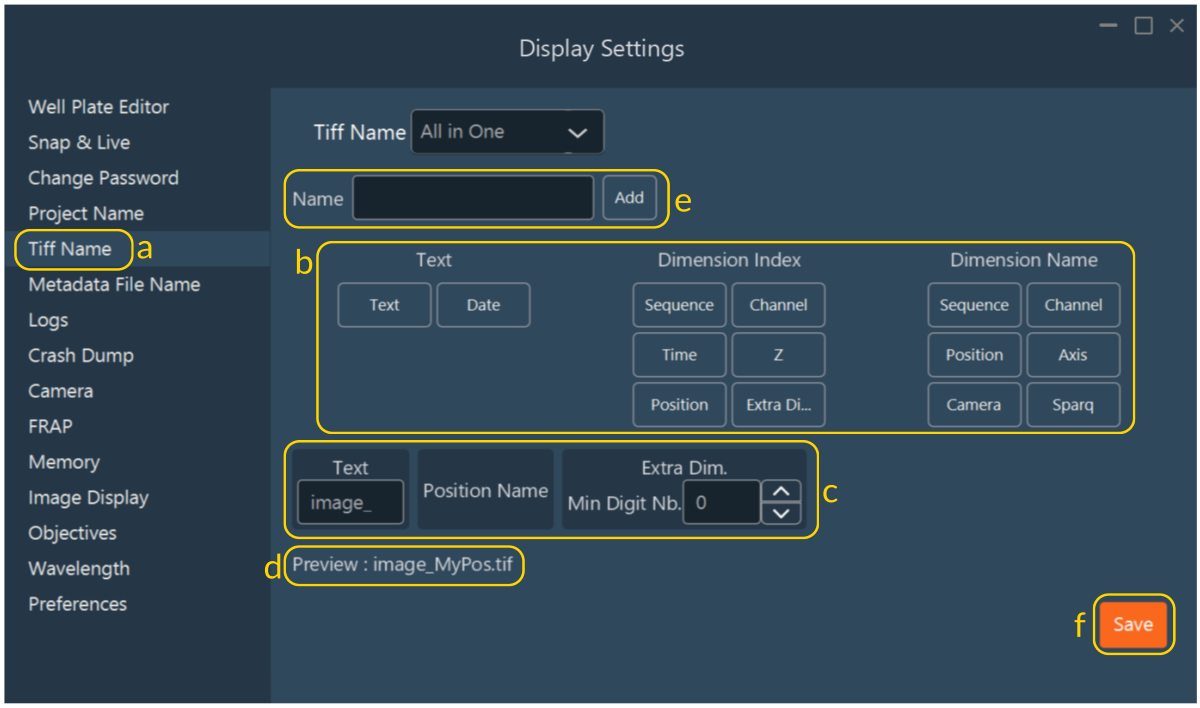
Establish a standardized naming pattern for new projects.



1. Select the **Project Name** tab.
2. Click the tags to add.
3. Modify the text and format of the selected tags in the line below.
4. Organize the tags through drag and drop, or remove them with a right-click.
5. Verify in the **Preview** that the result matches the desired format.
6. Click **Save** to apply the settings.

1.2.5 Tiff Export Naming Conventions

Customize how TIFF image files are named and exported.



By default, the software bundles all channels for a given position into a single multidimensional OME-TIFF file. Override this behavior to save each channel as a separate file.

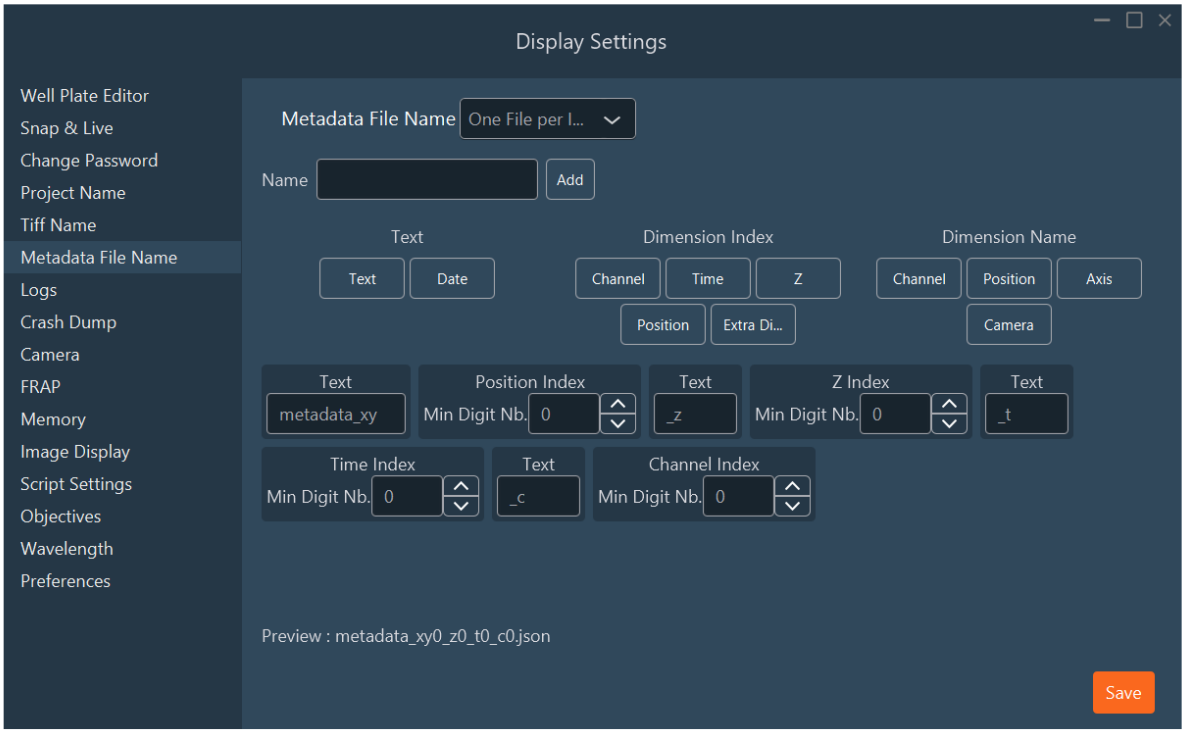
- **a.** Select the **Tiff Name** tab.
- **b.** Drag and drop metadata tags into the central construction area to build your naming convention (e.g., **text block + dimension name + dimension index**).

> **Index Zero-Padding**
>
> The **Min Digit Nb.** field determines the zero-padding length for the dimension index value (e.g., a value of 3 yields `001`, `002`).

- **c. Name Construction Tool**: Displays a visual schematic of the filename structure. Move tags with drag and drop, or remove them with a right-click.
- **d. Name Preview**: Displays an example of the generated filename.
- **e.** Enter a preset name and click **Add** to save the convention.
- **f.** Click **Save** to apply the settings.

1.2.6 Metadata Export

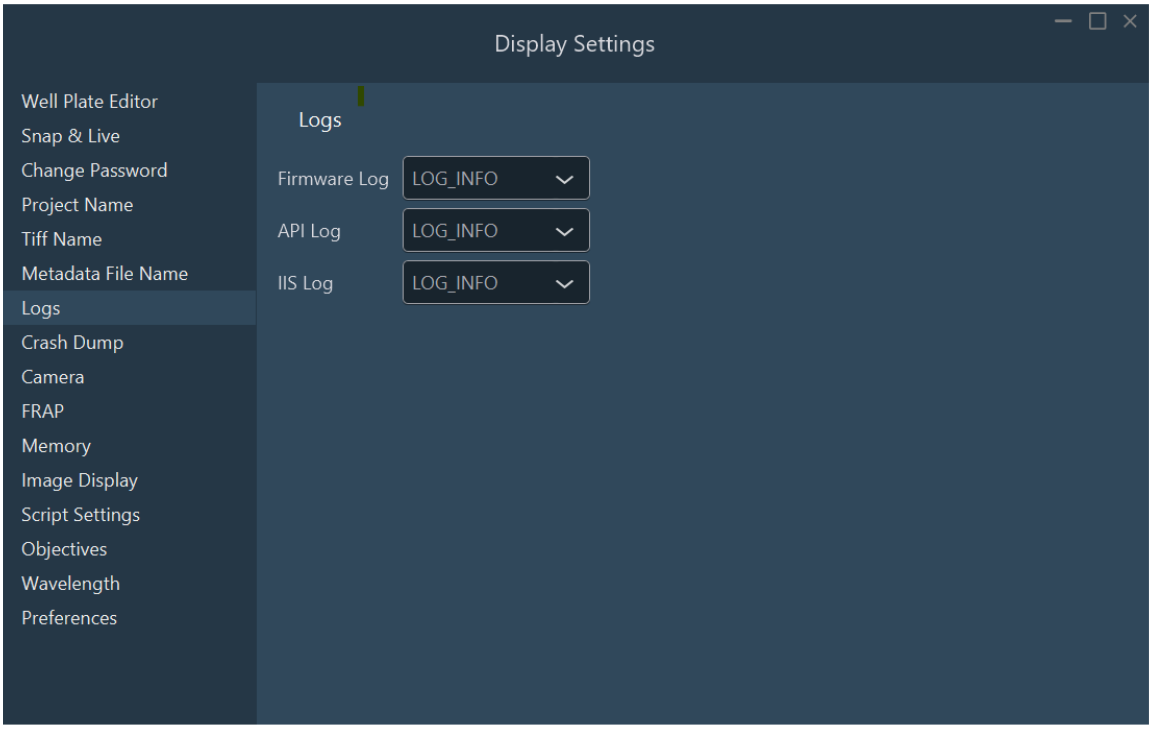
Customize how metadata files are named and exported.



1. Select the **Metadata File Name** tab.
2. Click the tags to add.
3. Modify the text and format of the selected tags in the line below.
4. Organize the tags through drag and drop, or remove them with a right-click.
5. Verify in the **Preview** that the result matches the desired format.
6. Enter a preset name and click **Add** to save the convention.
7. Click **Save** to apply the settings.

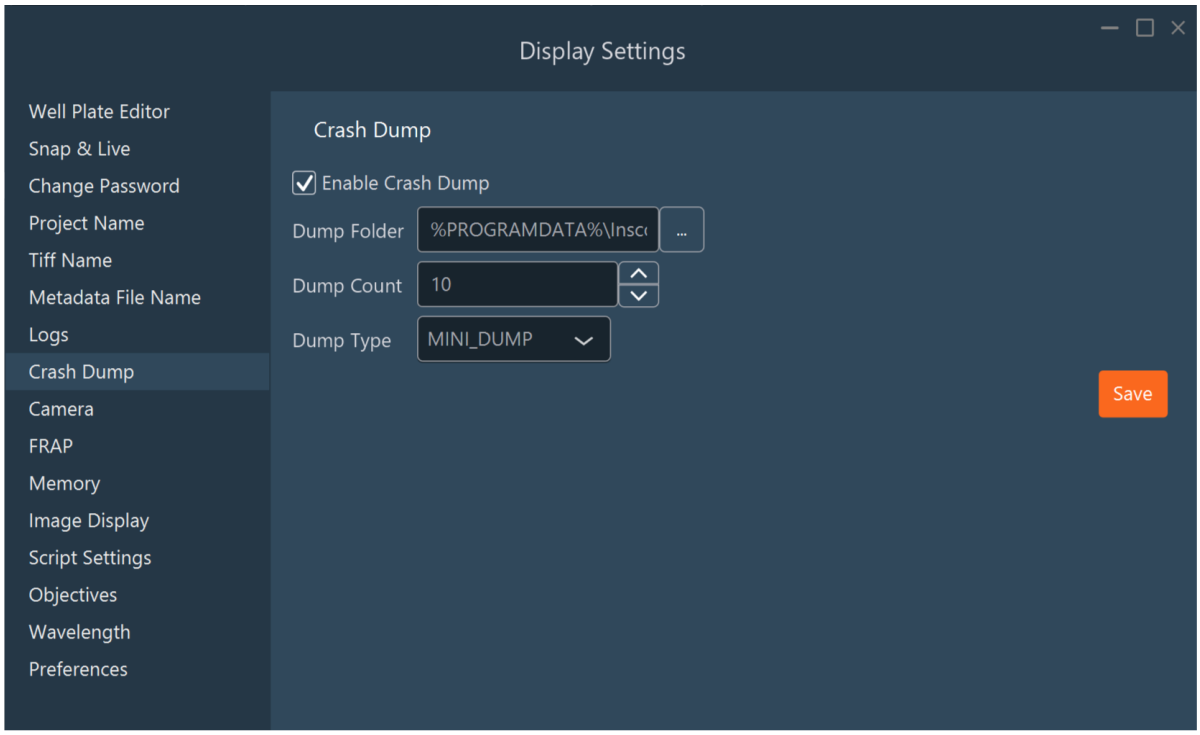
1.2.7 System Logs

Configure the log verbosity level (e.g., Error, Warning, Info, or Debug).



1.2.8 Diagnostic Crash Dump

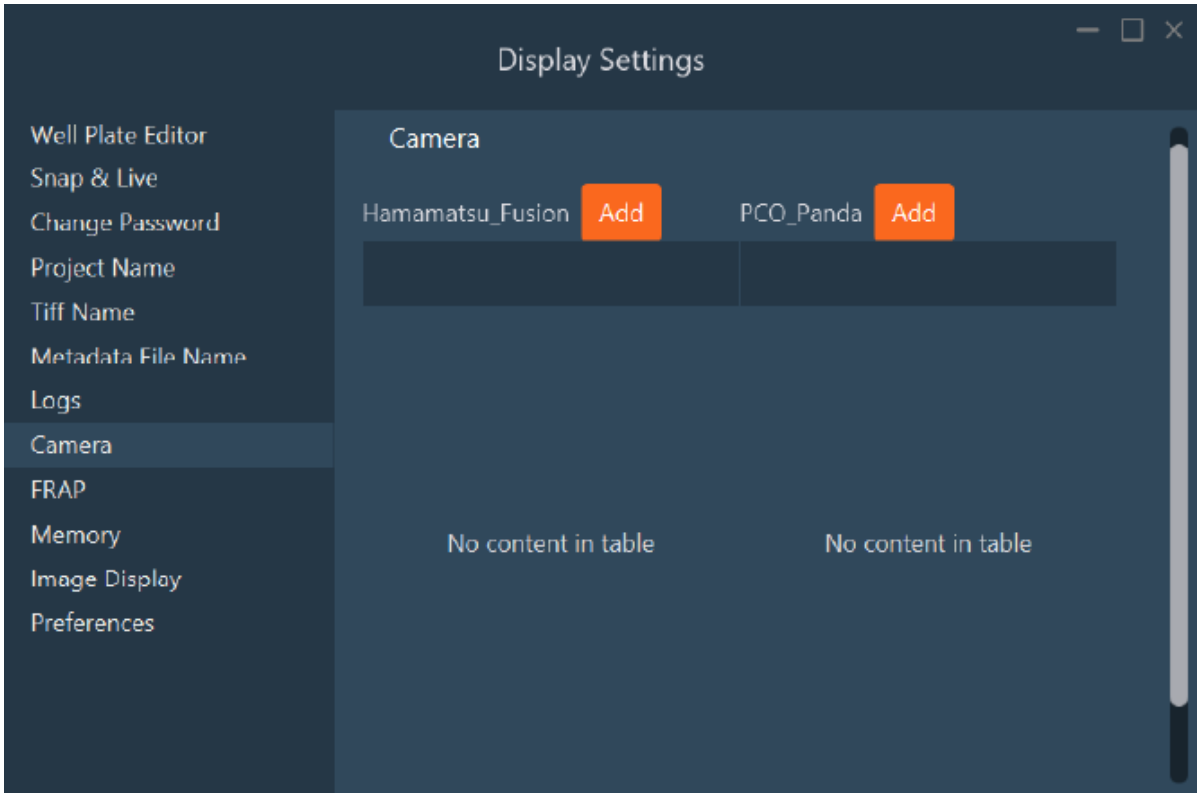
Configure crash reporting formats and retention policies.



- **Dump folder**: Define the directory path where the system saves crash reports.
- **Dump count**: Set the maximum number of reports retained (defaults to 10; the system deletes older files automatically).
- **Dump Type**:
  - `MINI_DUMP` (Standard Default): Saves essential crash stack information, sufficient for most debugging.
  - `FULL_DUMP`: Saves the complete memory state. Takes longer and consumes significant disk space.

1.2.9 Multi-Camera Management

Manage connected cameras. For multi-camera setups requiring precise field-of-view alignment, use the **Add** button to apply geometric image rotation vectors.

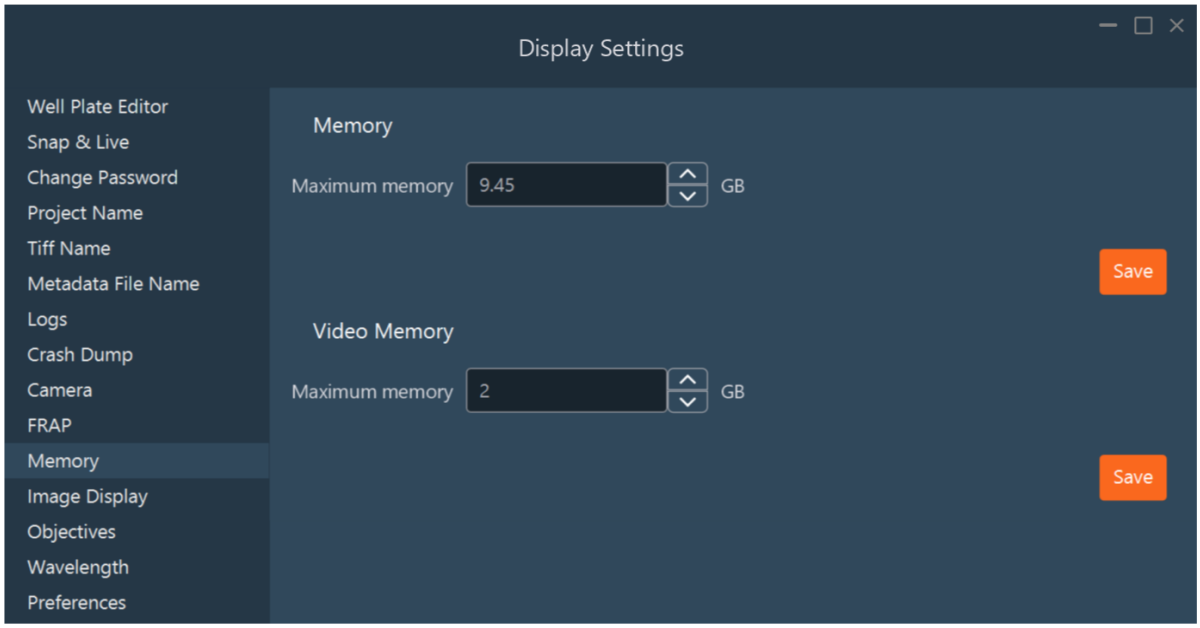


1.2.10 FRAP Sensitivity Parameterization

Review FRAP hardware limits and tune the virtual joystick sensitivity for FRAP optical calibration.

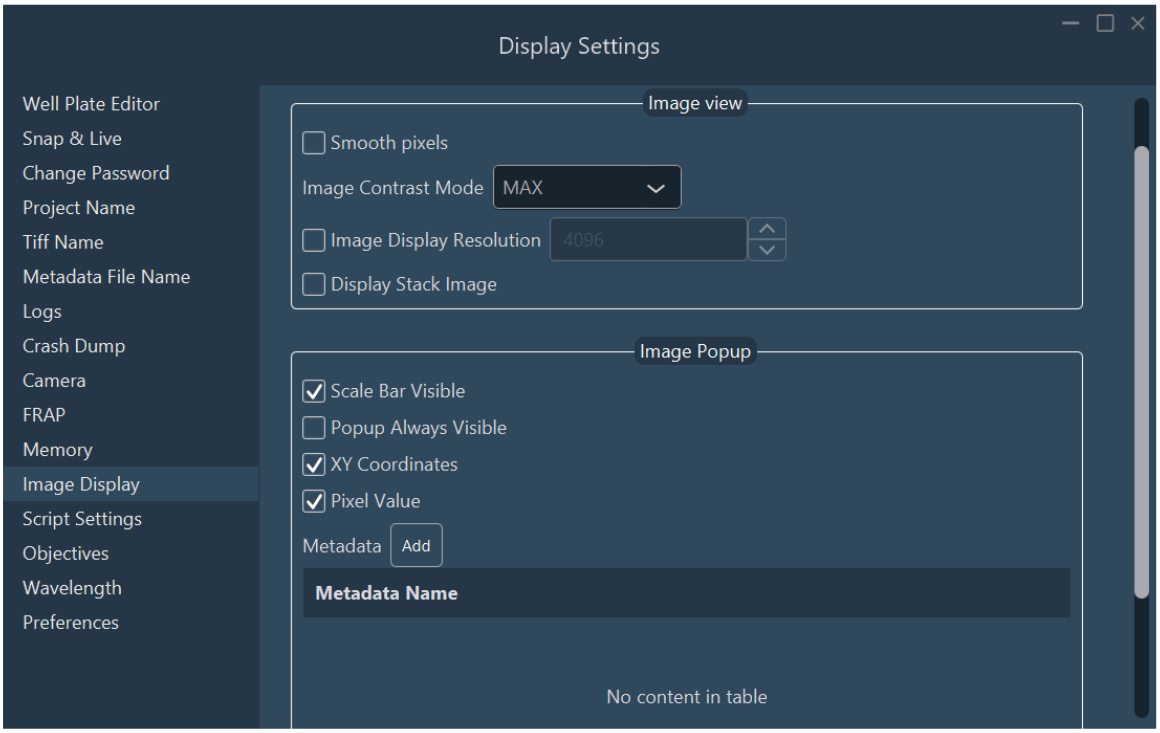
1.2.11 System Memory Allocation Engine

Manage memory allocation for Inscoper I.S. By default, the system reserves 2 GB for video buffer memory (VRAM) and allocates 60% of available system RAM to the application.



1.2.12 Primary Image Display Engine

Configure global visual rendering and display preferences.

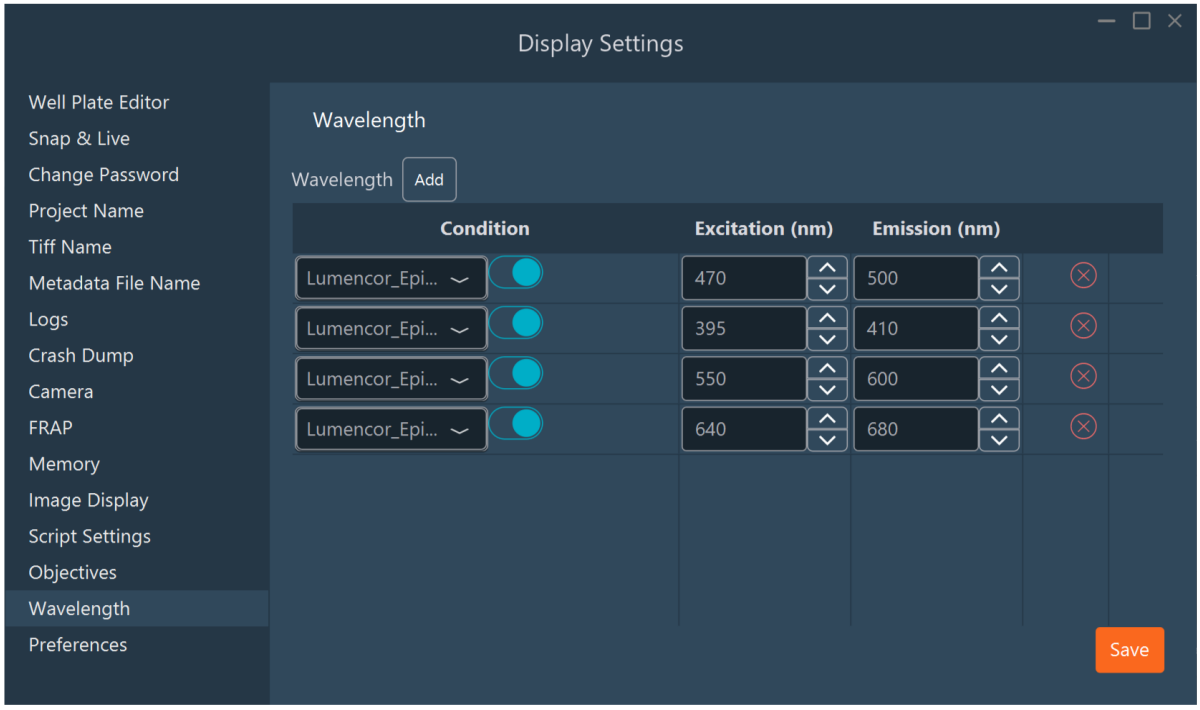


1.2.13 Objectives Database

Define the **Numerical Aperture (NA)** and the immersion **Refractive Index** parameters for your objectives. These values are critical for Nyquist sampling limit calculations and optical data processing.

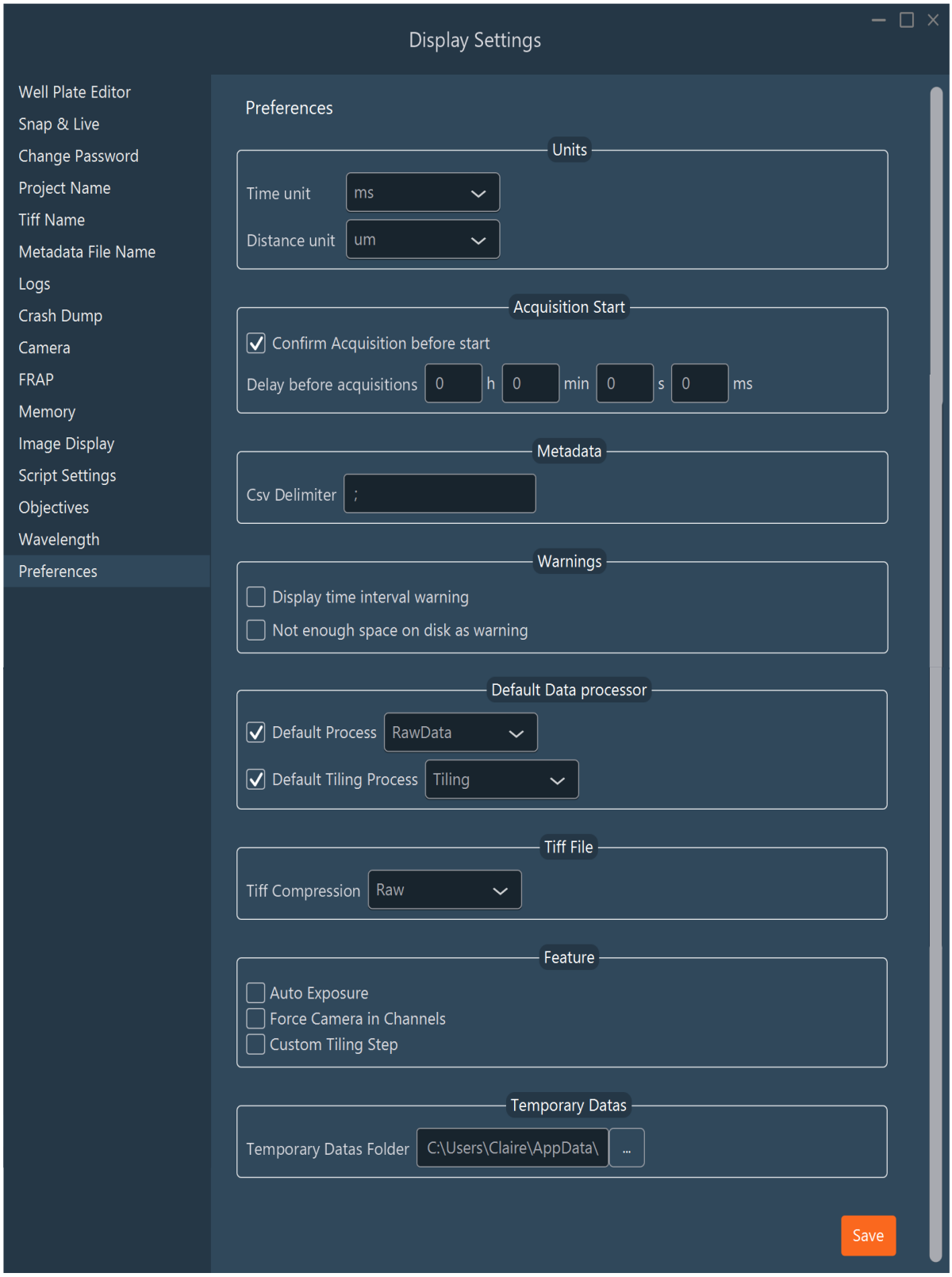
1.2.14 Source Wavelength Database

Define the **excitation and emission wavelengths** for your connected light sources, lasers, and filter cubes. The software uses this data for shading corrections, Nyquist sampling limits, and processing algorithms, and embeds it into the image OME-TIFF metadata.



1.2.15 Global System Preferences

Configure global preferences, including measurement units, system timing delays, and warning-prompt behavior.



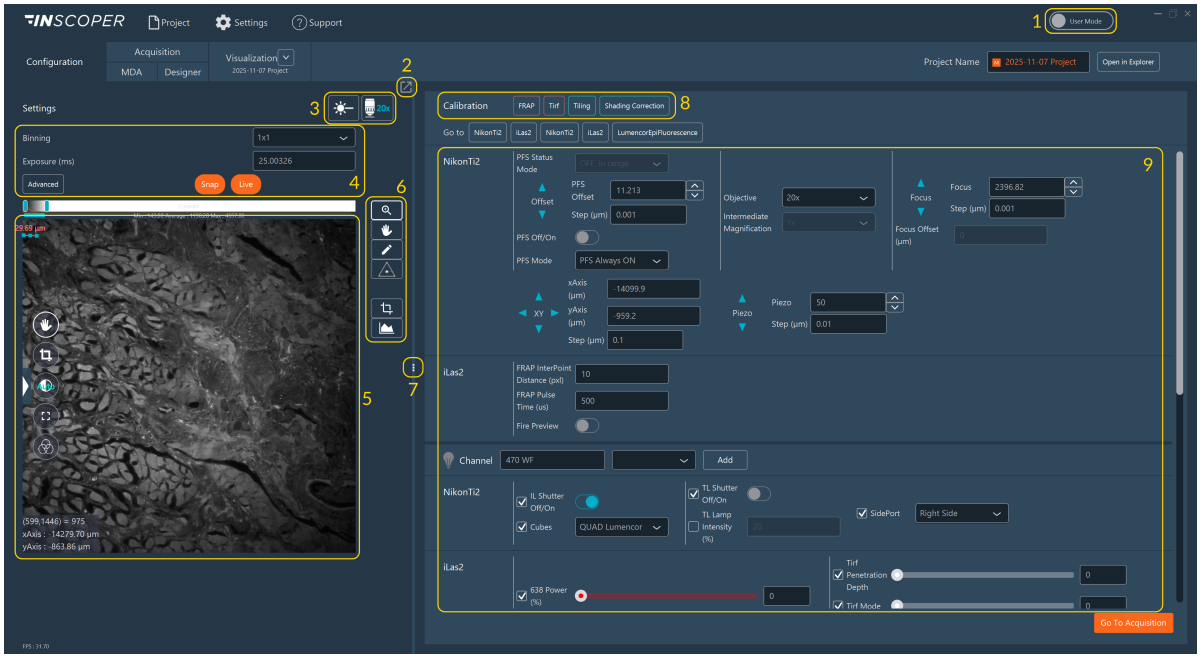
- **Auto exposure**: Enables the auto-exposure UI button on the primary dashboard.
- **Force camera in channels**: Associates specific cameras with specific channels, recommended for multi-camera or MAICO configurations.
- **Custom tiling step**: Define a translational overlap distance (in µm) rather than relying on a relative percentage for sequential tiling acquisitions.

1.3 Configuration

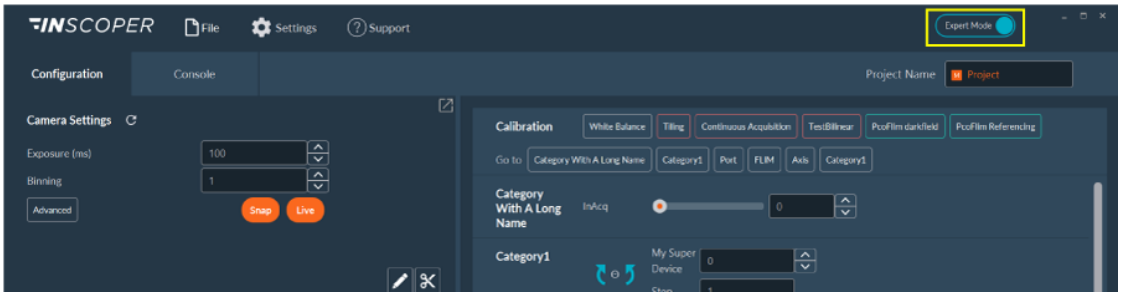
Use the **Configuration** tab to control each motorized device on the microscope, find the correct focus plane, select the optimal camera settings, and set up the channels.

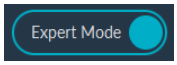
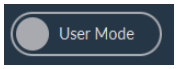
1.3.1 Overview

The configuration interface is divided into several sections:



1. **User Modes**: Two access levels are available in Inscoper I.S., based on the user's level of expertise in microscopy:



Mode	Description
	**Expert Mode** provides unrestricted access to all system settings and parameters. This mode is intended for microscope facility managers and researchers experienced in microscopy.
	**User Mode** provides restricted access to selected settings and parameters. The restrictions are fully customizable, ranging from basic channel configuration to the most advanced camera and device settings. This mode is intended for biologists who are not yet experienced with microscopy.

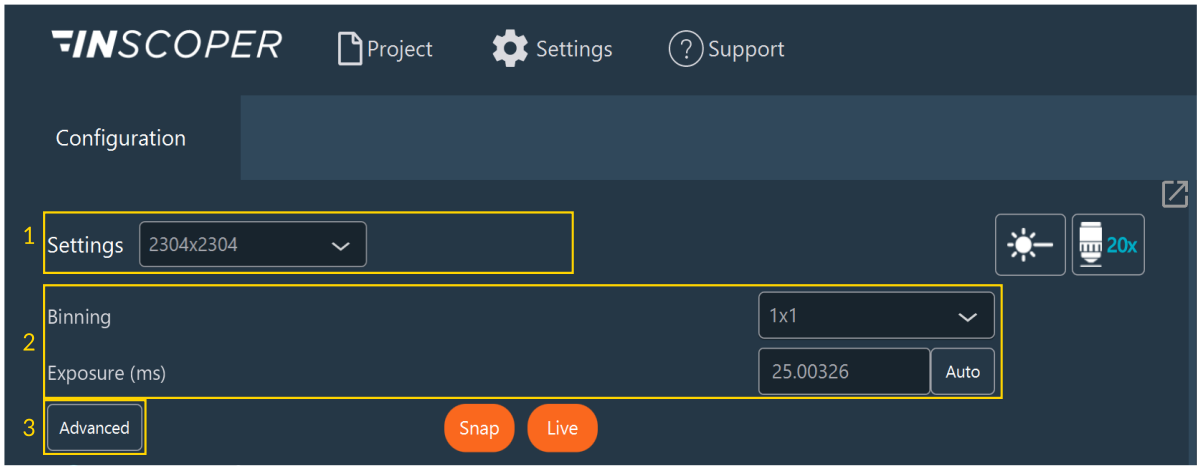
> **Note**
>
> Users can switch from User Mode to Expert Mode at any time. Facility managers can set a password to restrict access to Expert Mode.

2. **Split Screen**: For dual monitors, detach the live camera feed view to a separate screen to decouple it from the settings panel.
3. **Channel/Objective Access**: Switch between defined channels and objectives with a single click.
4. **Camera Settings**: Configure camera parameters.
5. **Live Viewing**: View the real-time image feed.
6. **Image Tools**: Interact with the image display.
7. **Divider**: Adjust the screen size of the central camera view and the side settings panel.
8. **Calibration Protocols**: Access optical and stage calibration tools (see Tiling Alignment and Shading Correction).
9. **Motorized Devices**: Access connected motorized hardware, create new channels, and save them to the project.

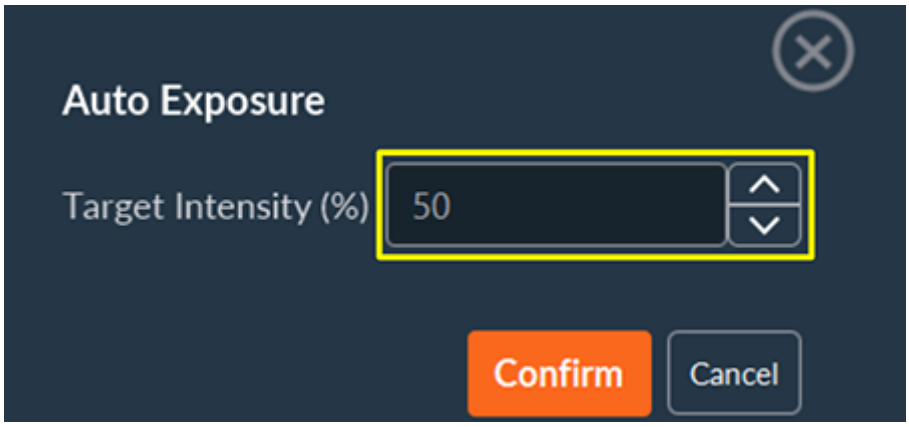
Rename the current project using the text field in the top-right corner.

1.3.2 Camera Settings

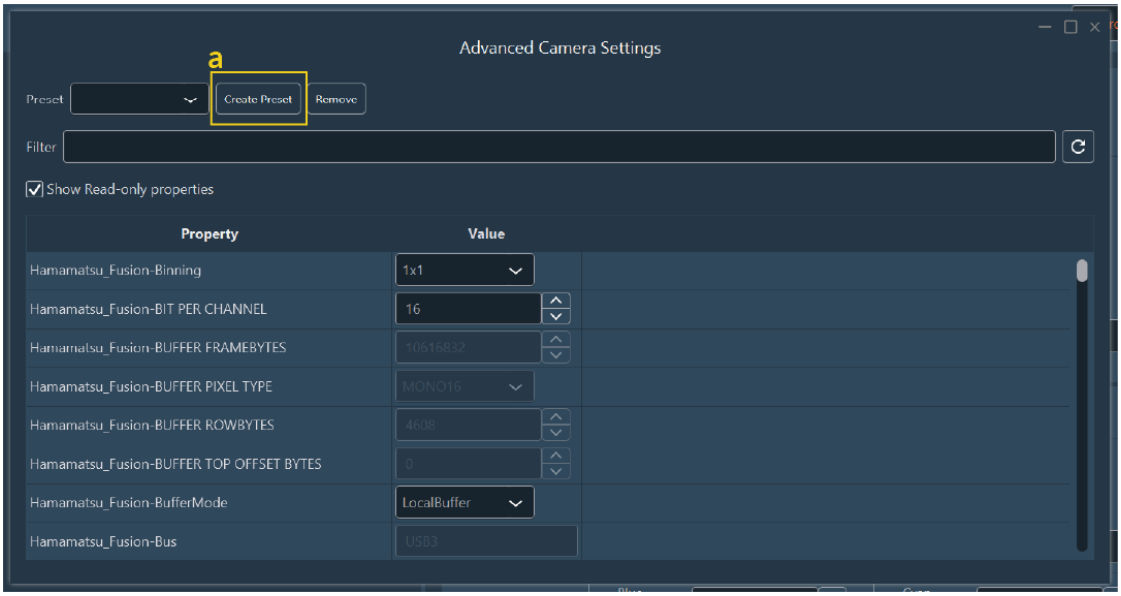
The software supports up to 4 cameras simultaneously. Create and save parameter presets for each camera using this section.



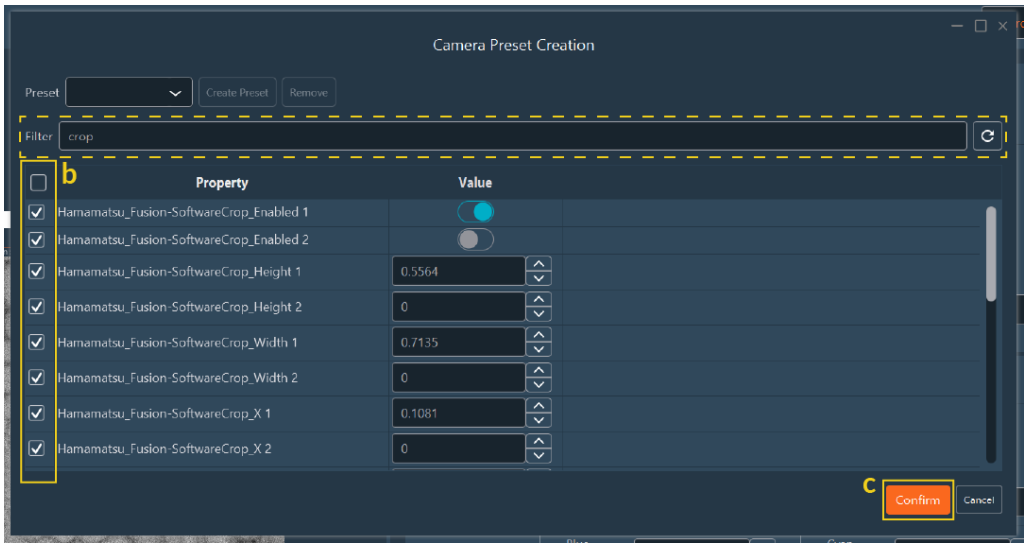
1. Select a camera preset from the drop-down list.
2. Adjust the exposure time and pixel binning. If **Auto Exposure** is enabled in the Display Settings, an **Auto** button is available. Auto Exposure adjusts the exposure time to prevent pixel saturation by analyzing the real-time pixel histogram.



3. Access **advanced camera settings** to create custom presets:

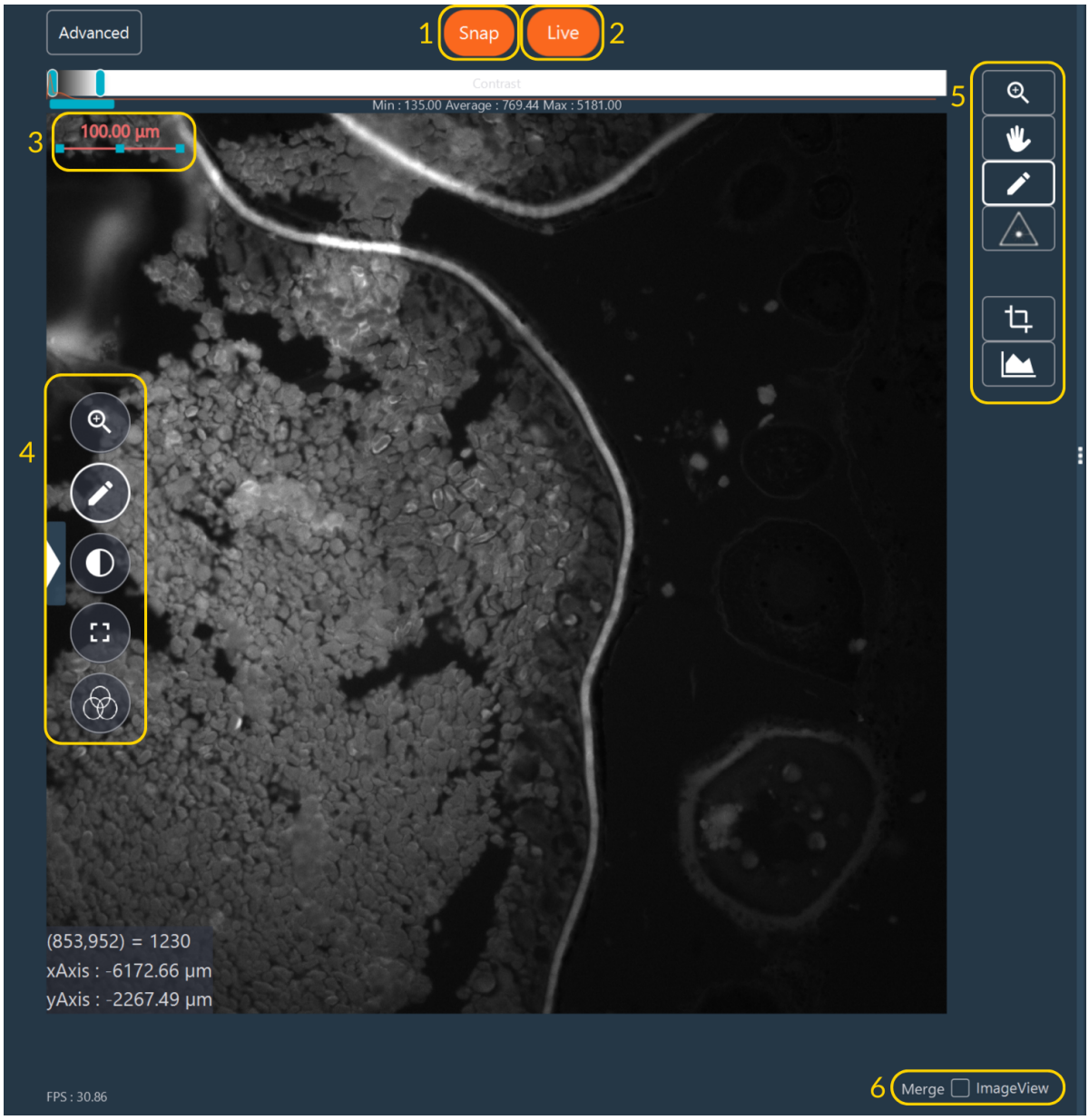


1. Click **Create Preset**:



2. Select the parameters to include in the new preset using the checkboxes. Use the search bar to filter the parameter list.
3. Click **Confirm**, then assign a name and save location for the preset.

1.3.3 Live Image Interaction



Use the toolbars to interact with the live image stream:

1. **Snap**: Capture a single image frame.
2. **Live**: Toggle the real-time camera stream.
3. **Scale Bar**: Double-click to modify orientation and physical length.
4. Tools to interact with the image:

Tool	Description
	Toggle automatic or manual contrast adjustments. Use the intensity sliders above the camera view to establish manual contrast boundaries.
	Toggle full-screen mode. Exit by pressing the button again or clicking the 'X'.
	Apply LookUp Tables (LUTs) to the image display in real time. Options include no LUT, single-color LUT, or a preset LUT.

LookUp Tables (LUTs)

Applying a LookUp Table maps raw grayscale intensity values to a specific colored gradient. This is critical for drawing ROIs accurately during quantitative assays, as pseudo-colored "heatmaps" (like the *fire* LUT) visually exaggerate subtle intensity differences that remain indistinguishable to the human eye in standard grayscale.

Available preset LUTs include:

- **Pixel indicator**: Highlights saturated (overloaded) pixels in red.
- **Inscoper ratiometric**: Designed for visualizing ratiometric images.
- **Multicolor LUTs**: Standard palettes such as "fire", "physics", etc.

5. **Advanced Image Tools** (a secondary toolbar with specialized functions):

Tool	Description
	Pan across the field of view by dragging and dropping. Scroll the mouse wheel to move along the Z-axis.
	Zoom digitally by scrolling the mouse wheel over the image.
	Draw Regions of Interest (ROIs) or crop the live image. Configure as a hardware sensor crop or software crop.

Create and manipulate ROIs to precisely control processing, cropping, and photo-manipulation.

Creation Modes

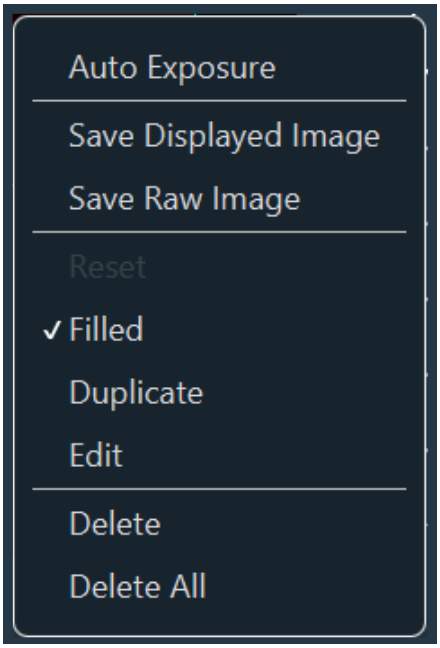
Tool	Mode	Description
	**Pen**	Add one or multiple discrete ROIs to the view.
	**Scissors**	Remove a drawn sub-shape from an existing ROI, retaining the external boundary.

Drawing Geometries

Select the desired geometry tool to draw the ROI boundaries:

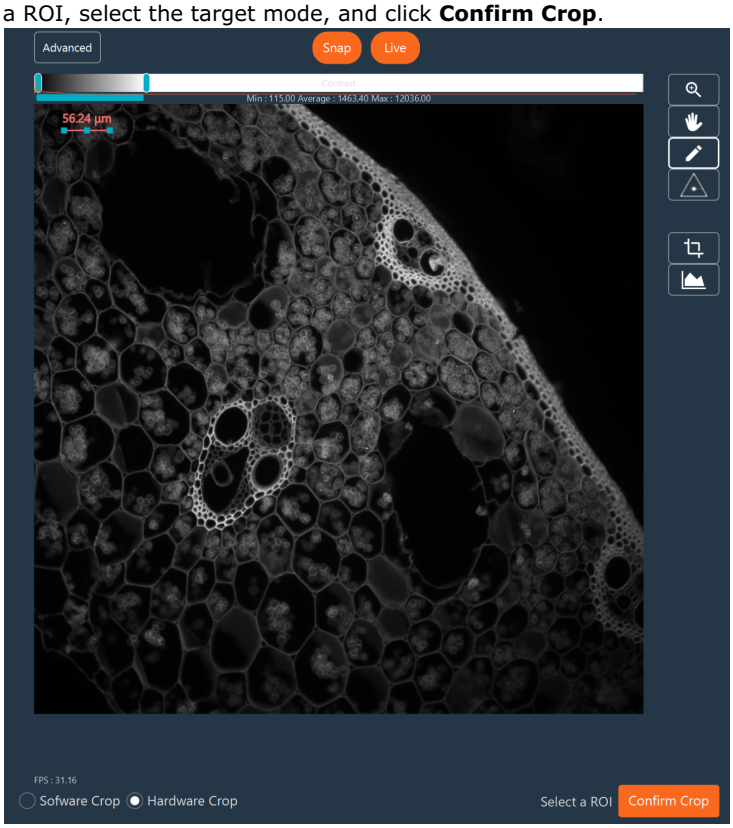
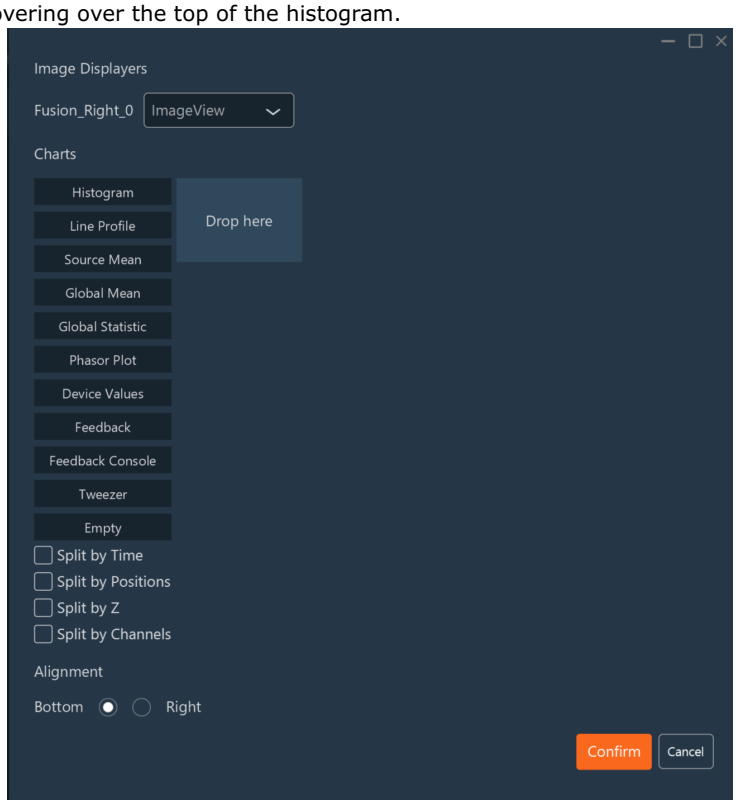
Tool	Shape	Description
	**Line**	Draw a 1D straight line.
	**Curve**	Draw a 1D freehand line.
	**Rectangle Edge**	Draw the 2D perimeter of a rectangle.
	**Filled Rectangle**	Draw a solid 2D rectangular area.
	**Circle Edge**	Draw the 2D perimeter of a circle.
	**Filled Circle**	Draw a solid 2D circular area.
	**Free Form Edge**	Draw a custom closed 2D perimeter.
	**Filled Free Form**	Draw a solid, custom closed 2D area.

ROI Management

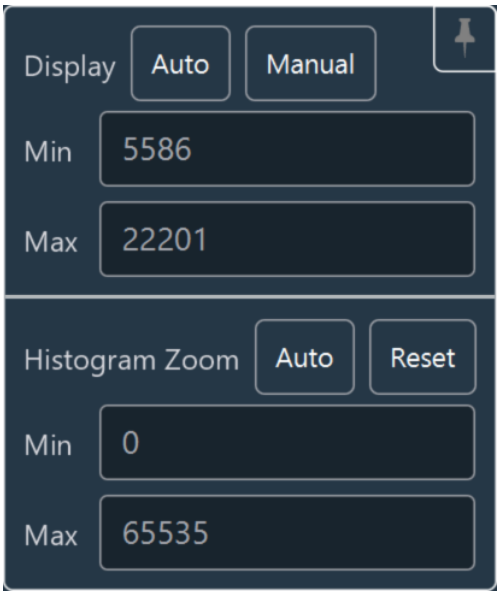


Right-click an existing ROI on the image canvas to access the spatial management window:

- Fill internal areas.
- Duplicate existing ROIs.
- **Edit ROI**: Manually define precise micron dimensions and specify absolute center coordinates within the camera field of view.
- Remove single or all drawn ROIs.

Tool	Plugin	Description
	**Fire on Click**	Trigger targeted photo-manipulation lasers. (Requires the FRAP module to be installed).
	**Crop Access**	Configure either **hardware** (sensor-level) or **software** (digital) cropping. Draw a ROI, select the target mode, and click **Confirm Crop**. 
	**Histogram Access**	Dock the quantitative histogram (or other 1D visualizations) below or to the right of the image panel. Adjust the X-axis scaling relative to camera bit-depth by hovering over the top of the histogram. 

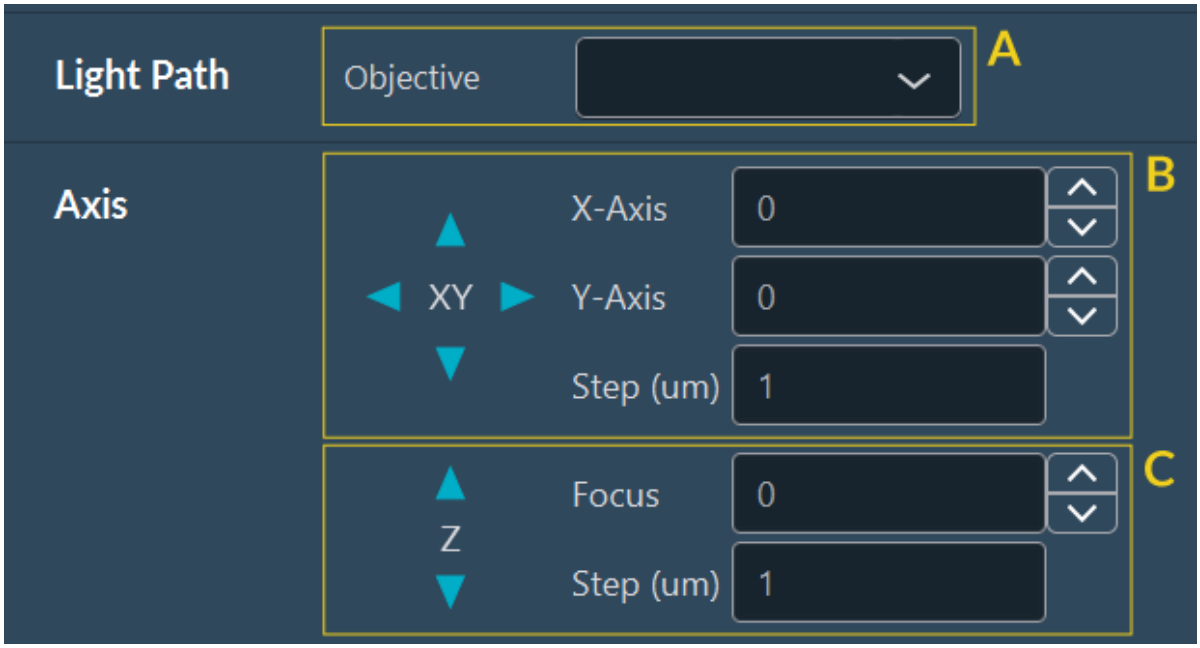
- **Contrast Adjustment**: Hover over the contrast bar located at the top of the image view. Use the high/low threshold sliders to modify the minimum and maximum displayed values, effectively redefining the black and white points to optimize the visual dynamic range. Zoom directly within this micro-histogram tool.



6. **Merged Images**: If the system is equipped with multiple cameras, use the **Merged images** toggle to display a combined live multiplexed superposition of all active camera streams.

1.3.4 Motorized Device Control

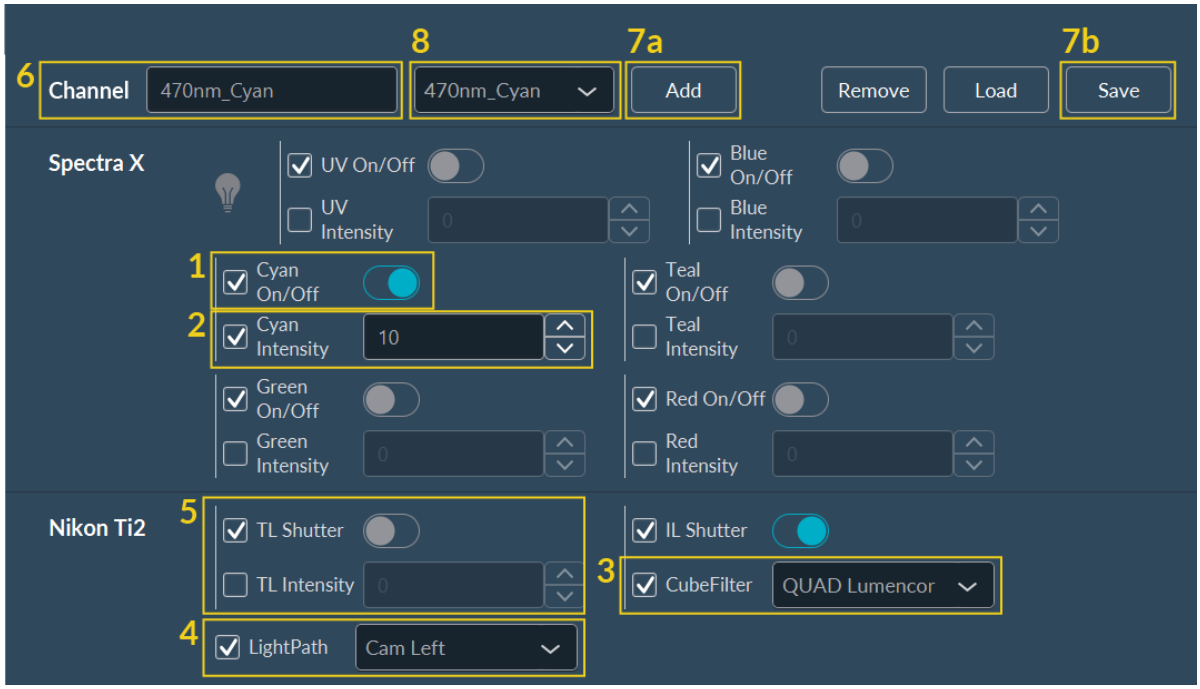
Interacting with Devices



This panel provides control over hardware connected to the Inscoper I.S., including the microscope stand, motorized XY stages, optical shutters, illumination sources, filter wheels, piezo Z-drives, and microfluidics pumps.

- Use panel **(A)** to move the objective turret.
- Panels **(B)** and **(C)** provide virtual joysticks to move the XY stage and Z-focus. Each arrow moves the stage or focus in the selected direction by the defined step.

Creating and Loading Channels



Expert users have direct control over optical path components (filter wheels, dichroic cubes, LEDs, laser lines). To configure a new fluorescent imaging channel:

1. Select the excitation light source.
2. Set its output intensity.
3. Select the optical filter cube.
4. Select the LightPath geometry.
5. *(Optional)* Select brightfield LED illumination to use instead of, or simultaneously with, the fluorescence path.
6. Assign a name to the channel configuration.
7. Save the channel:
    - **a. Temporarily** (for the current session) by clicking **Add** (**User Mode**).
    - **b. Permanently** (to the database) by clicking **Save** (**Expert Mode**).
8. The newly created channel populates the channel list.

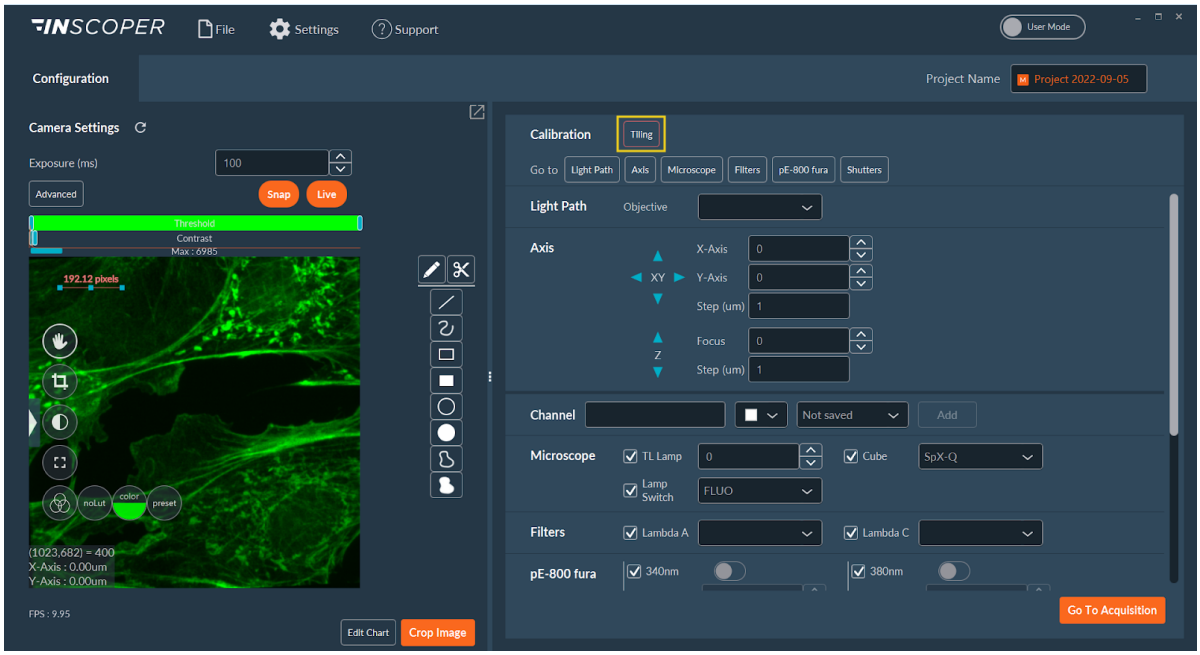
> **Camera Port Alignment**
>
> Many microscopes use more than one camera port (e.g., left and right ports). It is critical to select the correct active port before initiating imaging. Failing to select the active port may result in blank images or hardware errors during acquisition.

1.3.5 System Calibration

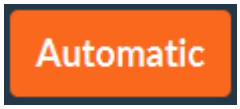
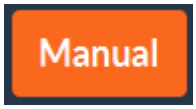
Access critical optical and stage calibration tools required for various sequences.

Tiling Alignment Calibration

Tiling calibration aligns the camera sensor's X/Y orientation with the motorized XY stage axes. This hardware alignment is strictly required for accurate image stitching.

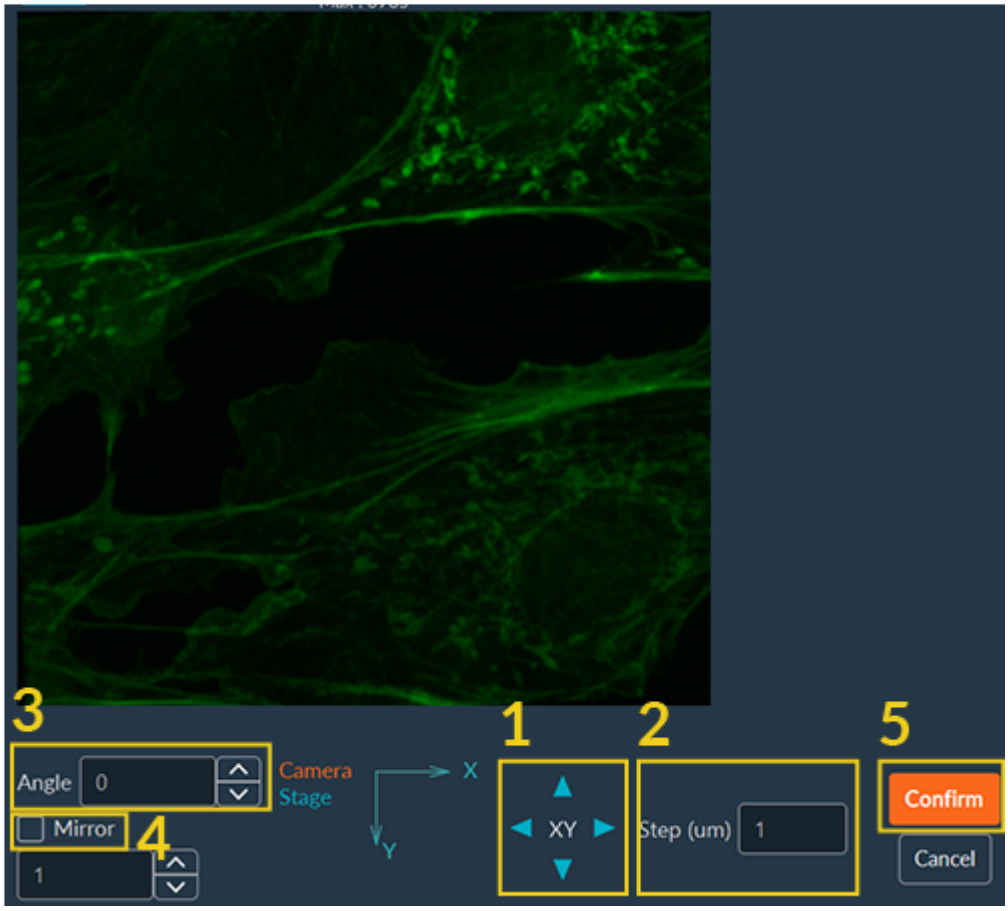


1. Click **Tiling** in the Calibration section. The indicator is orange if uncalibrated, and turns green when calibrated.
2. Select a calibration mode:

Mode	Description
	An automated routine. Click **Automatic** and confirm.
	A manual routine where you verify and correct the orientation.

> **Streaming Requirement**
>
> Stream camera data in **Live** mode and bring a sample into focus before initiating either calibration.

Manual Calibration Protocol:

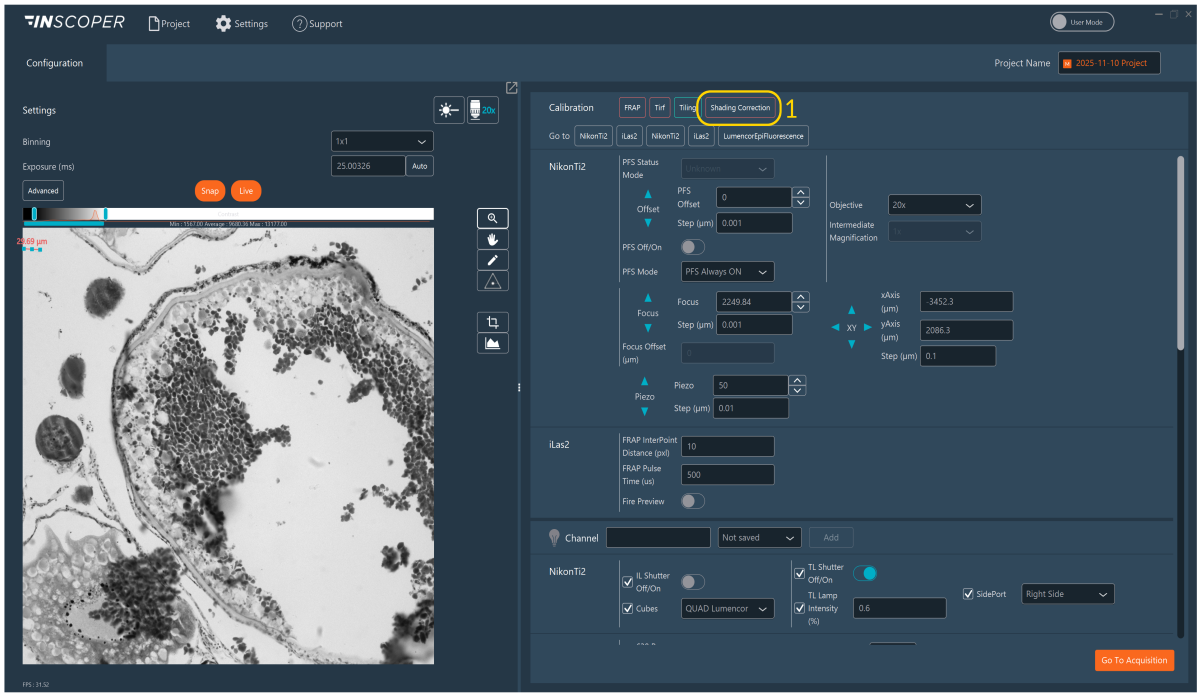


1. Move the XY stage using the virtual joystick. Confirm that the stage movement matches the image's orientation on the screen.
2. Adjust the movement step size for finer control.
3. Apply rotation to the live camera feed until it aligns with the stage movement axes.
4. Apply mirroring to the live camera feed until it aligns with the stage movement axes.
5. Click **Confirm** to save the calibration.

Shading Correction Calibration

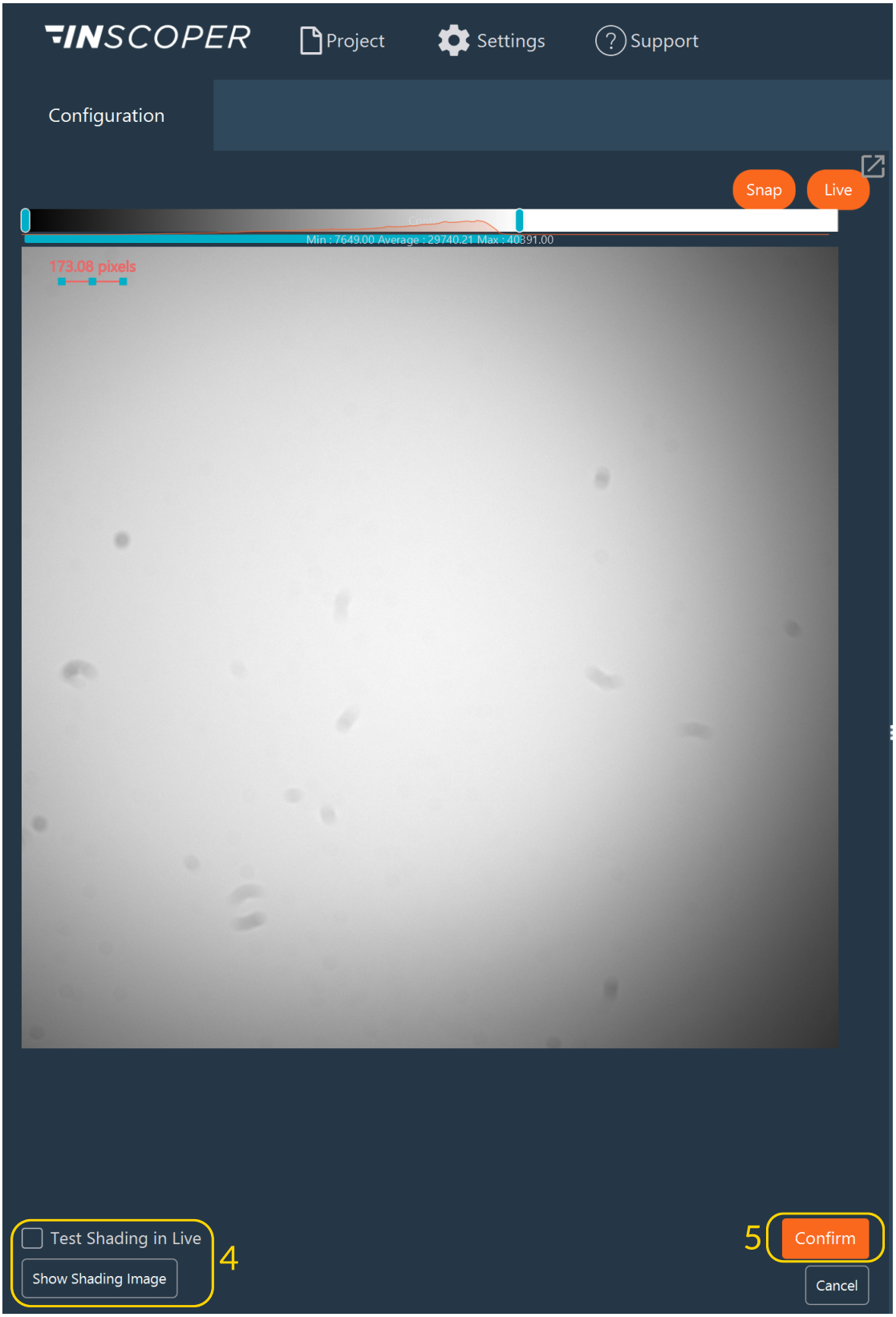
Shading Correction (flat-field correction) ensures uniform illumination intensity across the camera field of view, compensating for light source vignetting or unevenness. Perform this calibration using a uniform reference sample, such as an auto-fluorescent slide (e.g., Chroma).

EXECUTING THE SHADING CALIBRATION SEQUENCE



1. Click **Shading Correction**.

2. Specify the number of acquisitions. The software averages the captured frames to reduce noise.
3. Click **Confirm** to start the calibration routine.

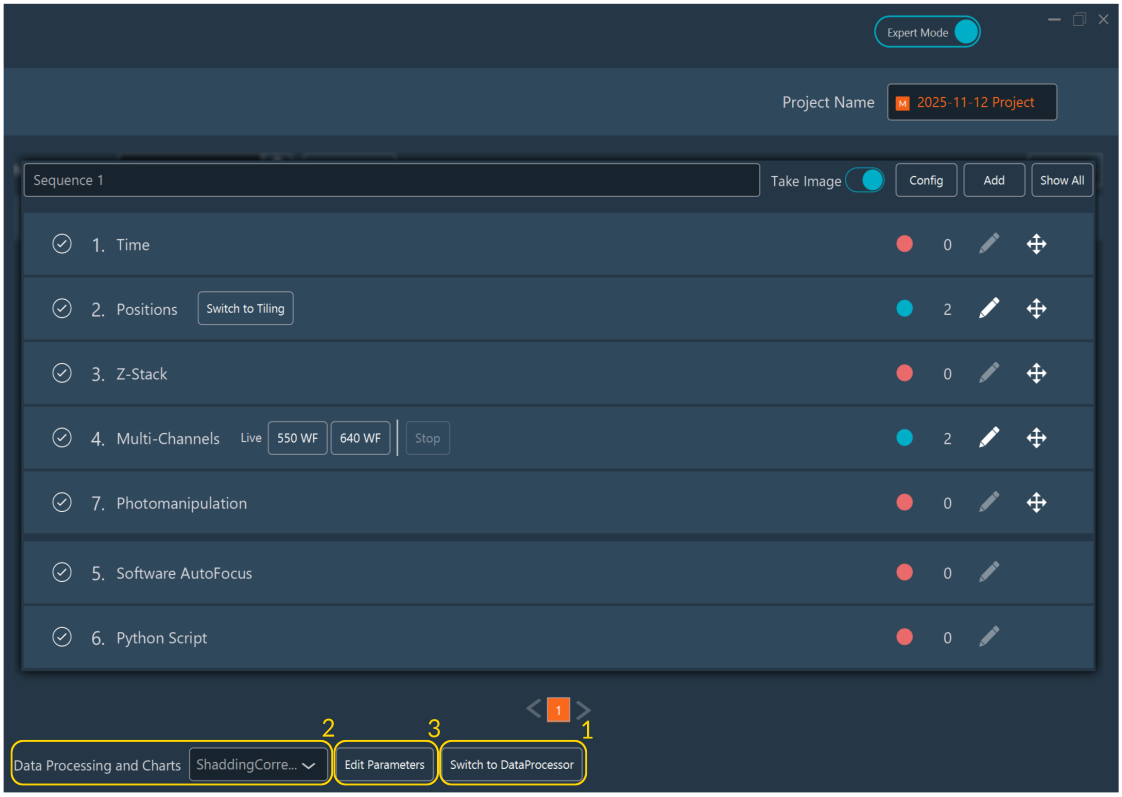


4. Once the calibration is complete, validate it:
    - Verify the result by clicking **Test Shading** in **Live** mode.
    - View the generated shading image map by clicking **Show Shading Image**.
5. Click **Confirm** to save the calibration.
6. Once calibrated, the **Shading Correction** indicator turns **green**.

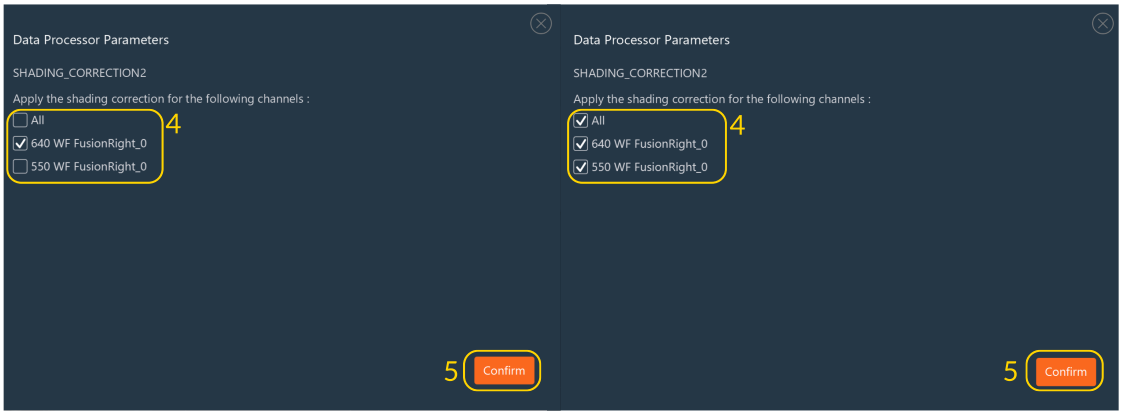
APPLYING SHADING CORRECTION VIA THE DATA PROCESSOR

Once calibrated, shading correction is not applied automatically; it must be explicitly invoked within the data processing pipeline.

1. In **Expert Mode**, construct a data processing pipeline containing the **Raw Data**, **Shading Correction**, and **Tiling** nodes.
2. Assign this pipeline to your sequence during the Acquisition setup.
3. Click **Edit Parameters** on the **Shading** node.



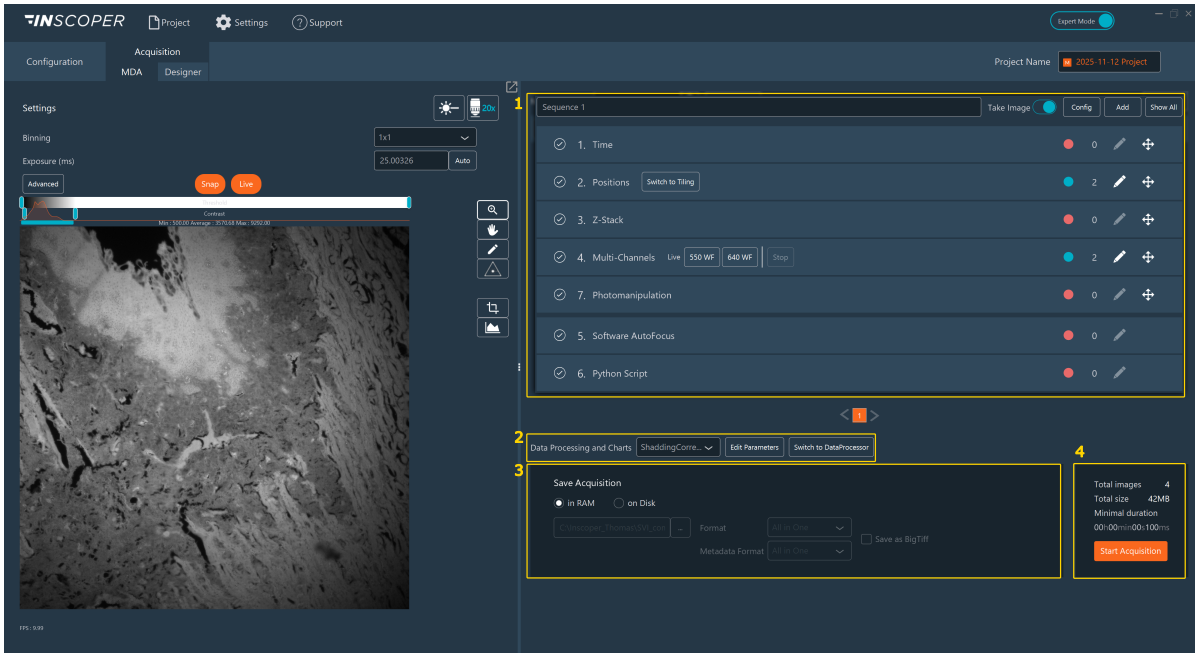
4. Select the channel(s) that require shading correction.
5. Click **Confirm** to validate your selection.



1.4 Acquisition

1.4.1 Acquisition

Use this tab to configure complete multi-dimensional acquisition sequences across all available dimension modules.



To set up a **multi-dimensional acquisition**:

1. Select the required dimension(s) from the panel.
2. Choose the appropriate data processor.
3. Specify the save directory for the output files.
4. Launch the acquisition.

> **Note**
>
> The interface varies depending on the microscope system hardware, particularly the available dimensions.

All dimensions share a consistent interface. The UI controls for configuring dimensions include:

Icon	Description
	**Inactive**: The sequence excludes the dimension. Click to activate.
	**Active**: The sequence includes the dimension. Click to deactivate.
	**Configured**: The dimension's parameters are set. Here, 11 images will be acquired for this dimension.
	**Edit**: Click to configure the dimension settings.
	**Reorder**: Drag and drop to change the dimension execution order. The dimension order defines the nesting loop of the sequence, executing from top to bottom.

Dimension Execution Order

The order of the active dimensions dictates the nested loop structure of the acquisition sequence. Use the **Reorder** handle to drag and drop dimensions to change their execution order.

Dimensions at the top of the list act as the outer loops of the sequence, progressing more slowly. Dimensions positioned lower in the list act as the inner loops, executing more frequently. For example, if **Time** is placed above **Channels**, the system acquires all channels for a single time point before progressing to the next time point. Conversely, if **Channels** is placed above **Time**, the system acquires the entire time-lapse sequence for the first channel before switching to the next channel. Permuting these dimensions correctly is critical, as it directly determines how the hardware coordinates the image acquisition.

1.4.2 Time

Use the **Time** dimension to configure time-lapse acquisitions to observe dynamic biological processes.

Configuring a time-lapse

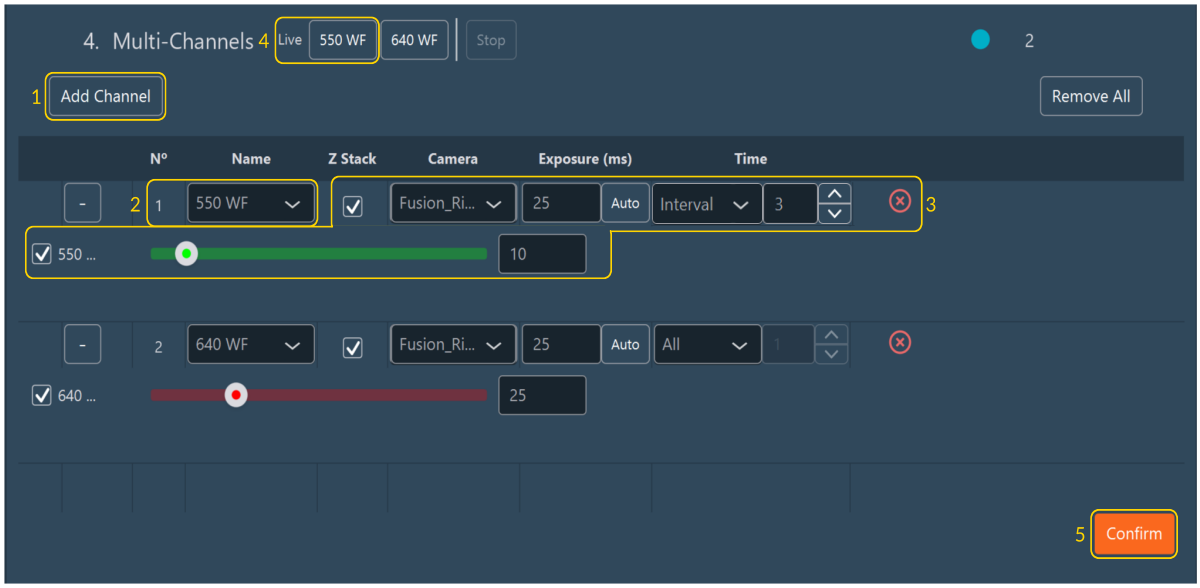
1. Specify the **total number of time points** (frames) required in the acquisition.
2. Define the **interval duration** between consecutive time points (minimum time to wait between frames).

> **Note**
>
> The software adapts the **Total Time** automatically. You may also set the total time manually and infer the interval duration for a fixed number of time points.

1. Enable **Burst Mode** to acquire images continuously at the maximum hardware frame rate, ignoring the specified interval. Use this to capture rapid kinetics.
2. Click **Confirm** to save the time dimension settings.

1.4.3 Multi-Channels

Use the **Multi-Channels** dimension to configure one or more channels (e.g., fluorescence, brightfield) for the acquisition sequence. Channels must have been defined previously in the Configuration Tab.



CONFIGURING CHANNELS

1. Click **Add Channel**.
2. Select a channel from the list.
3. Adjust the following parameters for each channel independently:
    - **Z-stack**: Enables the Z-stack dimension specifically for this channel.
    - **Z-Offset**: Applies a spatial Z-offset. This is used when focal planes vary between emission wavelengths (chromatic aberration) or when targets are at different depths.
    - **Shutter Blink**: Optimizes hardware shutter actuation to minimize sample exposure time and prevent photobleaching.
    - **Camera**: Assigns the active camera (for multi-camera systems).
    - **Exposure (ms)**: Defines the sensor exposure time for the channel.
    - **Time**: Defines the acquisition frequency within a time-lapse (e.g., acquire this channel at all time points, only the first time point, only the last time point, or at intervals).
    - **Intensity**: Adjusts the excitation light source intensity (hardware-dependent, usually % power).

    > **Note**
    >
    > To delete a channel, click the red cross icon.

4. To start a real-time preview of a channel, click the button corresponding to the channel's name (located next to the **Live** label in the multi-channel header). To apply any subsequent configuration changes, click the button again.
5. Click **Confirm** to apply the configuration.

> **Note**
>
> If you change the exposure time or intensity during a live preview, click **Live channel** again to refresh the hardware and update the image stream.

Tip

Reorder the execution sequence of the channels by dragging and dropping them in the list.

1.4.4 Positions

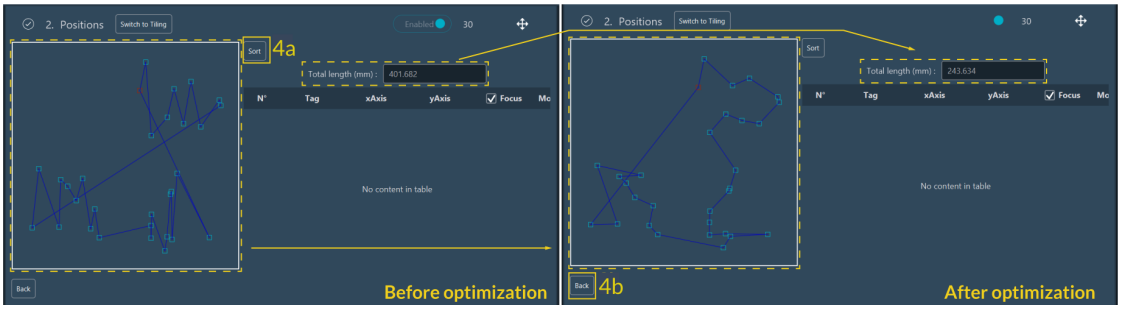
The **Positions** dimension captures multiple discrete XYZ locations across a sample.

Switch to Tiling

Toggle directly to the **Tiling** dimension using the **Switch to Tiling** button to capture a continuous mosaic rather than discrete points.

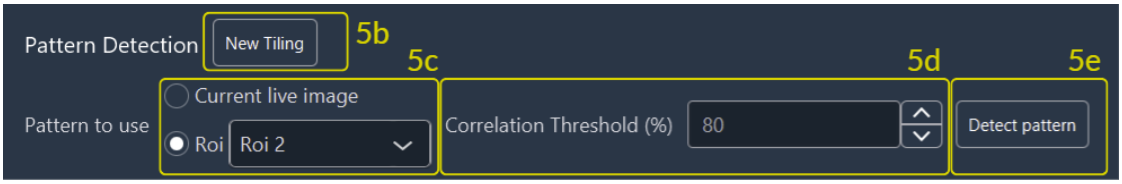
Adding and Optimizing Positions

1. Generate a fast Preview of the sample to act as a navigation map.
2. Move the **stage** to the target area and adjust the **focus**.
3. Click **Add position** to record these XYZ coordinates in the position list.
4. To optimize the **stage movement** trajectory across multiple positions, click the **Path** button.

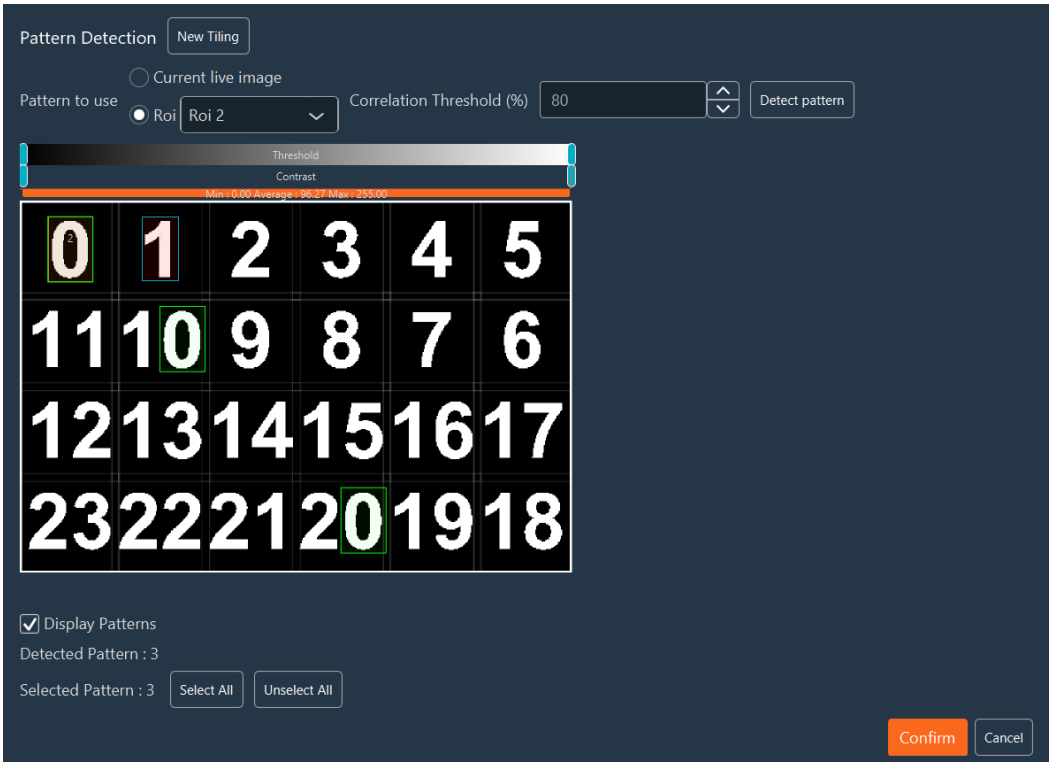


- **a**. By default, the stage visits positions in the order they were added. The interface visualizes this path and calculates the total travel distance. Click **Sort** to compute an optimized route (e.g., shortest path) that minimizes travel time.
- **b**. Once optimization is complete, click the **Back** button to return to the principal position view.

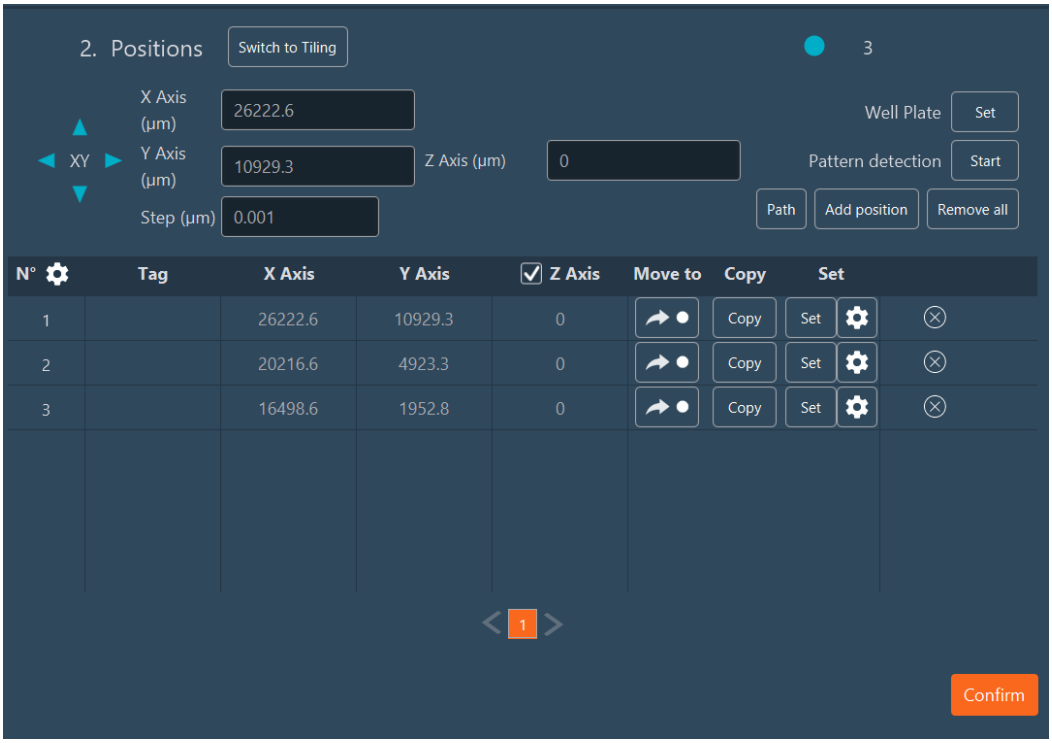
5. Use the **Pattern Detection** feature to automatically find XYZ coordinates matching a visual pattern in the sample.



- **a**. Click the **Start** button located in the Pattern Detection section.
- **b**. Create a reference scanning area by clicking **New tiling**. Configure it as described in the Tiling dimension section.
- **c**. Define the source pattern to detect:
  - **Current Live Image**: Extracts the pattern from the current live FOV.
  - **ROI**: Draw an ROI around the specific biological pattern to detect.
- **d**. Set the **Correlation Threshold** (this defines the minimum similarity percentage to identify a match).
- **e**. Click **Detect pattern** to initiate the scanning algorithm.
- **f**. Recognized patterns are highlighted with bounding ROIs over the preview. Manage these selections using **Select All**, **Unselect All**, or by clicking individual ROIs to toggle them.



- **g**. Click the **Confirm** button. The geometric coordinates of each detected pattern are appended automatically to the position list.

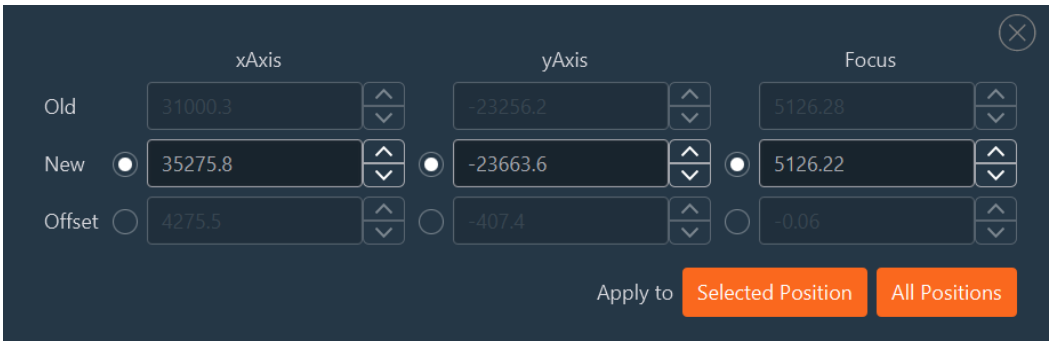


6. Configure the **Move Strategy**: For multiposition acquisitions spanning large travel distances, select how the stage translates:
    - **Fastest**: Navigates directly from point A to point B at maximum speed.
    - **XY Split**: Translates between positions incrementally. Specify the movement interval in the **Step (µm)** field.

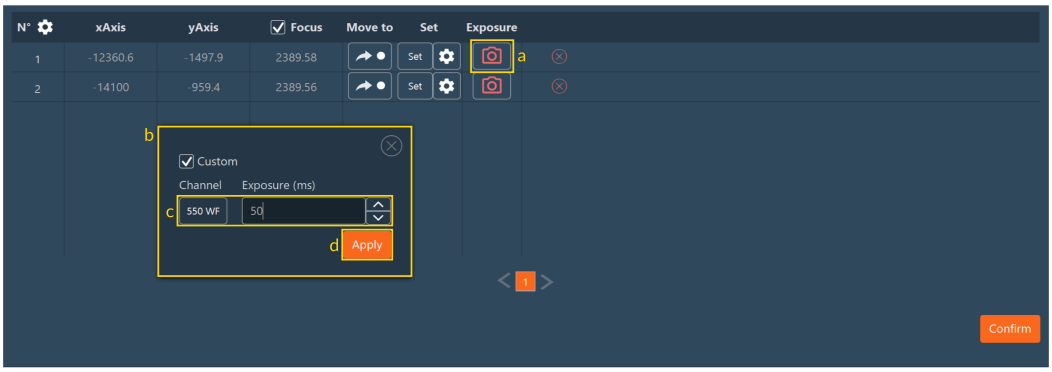
7. **Position Editing & Overrides**:

- **X-Axis/Y-Axis**: Manually modify coordinates.
- **Focus / Z-axis**: Check this box to force the system to use the Z-value saved for this position. If unchecked, the microscope maintains its current live Z-value upon arrival.
- **Move to**: Immediately drives the stage to this coordinate.
- **Set**: Overwrites the saved coordinates for this position with the current live XYZ stage position. Clicking the

modifies specific axes (X, Y, or Z) and applies global offsets.

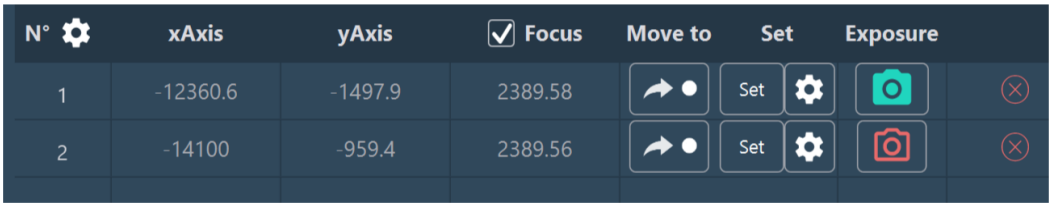


- **Custom Exposure per Position**: Assign unique camera exposure times to specific spatial positions (useful if sample brightness varies drastically across a tissue section).



1. Click the **camera icon** on the row corresponding to the position where you want to adjust the exposure time.
2. Check the **Custom** box to unlock exposure time modification.
3. Modify the exposure value in milliseconds. This override applies only to the currently active channel.
4. Click **Apply** to confirm the changes.

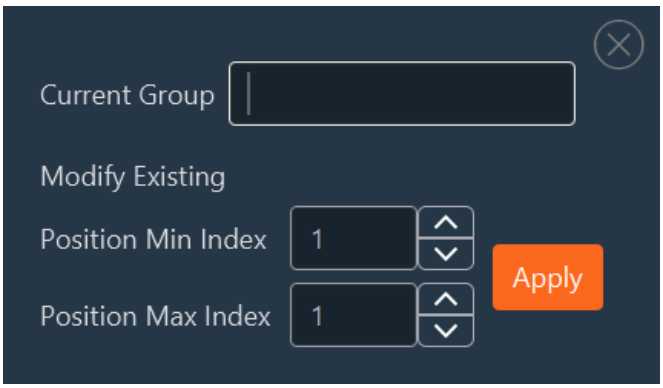
When a row is modified, the **camera icon color** changes from **red to green**, serving as a visual indicator that a custom exposure time is active.



- 

: Permanently deletes this position from the list.

8. **Grouping**: To manage large numbers of positions, cluster them into logical groups. Click the **gear icon** to access group settings:
    - Assign a semantic name to the group.
    - Indicate the contiguous range of position indices to group together (e.g., 1-15).
    - Click **Apply**.
    - Repeat these steps to create multiple, non-overlapping groups.

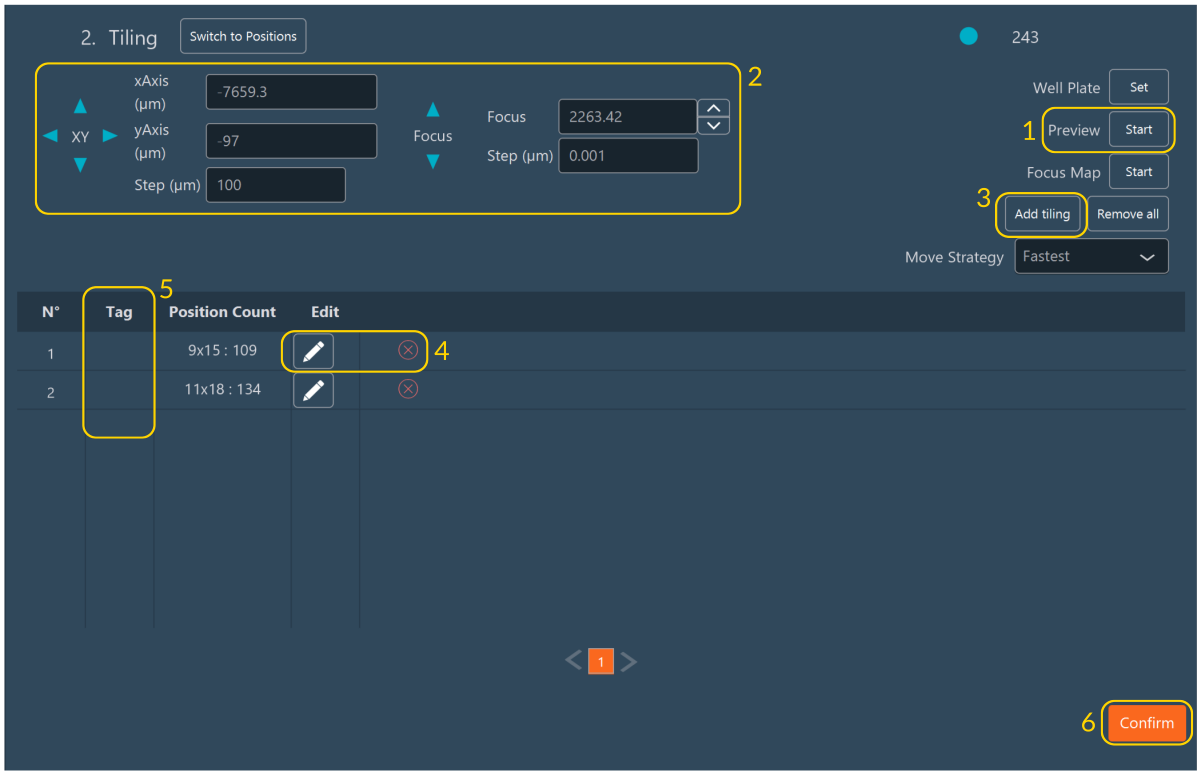


9. When all position geometries and parameters are finalized, click **Confirm**.

1.4.5 Tiling

Creating a Tiling

Use the **Tiling** dimension to capture large sample areas through automated mosaic acquisition.



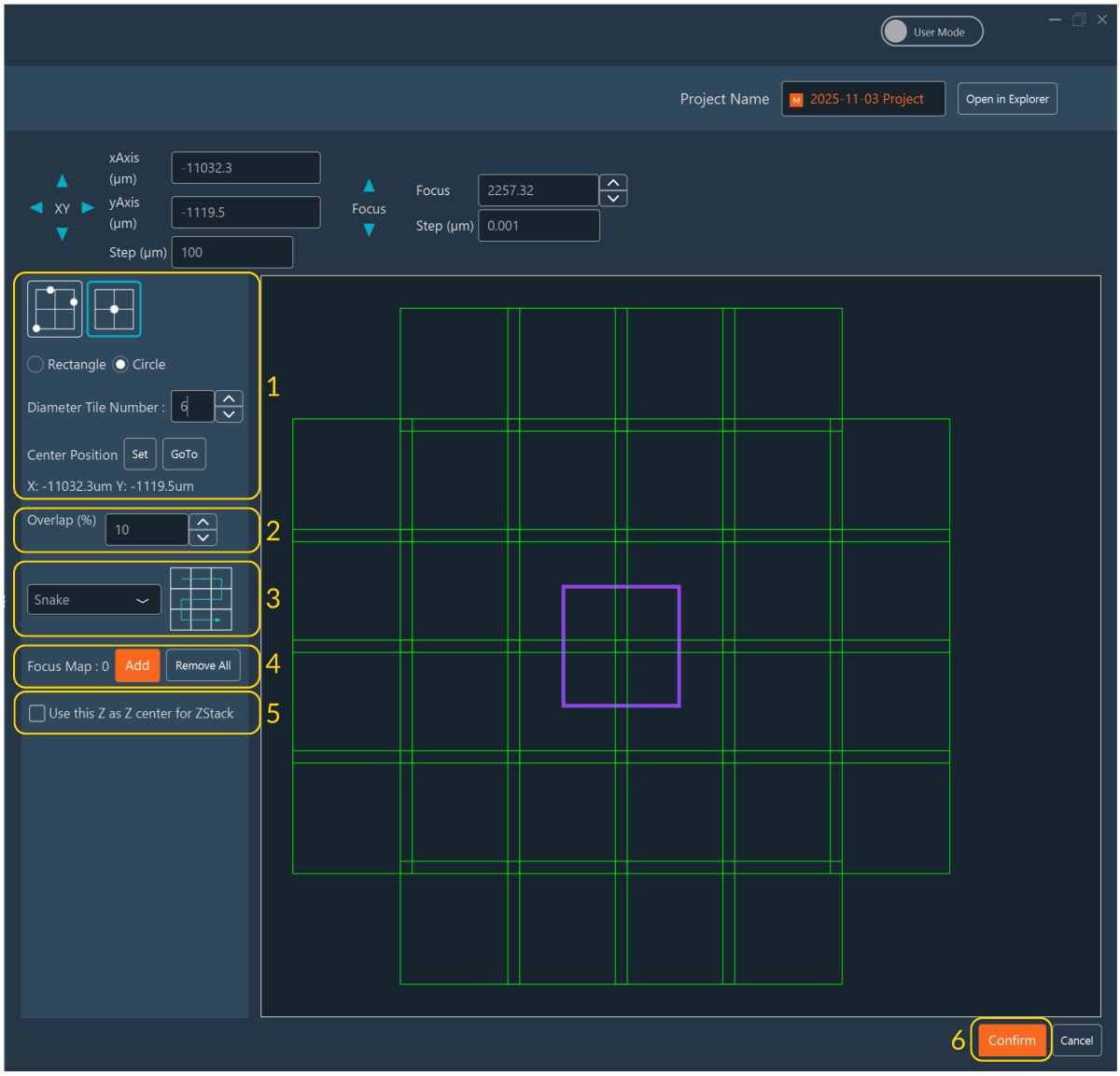
Switch to Positions

You can toggle directly to the Positions dimension using the **Switch to Positions** button if you require discrete capture points instead of a continuous mosaic.

1. Generate a low-resolution Preview of the sample.
2. Move the stage to the region of interest and focus the image.
3. Click **Add tiling** to append a new tiling configuration block to the dimension list.
4. To configure the parameters of a tiling block, click the **Pen** icon. To delete a block, click the **Red Cross**.
5. Assign a label to each tiling block by modifying its field in the **Tag** column.
6. Click **Confirm** to save the tiling configuration.

Setting Up a Tiling Layout

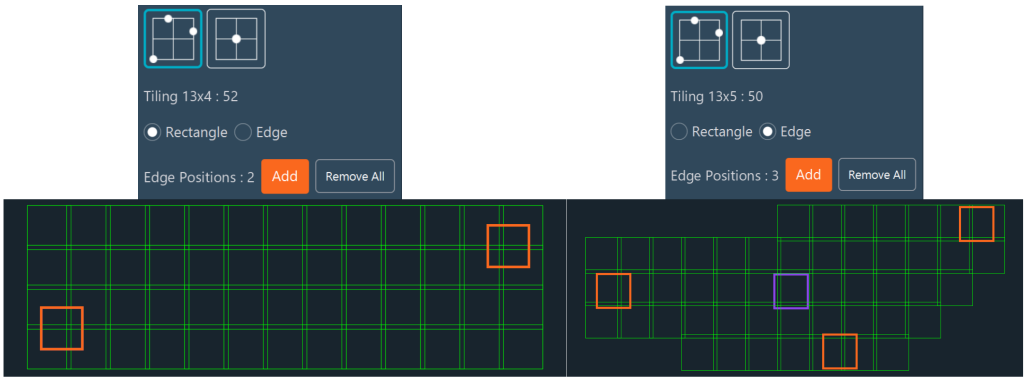
Define the physical tiling boundary using two topological methods:



1. **Define Boundaries:**

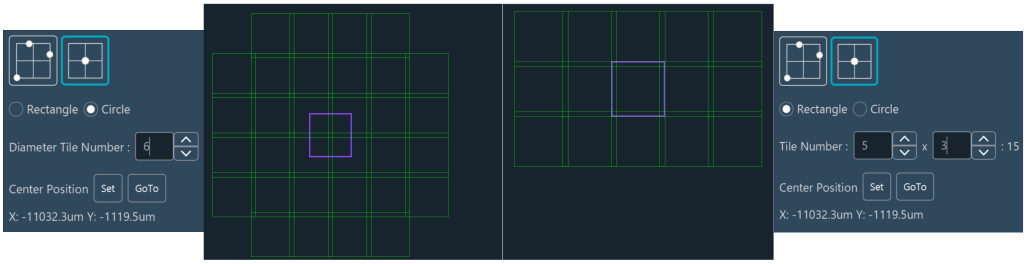
    1. **By Edge**: Click **Add** to drop coordinates outlining the region of interest. The software connects these points to form a polygon.

        - Clear the bounding box using **Remove All**, or delete a single vertex by right-clicking it.



    2. **By Center**: Drive the stage to the center of the acquisition region and click **Set**. Then, select either a rectangular or circular tiling geometry.

        - For a rectangle, specify the bounding grid dimensions (e.g., 3x3 tiles). For a circle, specify the bounding diameter.



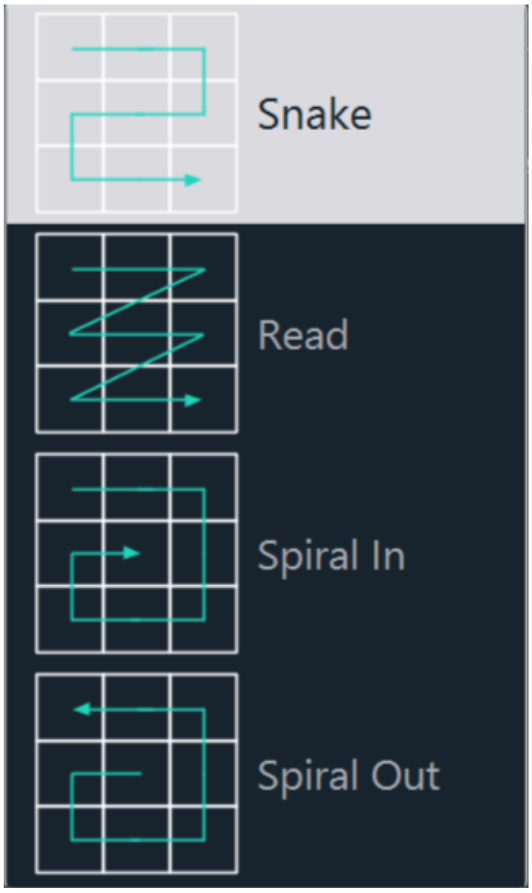
> **Note**
>
> Use the right-hand panel controls to manually jog the hardware stage and manage your defined coordinate vertices.

2. **Define Tile Overlap**: Set the overlap percentage between adjacent frames. A 10% overlap is standard for downstream stitching.

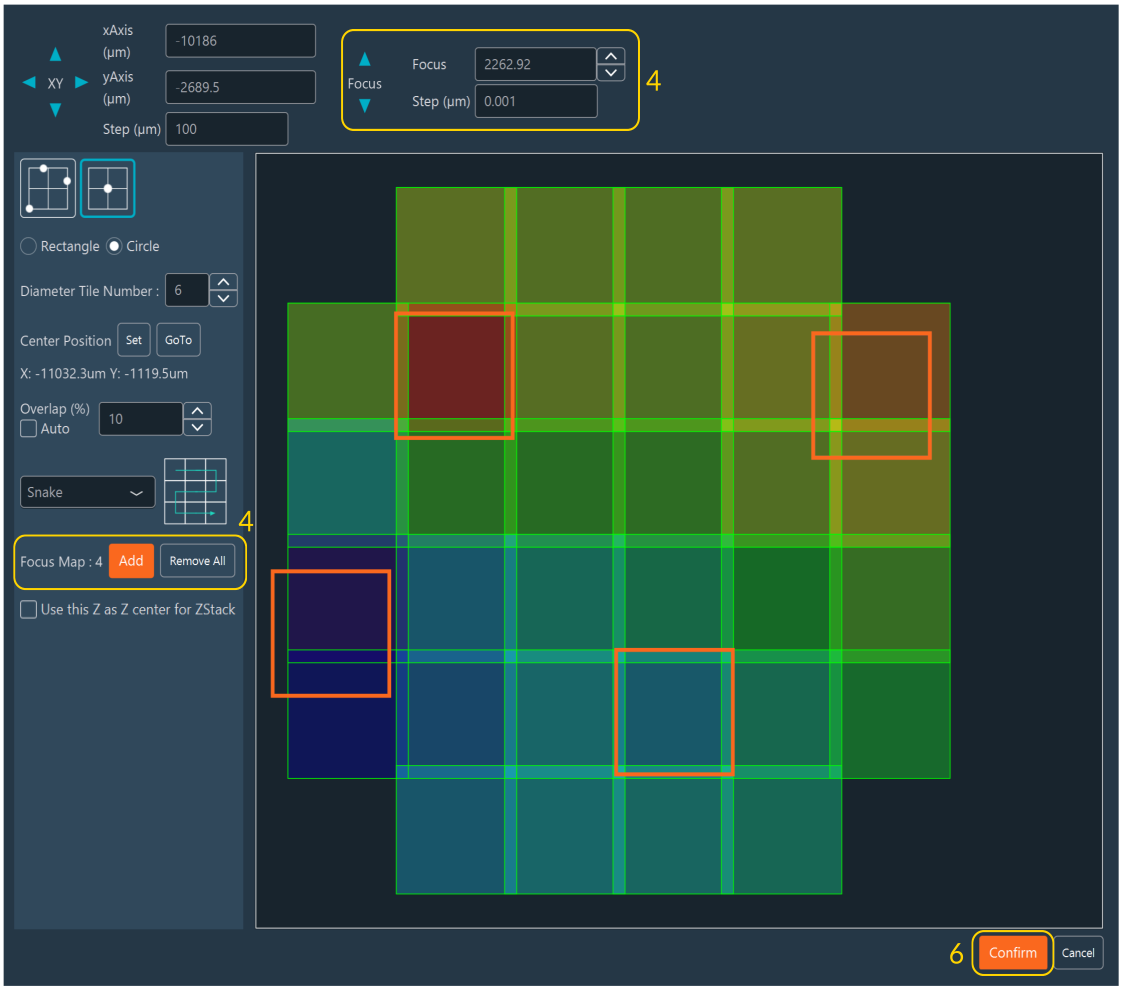
    - Enable **Auto Mode** to let the software calculate the overlap ratio based on your specified coordinate boundaries.
    - To manually dictate the translation distance between individual tiles, enable the **Custom Step** option via the Display Settings menu.

> **Important**
>
> Accurate overlap is necessary for successful post-acquisition image stitching.

3. **Select Read Mode**: Choose the scanning path (e.g., Spiral, Snake) the stage executes during acquisition.



4. **Generate a Focus Map**: To compensate for uneven sample topology across a tiling area, create a Focus Map. Navigate the XY plane using the interface joystick or by clicking within the tiling map. Adjust the Z-focus at various locations, and click **Add** to store each point.
    - Add points as necessary to model the sample's topography. Right-click any point (orange indicator) to remove it.
    - The software applies a color gradient overlay to the map, representing the interpolated Z-surface altitude across the acquisition region.



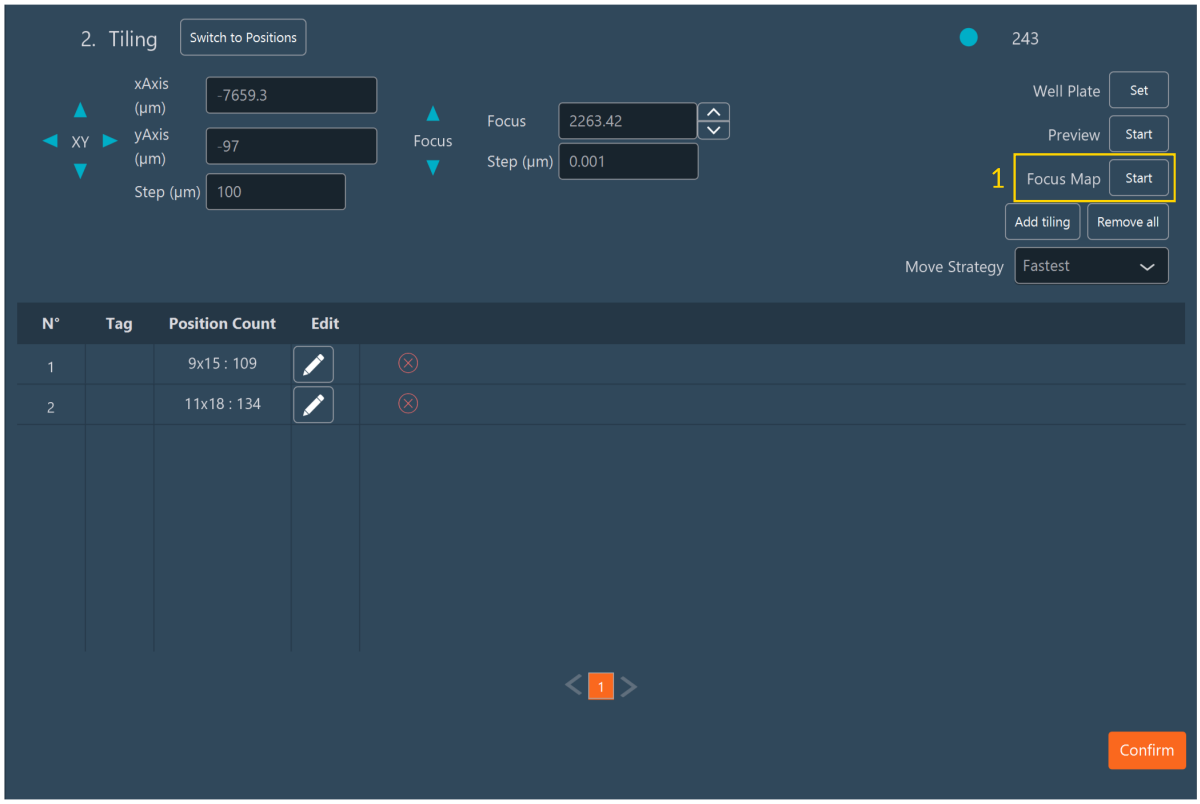
5. **Configure Z-Stack Integration**:
   - Select the corresponding checkbox inside the tiling editor to center Z-Stacks around the interpolated Z-values stored in the Focus Map.
   - If unchecked, the sequence uses either global Z-Stack parameters or the current live Z-position.
6. Click **Confirm** to save the spatial tiling configuration.

> **Note**
>
> - **Orange squares**: Indicate saved boundary vertices or focus map reference anchors.
> - **Purple square**: Indicates the real-time physical position of the XY stage.

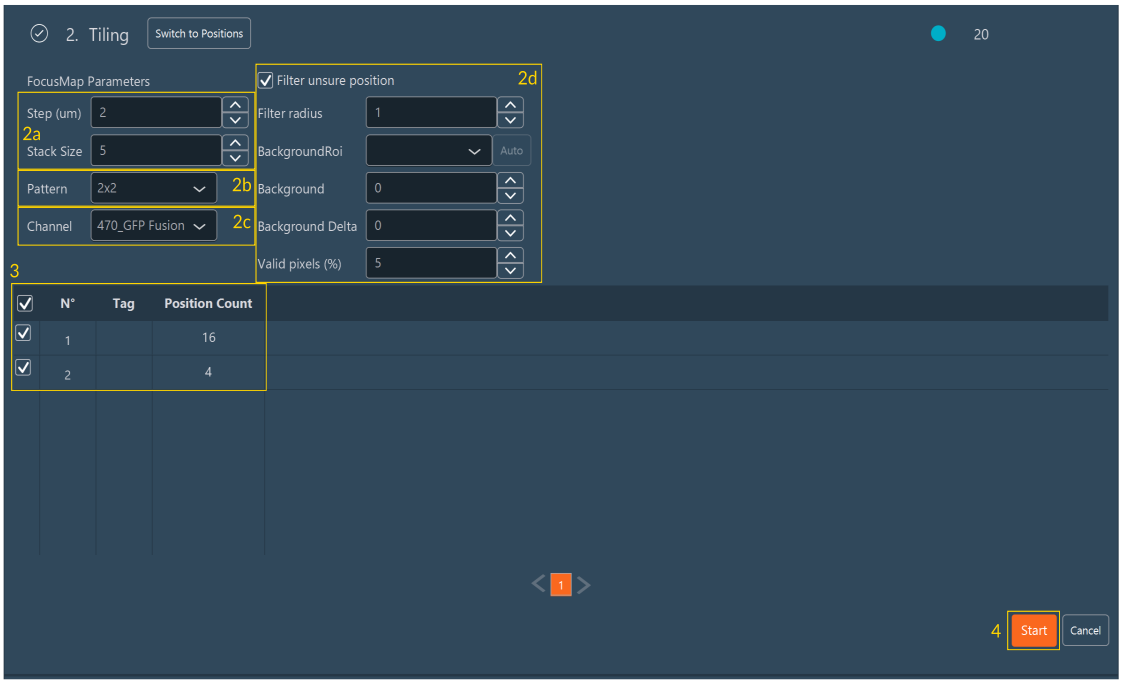
> **Tip**
>
> Use the **GoTo** button to rapidly drive the mechanical stage back to a previously saved topological reference position.

Focus map automated setup for simple and multiple tilings

The Automated Focus Map is an optional software feature of the Inscoper I.S. that automatically generates a relief map of the biological sample when performing tiling imaging.



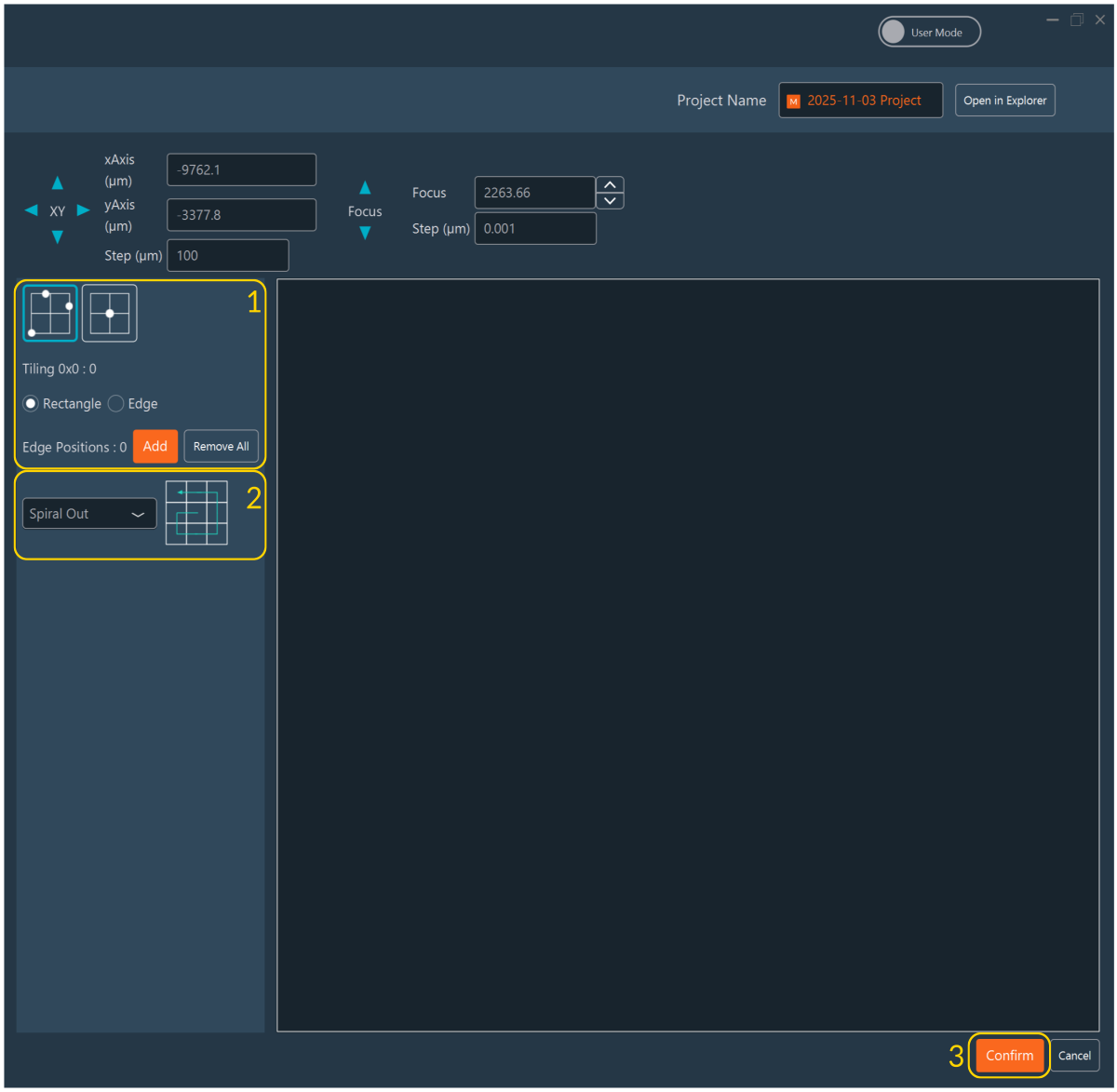
1. Click **Focus Map Start** within the **Tiling** dimension to begin configuration.
2. Set the following parameters for both simple and multiple tilings:
    - a. Select the step and size of the Z-stack.
    - b. Select an existing pattern.
    - c. Select the channel used to acquire the stack.
    - d. *(Optional)* Select **Filter unsure position** and configure the corresponding parameters:
        - i. **Filter Radius**: Smooth noise by replacing each pixel with the median of its neighbors (e.g., 0 = no smoothing, 1 = median of a 3x3 square surrounding the pixel, 2 = median of a 5x5 square surrounding the pixel).
        - ii. **Background ROI**: Select the Region of Interest (ROI) used to automatically calculate background values.
        - iii. **Background**: Define the mean background value.
        - iv. **Background Delta**: Define the acceptable noise amplitude.
        - v. **Valid Pixels**: Define the minimum percentage required to validate a position.
3. Select the tiling region for which to create the Focus Map.
4. Click **Start**. The system scans all points, generates the Focus Map, and displays the determined focal points. A progress bar indicates the current progression per tiling region. Once the Focus Map is complete, the software redirects you to the **Tiling** dimension interface.



5. Click **Confirm** to validate and proceed with the acquisition sequence setup.

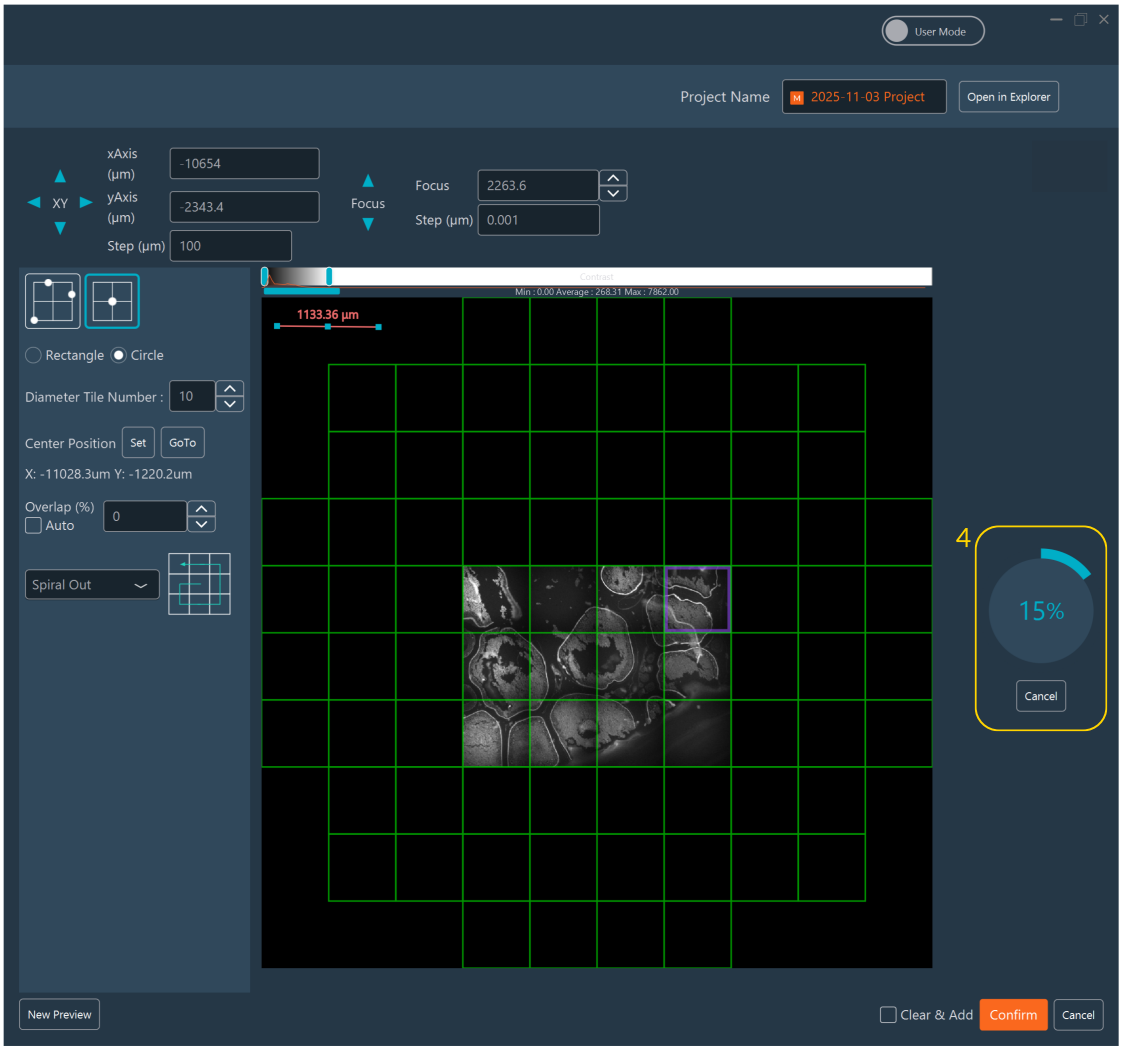
1.4.6 Preview Feature

The **Preview Feature** performs a fast, low-resolution scan of the sample to generate an interactive map. Use this map to identify regions of interest, set up multiposition coordinates, or define mosaic boundaries without exposing the entire sample to high-intensity light or capturing full-resolution data.

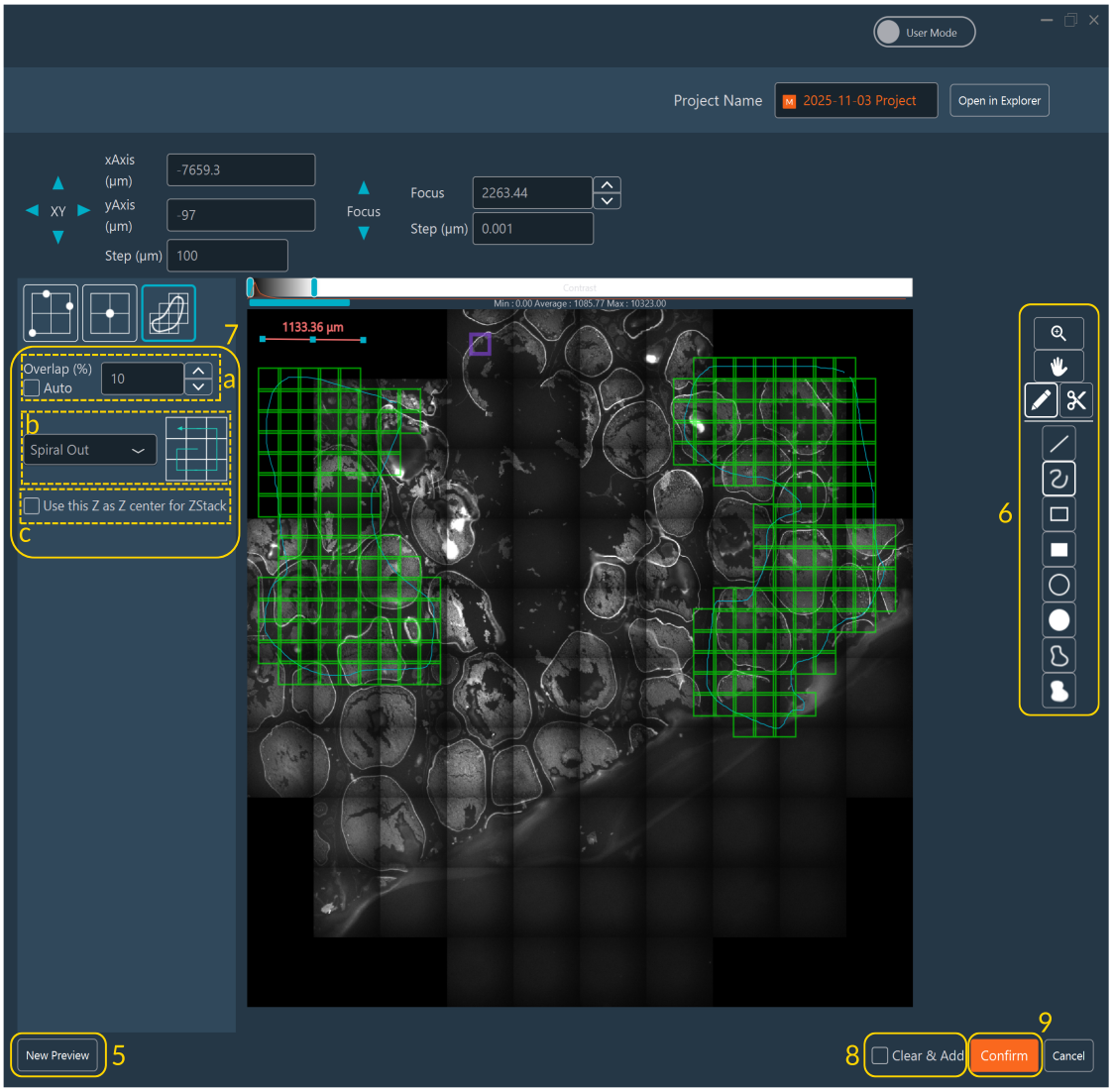


Setting up a Preview Scan

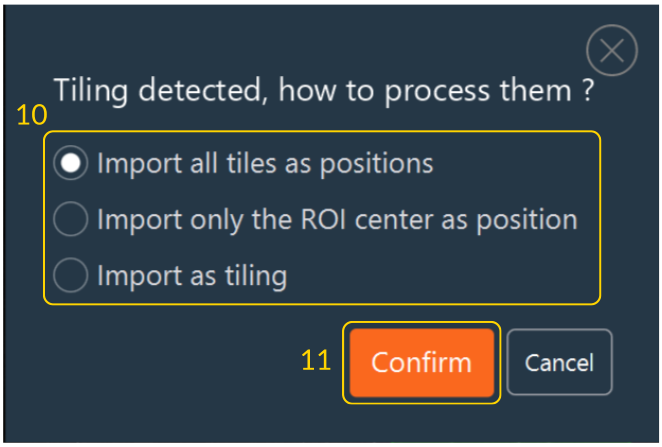
1. Configure the boundaries of the **Preview** area. Similar to the Tiling dimension, you have two setup options:
   - Click **Add** at multiple boundary points while moving the stage to trace a custom polygon outlining your region of interest.
   - Alternatively, click **Set** to define the center coordinate, then specify the bounding grid dimensions (horizontal/vertical tile count).
2. Select a **Read Mode** for how the stage traverses the area (e.g., Spiral, Snake).
3. Click **Confirm** to lock in the configuration.



4. The preview scan begins, rendering the map in real-time. You can abort the acquisition at any time by clicking **Cancel**.

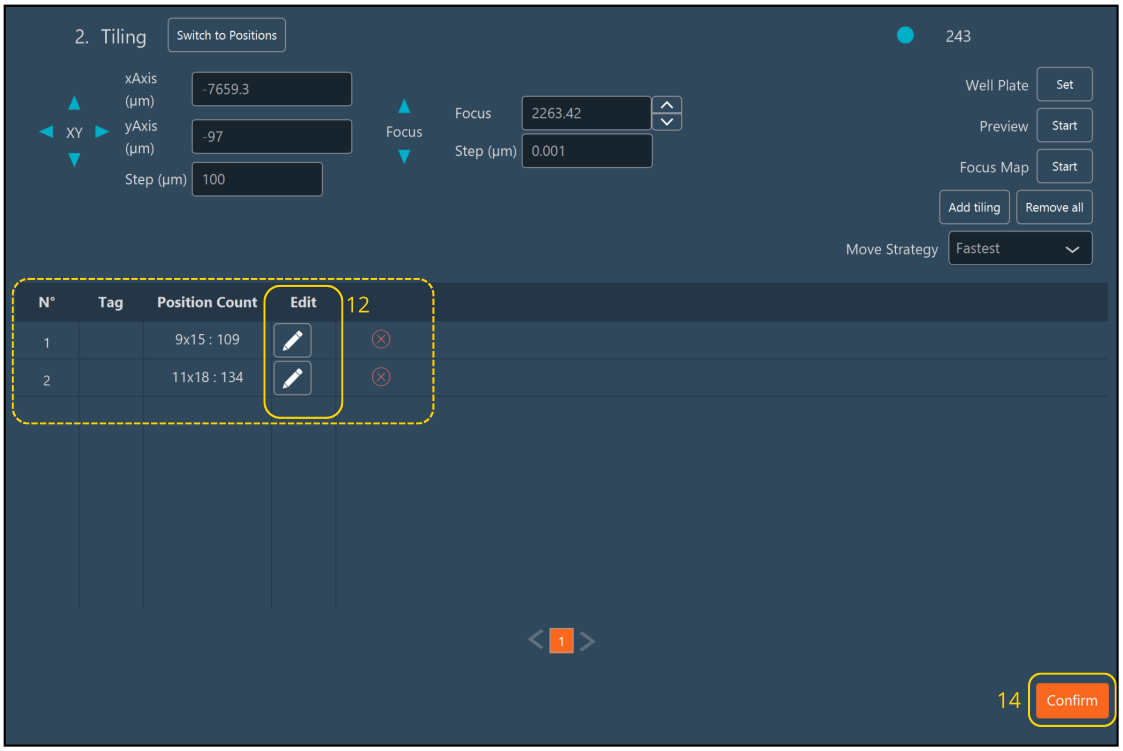


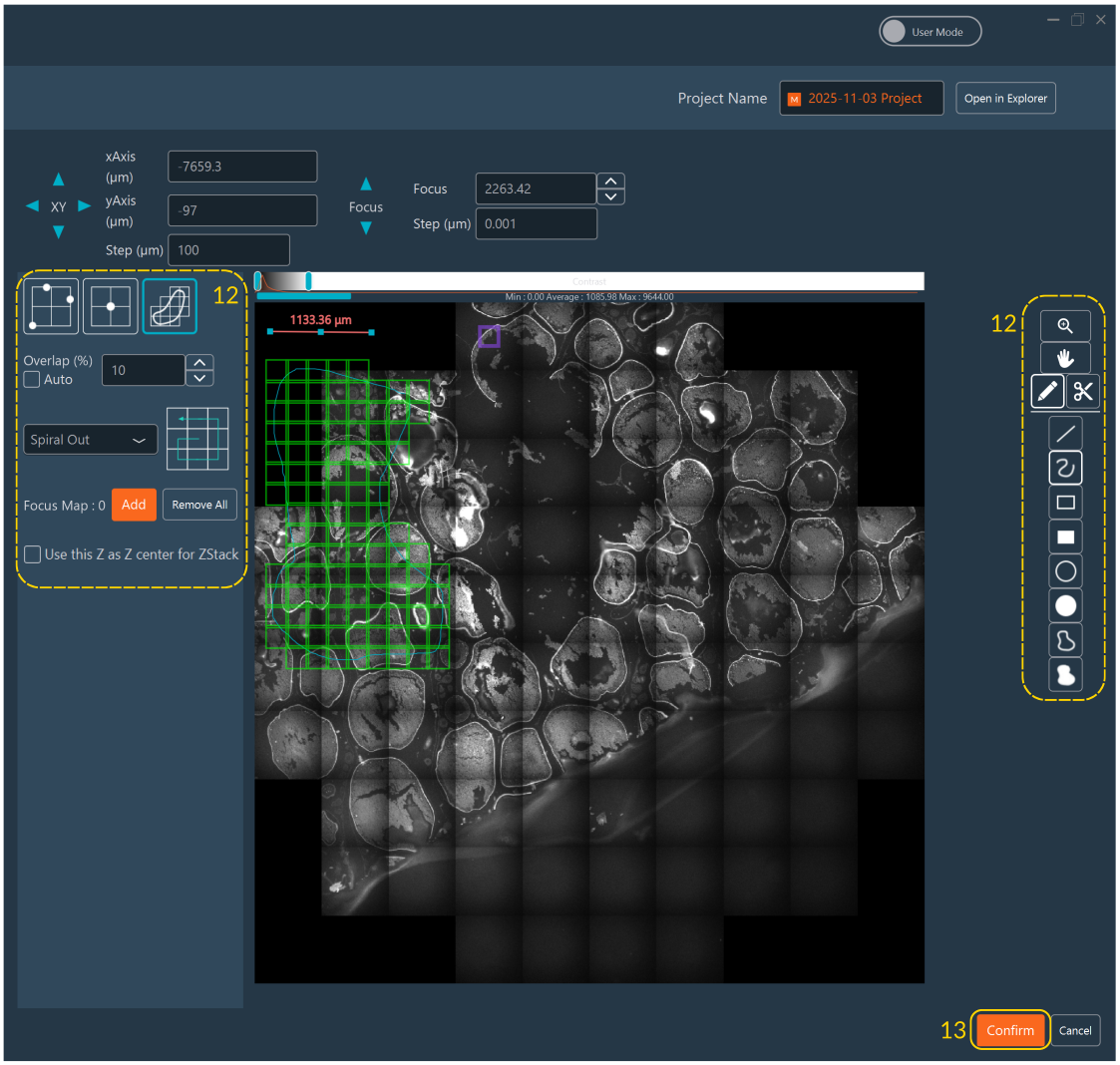
5. Click **New Preview** to discard the current map and start over in a different region.
6. Draw **Regions of Interest (ROIs)** directly onto the map to mark targets. Use the provided tools: **point** (ideal for discrete focal points used in the *Positions dimension*), **freehand**, **circle**, or **rectangle**.
7. Adjust the overlapping tiling settings for the selected ROIs:
    1. Enable **Auto Mode** to let the software calculate the image overlap threshold (available only for *Tiling* imports).
    2. Choose the specific scanning **Read Mode**.
    3. Check the **Z-reference** box to explicitly center subsequent Z-Stacks around the Z-value established during this preview/focus map phase. If unchecked, the system reverts to the global Z-Stack parameters.
8. Check **Clear & Add** if you want these new targets to overwrite previously defined tilings or positions. Leave it unchecked to append them.
9. Click **Confirm** to validate the ROI selections.
10. If you drew ROIs that encompass multiple camera tiles but intend to import them into the **Positions dimension**, the software will prompt you on how to handle the spatial discrepancy:
    - **Import all tiles as positions**: Generates a discrete capture point for every tile covered by the ROI area.
    - **Import only ROI center as position**: Registers a single coordinate at the geometric center of the ROI.
    - **Import as tiling**: Reroutes the coordinate set to the *Tiling dimension* for continuous mosaic acquisition.



11. Click **Confirm** to apply your routing choice. After confirmation, all generated positions or tilings appear in their corresponding dimension tabs.
12. In the Tiling dimension tab, click the **Pen** icon next to an imported entry to edit it against the original preview map:
    - **Move**, **modify**, or **delete** an ROI visually.
    - Change the scanning **Read Mode**.
    - Add a local **Focus Map** within the ROI boundary if your sample is not topologically flat or if you need to precisely adjust the Z-focal plane.

    Select whether to use the Z-position saved during this preview as the absolute center of your Z-Stack. If left unchecked, the Z-Stack will default to the Z-value defined in the standalone Z-Stack Dimension parameters.





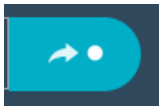
13. Click **Confirm** to save your modifications to the individual ROI.
14. Finally, click **Confirm** in the main panel to validate the entire **Tiling Dimension** sequence.

> **Tip**
>
> To save the visual output of your preview scan as a reference image, **right-click** anywhere on the preview rendering window.

1.4.7 Multiwell Plate Mode

Overview

The **Multiwell Plate Mode** automates the imaging of standardized microplates. Access this feature from either the **Positions** or **Tiling** dimensions by clicking the **Set** button.

Button	Description
	**Select Wells**: Click to highlight the wells you intend to image.
	**Auto Move**: Once the plate is calibrated, activate this toggle to automatically physically move the XY stage when you select a well.
	**Unselected**: The well is excluded from the acquisition.
	**Selected**: The well is included in the acquisition.
	**Current Position**: Indicates the real-time position of the objective relative to the plate.

Calibration

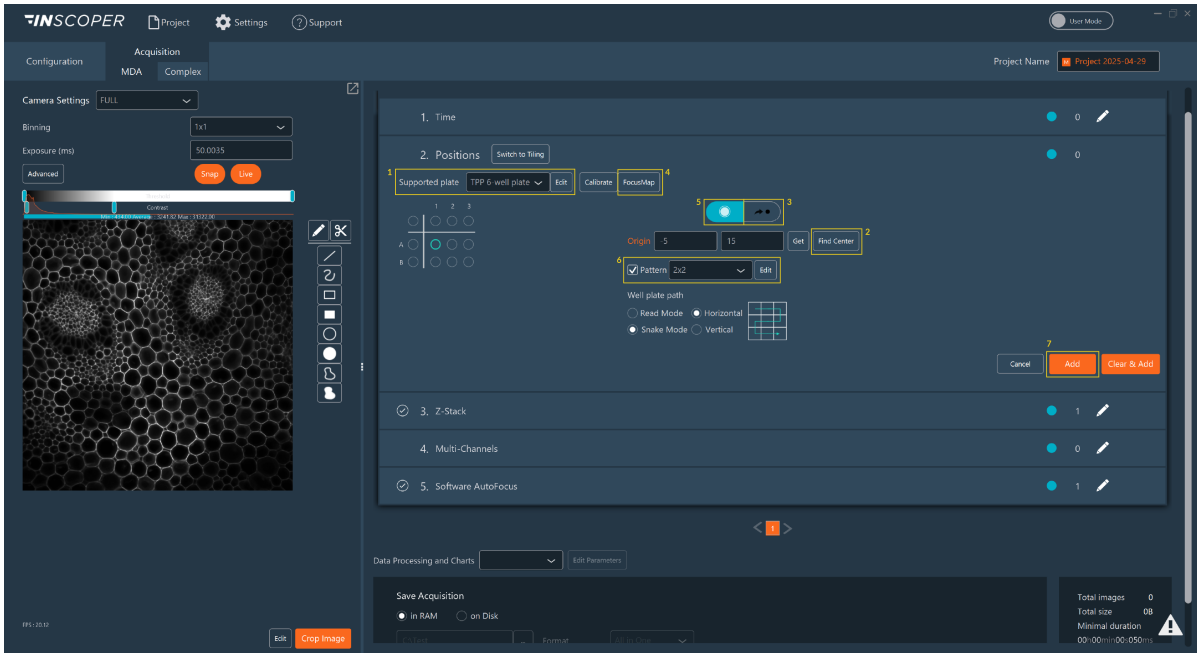
You must calibrate the system to map the software's virtual coordinates to the physical dimensions of the specific well plate, optimizing precise XY stage movement.

1. Click **Find Center** to initiate the semi-automated calibration protocol.
2. Select a well on the virtual map to serve as the calibration reference.
3. Manually drive the mechanical stage to this physical well, if not already positioned there.
4. While viewing the live image, move the XY stage to align the virtual crosshair precisely with the physical edge of the well, then click **Add**.
5. Repeat this alignment process two more times on different sides (e.g., top, bottom, left) of the same well to establish the geometry.
6. Click **Confirm** to complete the calibration.

Once calibration is complete, the software reliably and automatically navigates to any well on the plate.

Multiwell Mode in the Positions Dimension

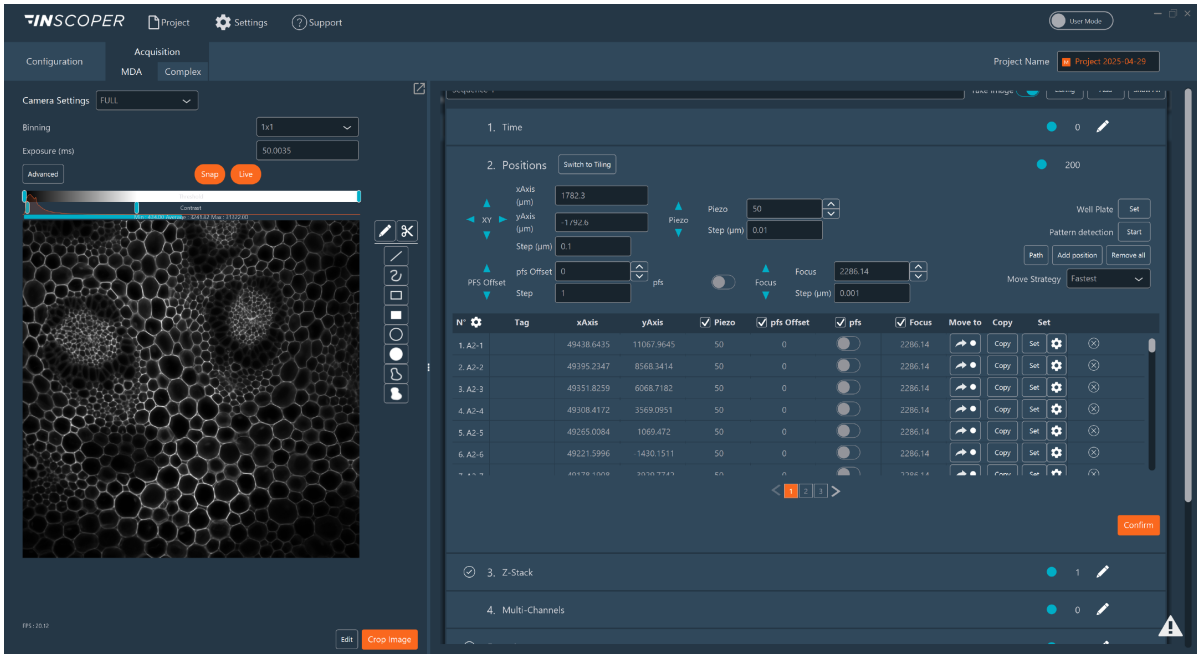
Use this mode to capture distinct, single-field-of-view images within designated wells.



1. Select the plate format (e.g., 96-well, 384-well) from the **Supported plate** dropdown list.
2. Calibrate the well plate using the **Find Center** tool.
3. You can now automatically move the stage directly to specific wells.
4. Create a **Focus Map** for the desired wells, if necessary (crucial for maintaining focus across uneven plate bottoms).
5. Select the wells to image on the virtual map.
6. To generate an automated imaging pattern within each well, check the **Pattern** box. Specify the number of desired capture points and the minimum Euclidean distance between them.
7. Click **Add** to confirm the settings and populate the position list.

> **Tip**
>
> **Fast Selection**: In the virtual well plate map, click a row letter or column number to select the entire row or column. You can also click the top-left corner to select the entire plate. To select multiple discontinuous wells, click and drag your mouse to create a selection box.

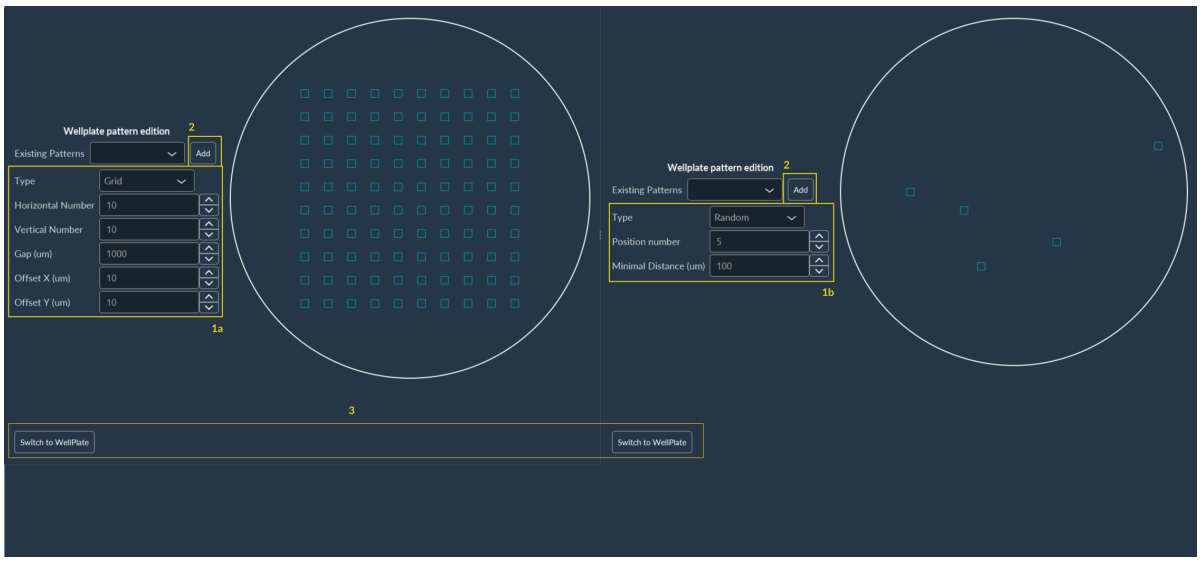
All generated coordinates are added to the Positions dimension list.



EDITING A PATTERN

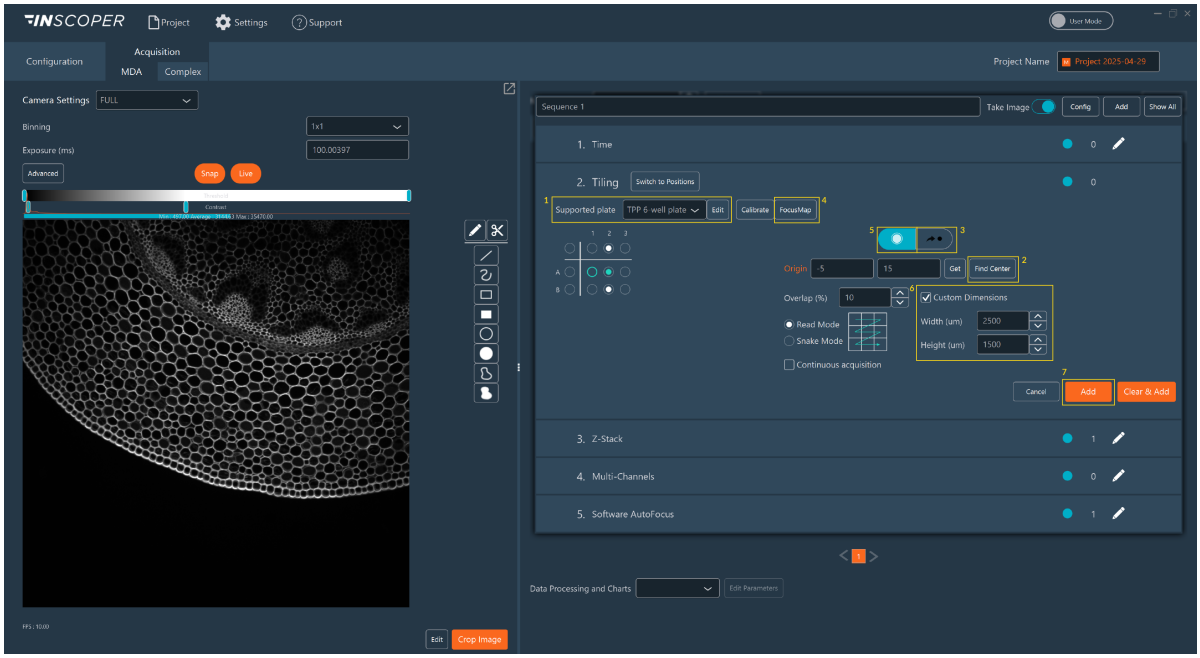
Click **Edit** next to a pattern to modify its distribution parameters:

1. Select the pattern strategy: **Grid** or **Random**.
   - For a **Grid**, specify the number of horizontal and vertical images, the spacing distance between adjacent positions, and the X/Y offset (to translate the entire grid relative to the well center).
   - For a **Random** pattern, specify the total number of positions and the minimum required exclusion distance between them to prevent overlap.
2. Click **Add** to update the pattern.
3. Click **Switch to WellPlate** to return to the interactive plate overview.



Multiwell Mode in the Tiling Dimension

Use this mode to acquire large, stitched mosaic images covering entire wells or significant portions of them.

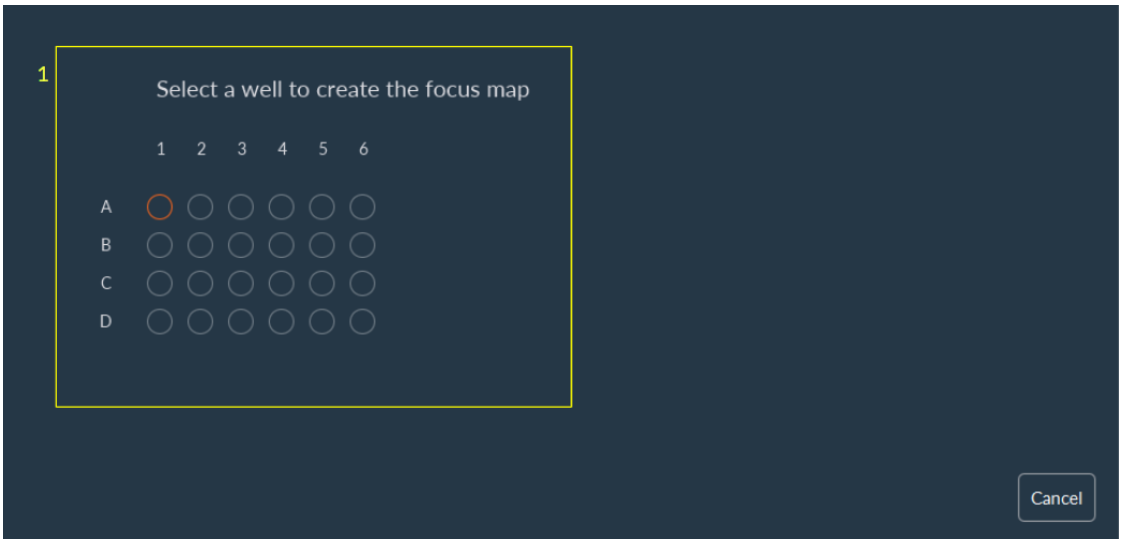


1. Select your plate format from the **Supported plate** dropdown list.
2. Calibrate the well plate using the **Find Center** tool.
3. You can now automatically move the stage exactly to specific wells.
4. Create a **Focus Map** for the desired wells, if necessary.
5. Select the wells you intend to image on the virtual map.
6. Check the **Custom Dimensions** box to configure the specific tiling size, tile overlap percentage, and scanning read mode to be applied uniformly within each selected well.
7. Click **Add** to confirm your settings and generate the tiling matrices.

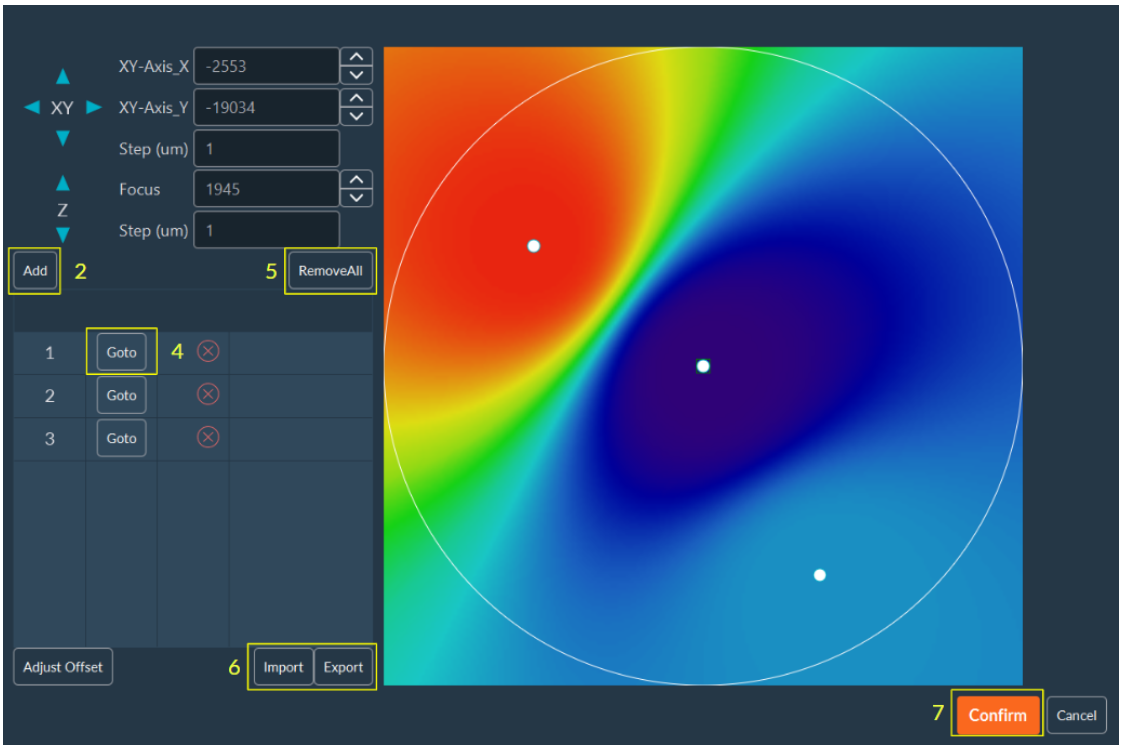
Manual Focus Map Setup

A Focus Map creates a topological Z-surface to compensate for sample drift or uneven plate substrates.

1. Select the specific well for which you want to create a Focus Map.



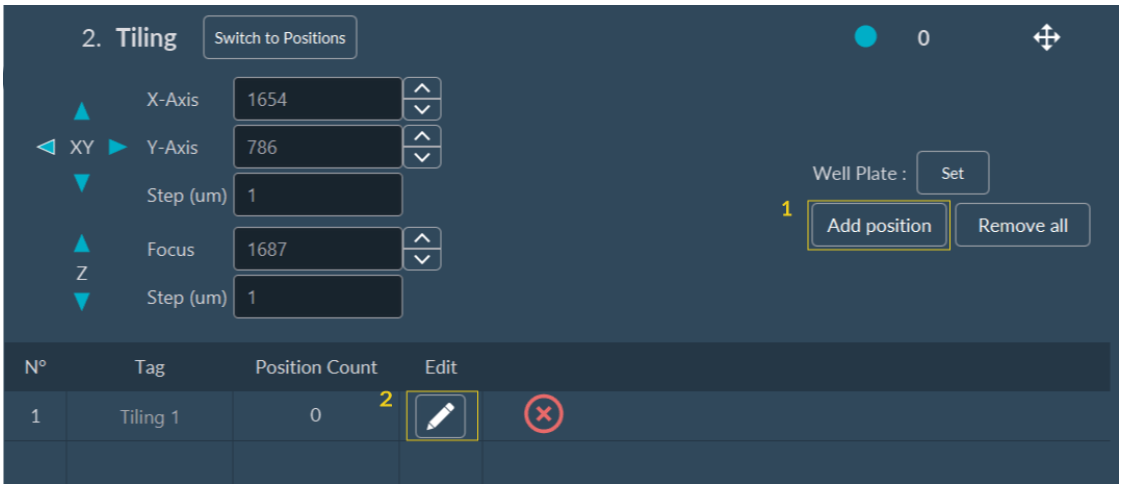
2. Move the stage to a desired topological anchor point within the well, manually adjust the focus until sharp, and click **Add**.
3. Repeat this process at multiple disparate points until the Focus Map adequately covers the imaging area. The software interpolates the focus between these points.
4. Click **Go to** to move the stage back to a previously saved focus anchor point for verification.
5. Click **Remove all** to clear all points from the current map.
6. You can also import or export a saved Focus Map using the respective buttons (useful for standardized, recurring assays).
7. Click **Confirm** to lock in your Focus Map.

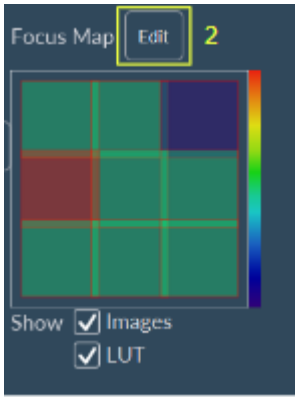


Automated Focus Map Setup

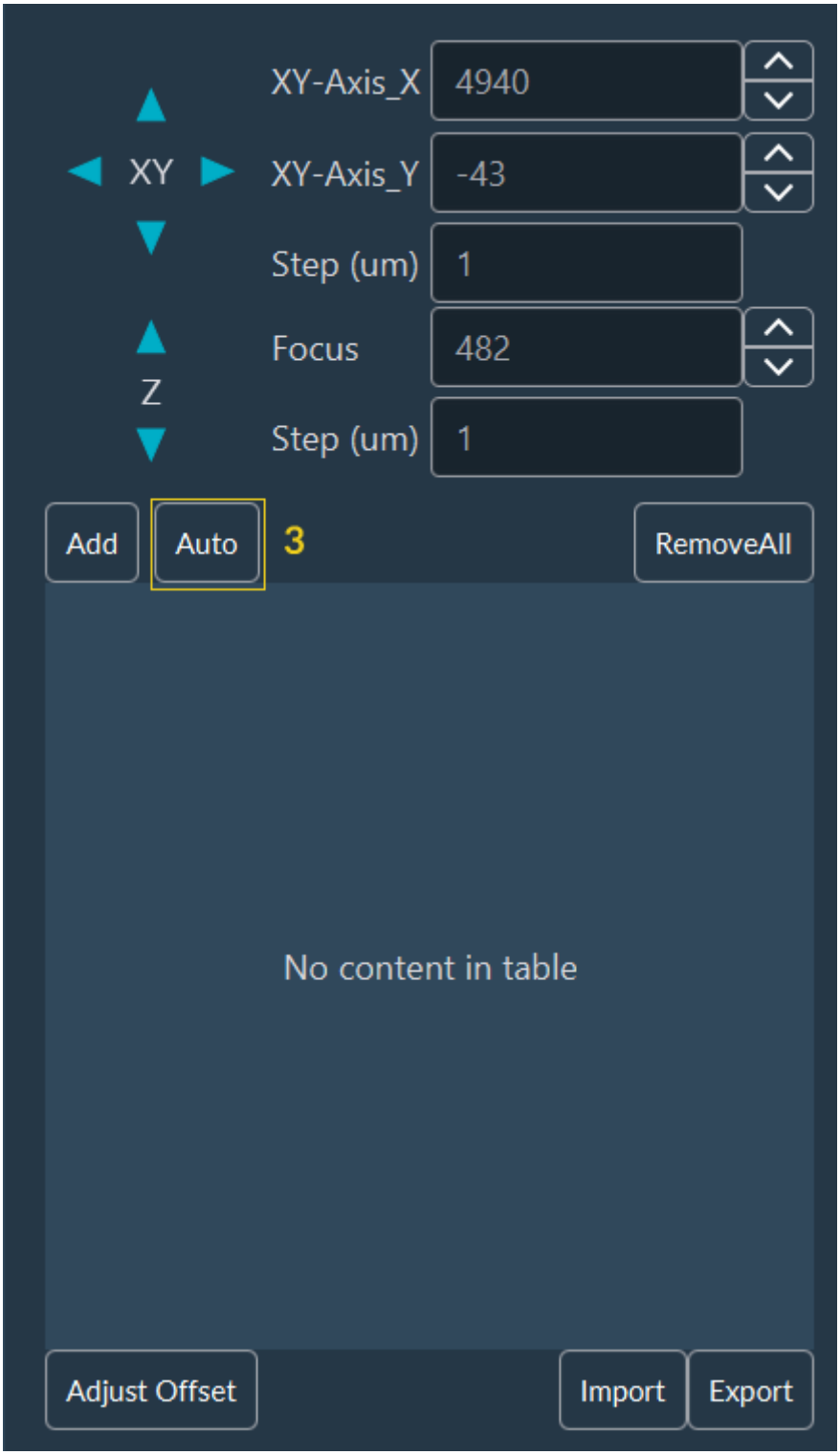
The **Automated Focus Map** is a software feature of the Inscoper I.S. that automatically generates a hardware-driven relief map of the biological sample before executing tiling imaging.

1. In the Tiling dimension, click **Add position** to configure the tiling matrix.
2. Click **Edit** to customize the tiling parameters and adjust the Focus Map.

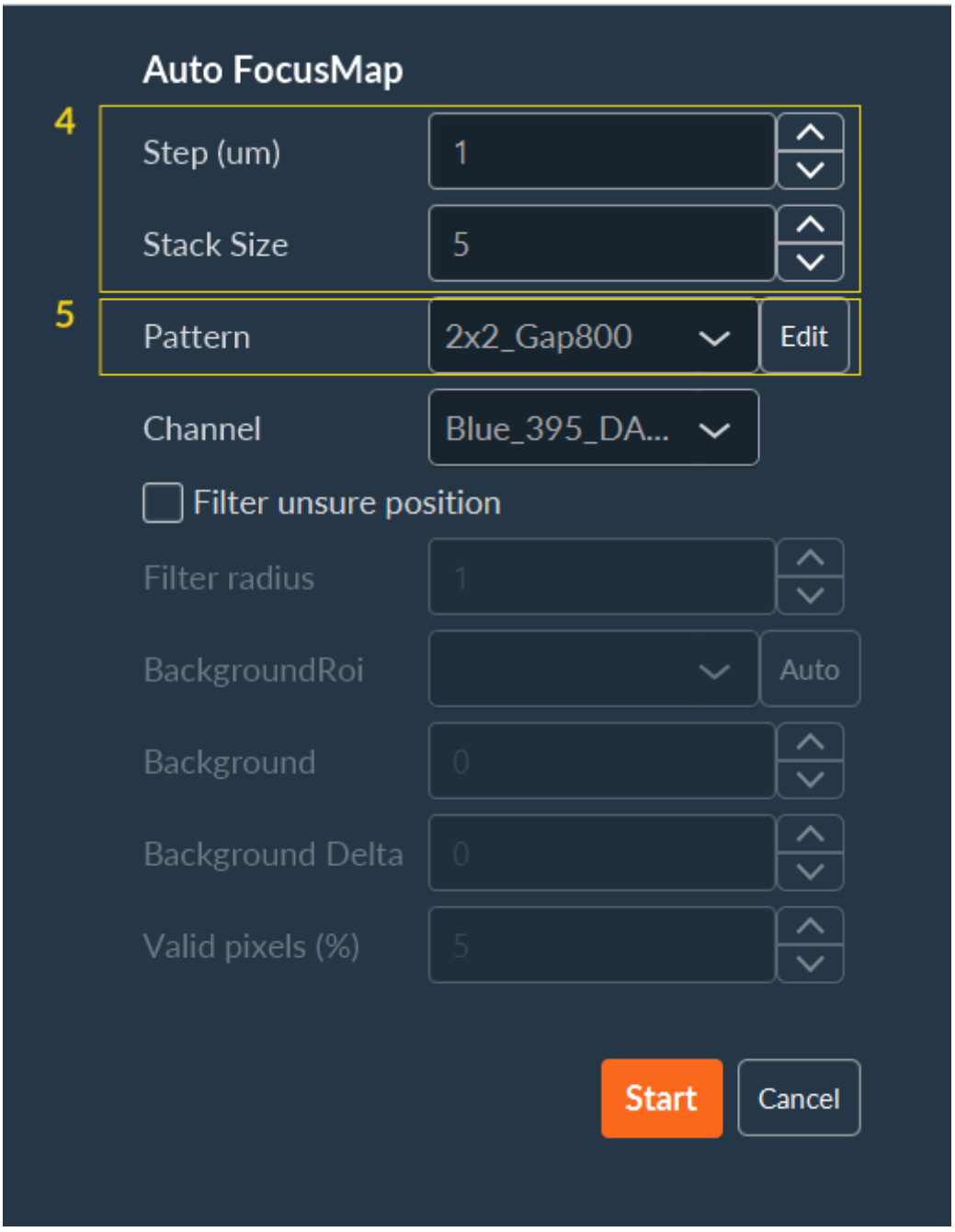




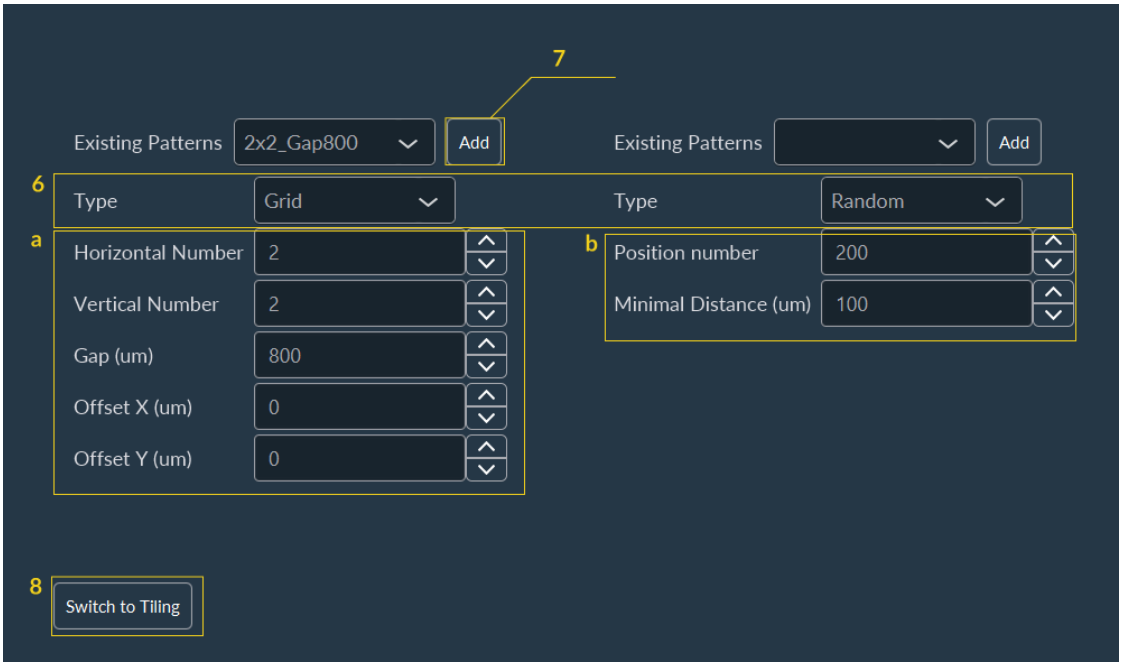
3. Click **Auto** to configure the automated Focus Map generation.



4. Define the **step** interval and **size** for the Z-stack.
5. Select an existing sampling pattern from the dropdown list (and proceed to step 9), or click **Edit** to define a new spatial pattern.



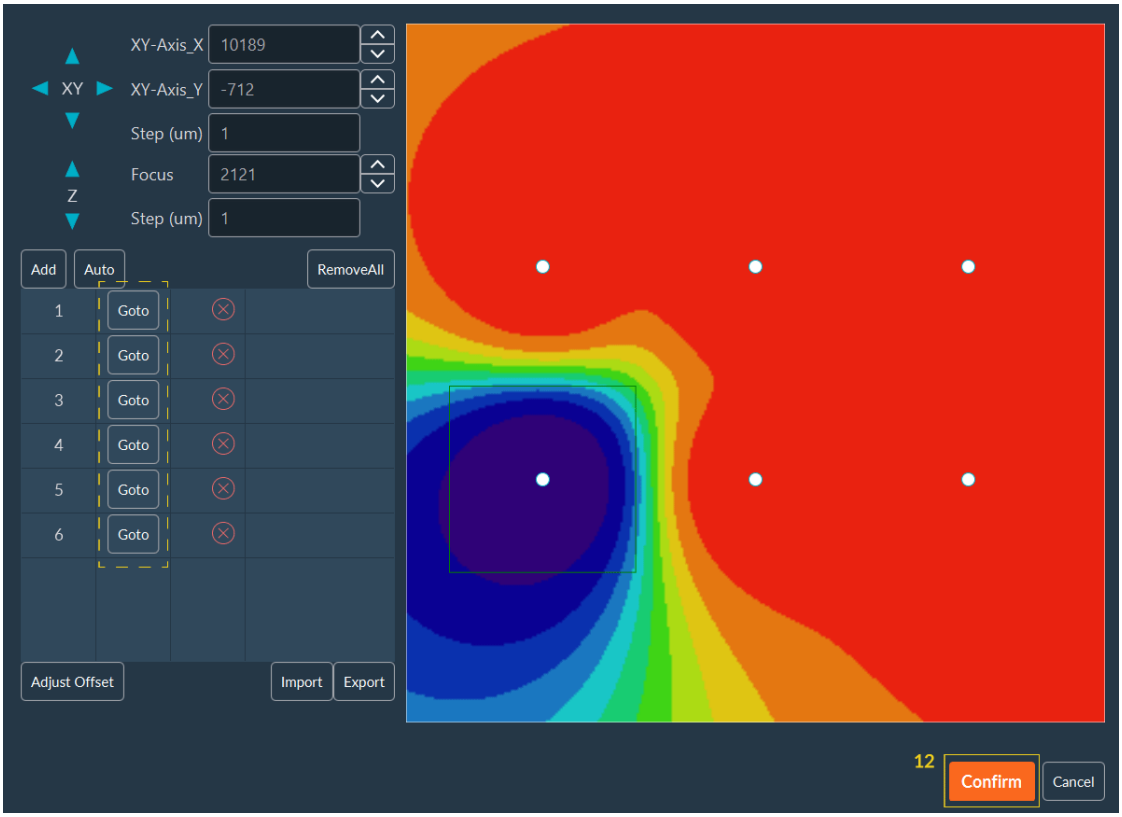
6. If creating a new pattern, select either a **Grid** or **Random** geometric distribution:
    - **Grid:** Specify the total number of images to acquire horizontally and vertically, along with the gap distance between adjacent image boundaries. The X and Y offset values translate the entire pattern grid across the well.
    - **Random:** Specify the total number of positions and the minimum required exclusion distance between coordinate points.
7. Click **Add** to save the distribution pattern.
8. Click **Switch to Tiling** to return to the primary dimension configuration.



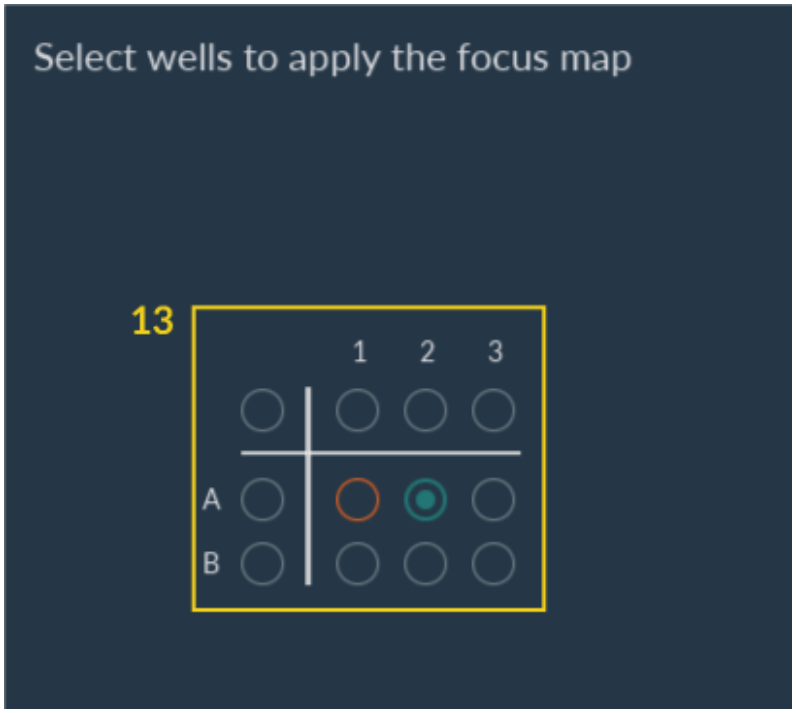
9. Select the **channel** designated for acquiring the Z-stack during the Focus Map generation.
10. To mathematically reject optical artifacts or empty fields, select the **Filter unsure position** checkbox and configure the rejection parameters:
    - **Filter Radius:** Applies a spatial median filter to reduce high-frequency noise before evaluation (e.g., 0 = no smoothing; 1 = median based on a 3x3 pixel kernel; 2 = median based on a 5x5 pixel kernel).
    - **Background ROI:** Automatically calculates absolute *Background* and *Background Delta* values by evaluating a user-drawn Region of Interest (ROI). The *Background* equals the mean pixel intensity of the ROI, and the *Background Delta* equals two times the standard deviation.
    - **Background:** Specifies a static mean background intensity value.
    - **Background Delta:** Specifies the permitted noise amplitude threshold.
    - **Valid Pixels:** Sets the required percentage of pixels whose intensity must exceed the background threshold to validate the coordinate position.
11. Click **Start** to initiate system generation.

The system navigates to all defined coordinates to create the Focus Map and displays a list of validated anchor points. Click **GoTo** adjacent to the coordinate or click directly on the thumbnail projection to verify physical focus at any anchor point.

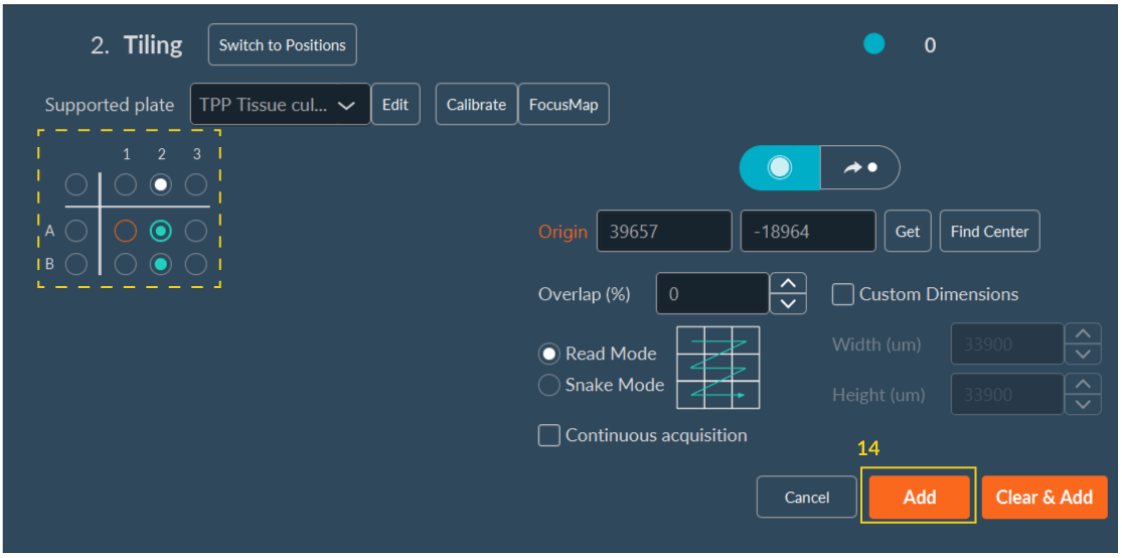
12. Click **Confirm** to finalize the generated Focus Map.



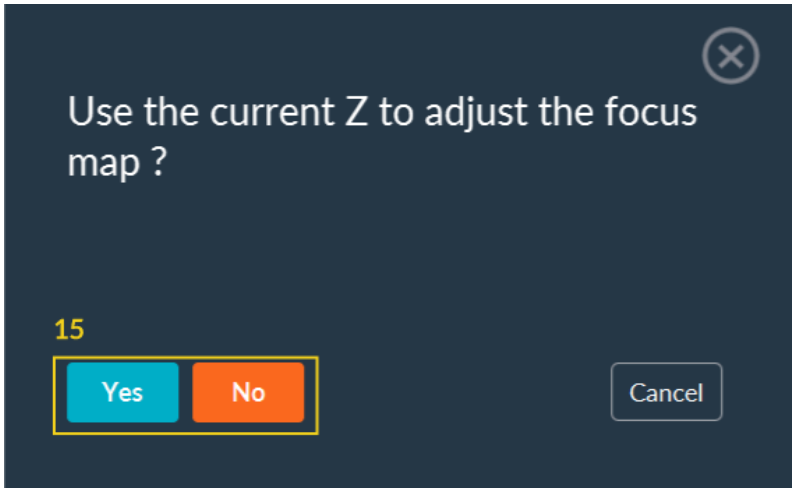
13. Select the wells to apply the Focus Map to, then click **Confirm**.



14. Select the wells to acquire, then click **Add**.



15. Select whether to use the current Z-position to adjust the Focus Map.



After this step, you can continue configuring your acquisition sequence.

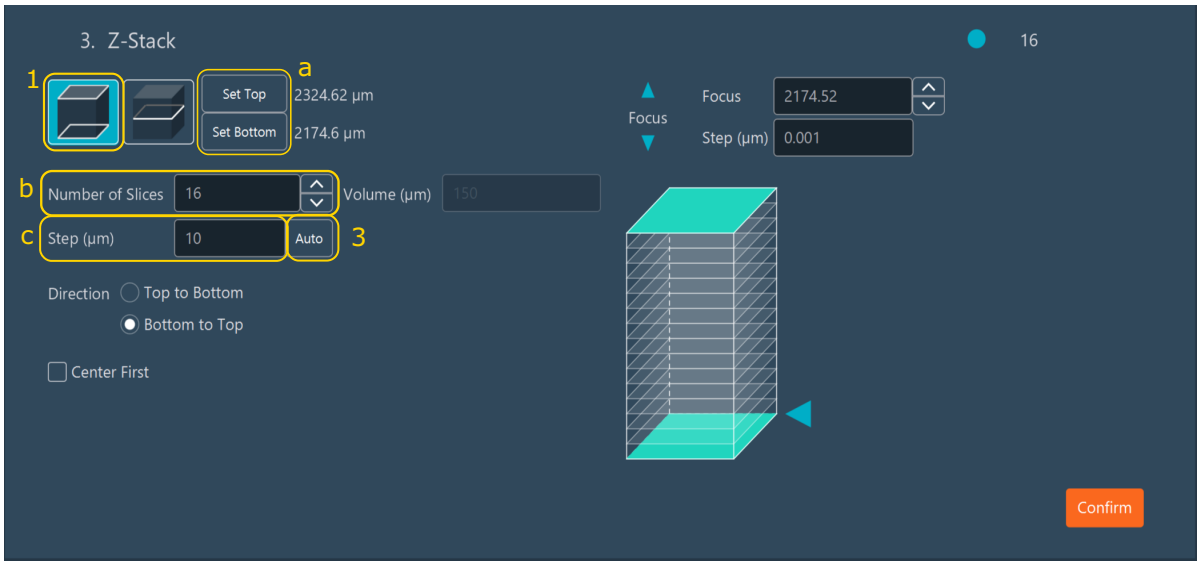
1.4.8 Z-stack

Use the **Z-Stack** dimension to capture volumetric data by acquiring a series of slices along the Z-axis.

Configuring a Z-Stack

Configure the volumetric boundaries using one of two primary methods:

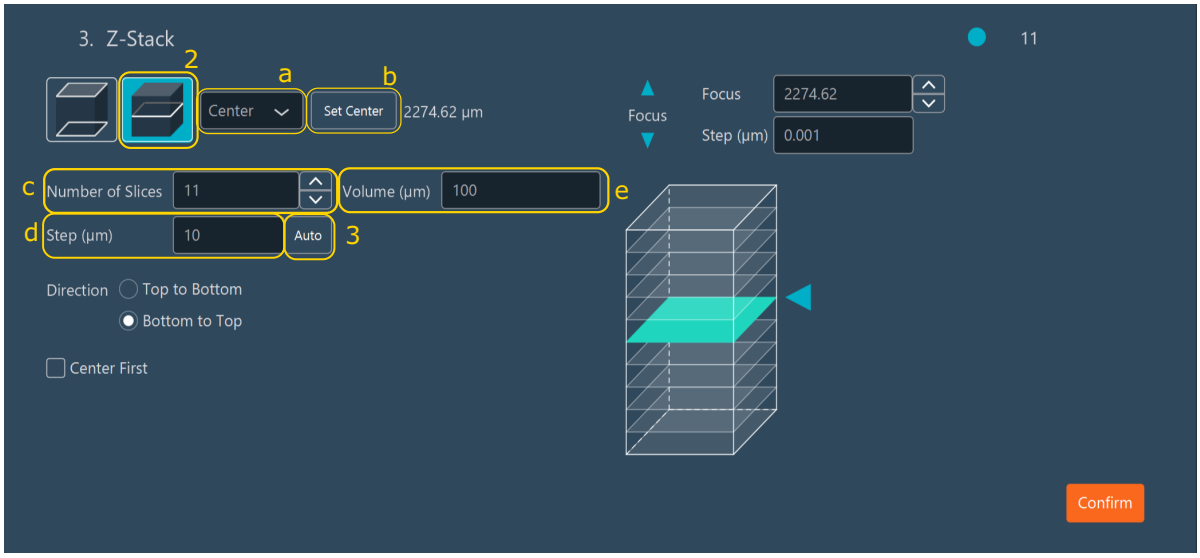
DEFINE TOP AND BOTTOM LIMITS



Move the focus to the desired upper and lower focal limits (1):

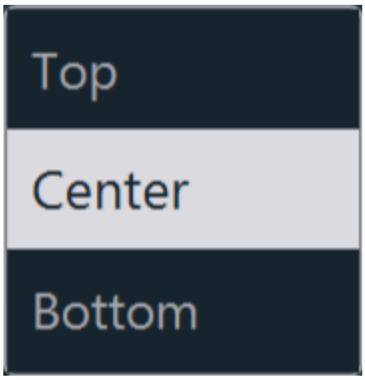
- a. click **Set Top** and **Set Bottom** to establish the boundaries.
- b. Specify the **number of slices** (focal planes) within this volume.
- c. Set the **step size** (in µm) between adjacent focal planes. Enter your own value or click **Auto** (3) to use Nyquist Calculation.

DEFINE VIA A REFERENCE PLANE



Move the focus to the desired center focal plan (2):

- a. Select whether the current live focal point represents the **center**, **top**, or **bottom** of the intended Z-stack.

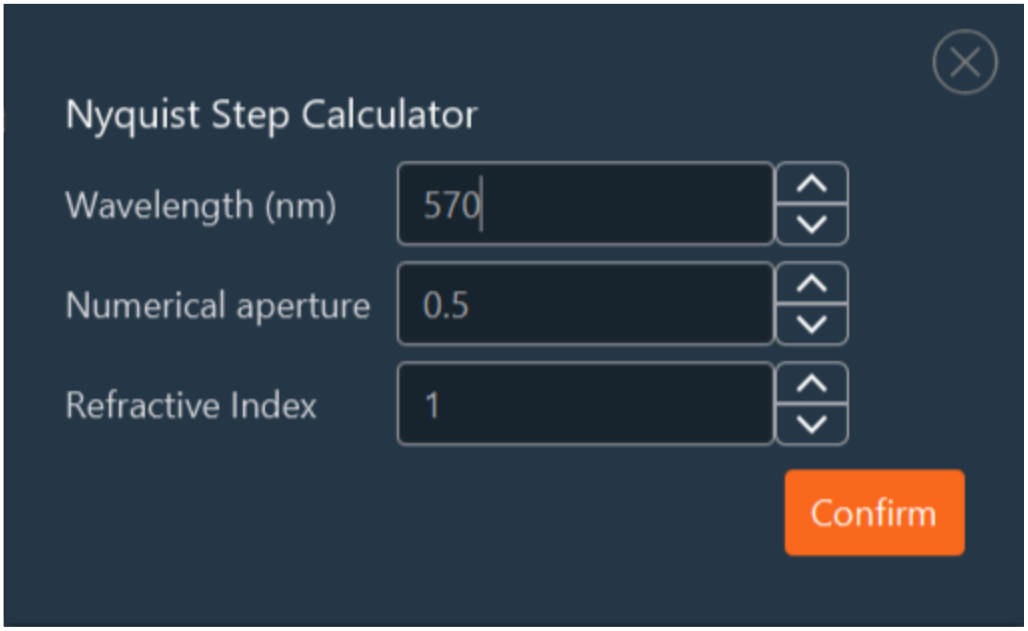


- b. Click the corresponding registration button (**Set Center**, **Set Top**, or **Set Bottom**) to log the current focal plane as the anchor.
- c. Specify the total **number of slices**.
- d. Set the **step size** (in µm). Enter your own value or click **Auto** (3) to use Nyquist Calculation.
- e. Alternatively, define the **volume depth** of the Z-stack (in µm). The software computes the remaining variables automatically.

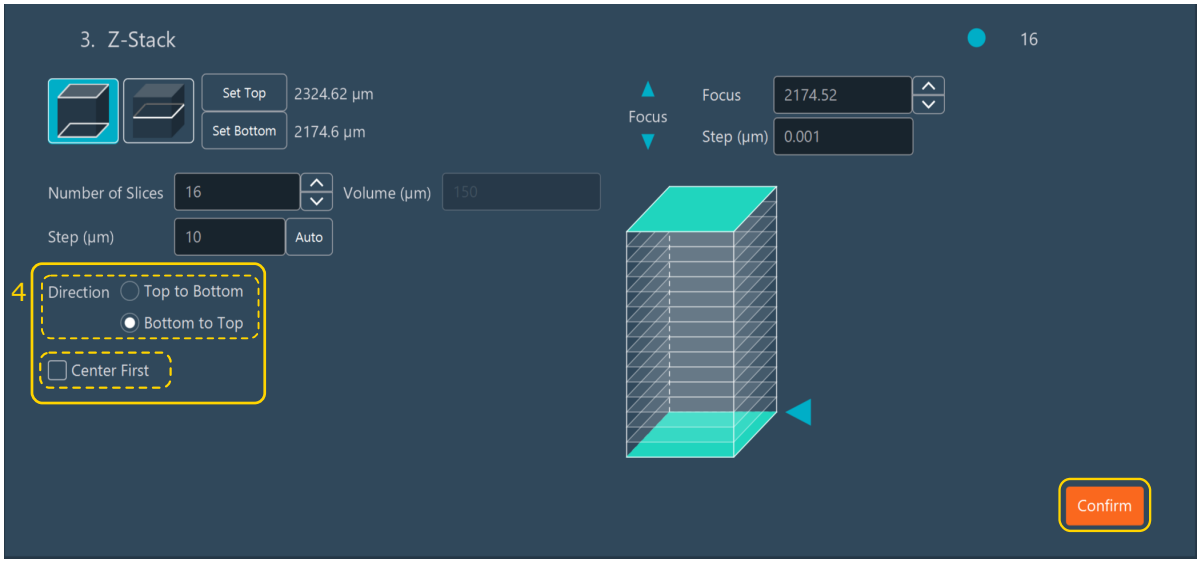
NYQUIST CALCULATION

Use this method to automatically compute the optimal step size for maximum 3D optical resolution without oversampling the volumetric data. Click **Auto** to generate the ideal Nyquist Z-sampling rate.

- The software calculates the ideal physical step size based on the **longest emission wavelength** configured in the multi-channel parameters.
- These values remain **editable**. Manually override the calculated step size or tune the inputs, such as the emission wavelength, the objective's numerical aperture (NA), and the refractive index (RI) of the immersion medium.
- Click **Confirm** to save the computational settings.



Execution settings



Once the geometric volume is defined, set the execution parameters (4):

- Select the **acquisition direction**: either **Top to Bottom**, or **Bottom to Top**.
- Enable **Center First** to force the sequence to begin acquisition at the **central focal plane**.

> **Tip**
>
> This setting is required when utilizing hardware-based autofocus systems (e.g., PFS, Definite Focus) to ensure they anchor correctly before sweeping the volume.

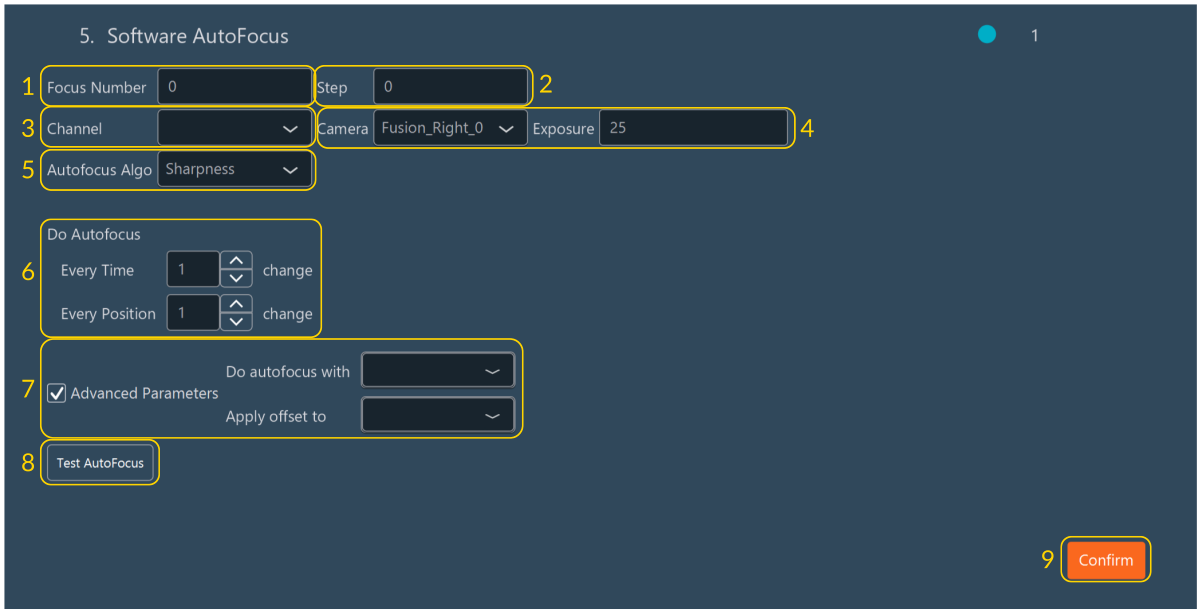
Click **Confirm** to save the Z-stack configuration block.

> **Tip**
>
> Use the interactive **Z-stack diagram** to verify the setup — drag the **blue arrow** or click the diagram to drive the stage and preview the stack bounds.

> **Important**
>
> When configuring a Z-stack from a center reference plane, the **Volume**, **Step**, and **Number of Slices** fields are mathematically linked. Adjusting one automatically recalculates the others. If the **Positions dimension** is active and the Z-axis override is enabled for a specific position, that local spatial focus value overrides the global center coordinate of the Z-stack.

1.4.9 Software Autofocus

The Inscoper I.S. software autofocus mode keeps samples in focus throughout the acquisition sequence.

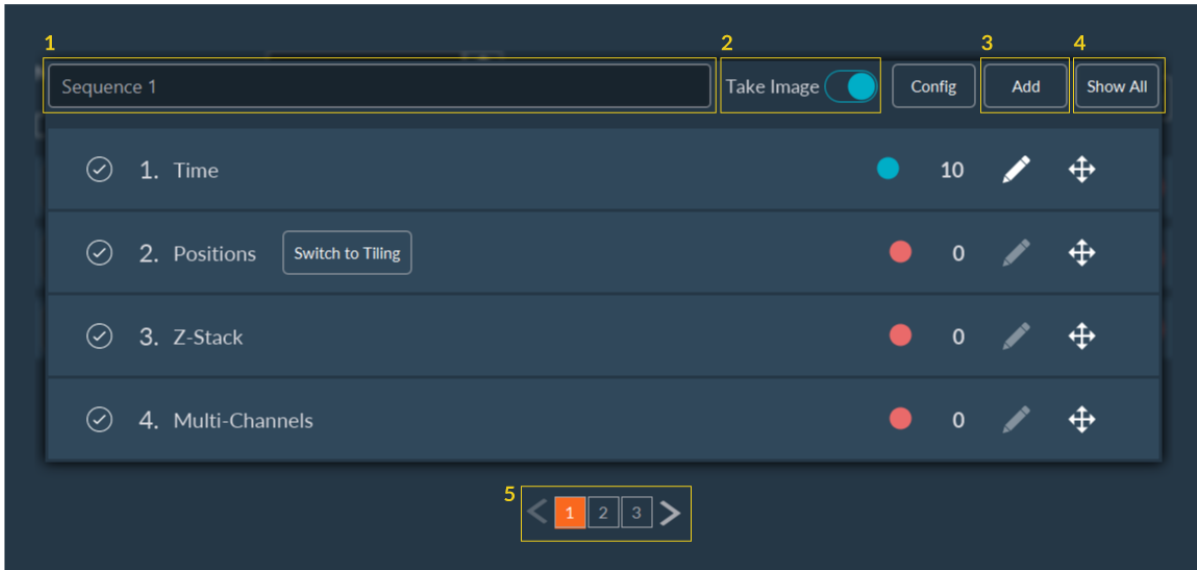


Configuring Software Autofocus

1. Specify the number of autofocus **attempts** per position (this defines how many Z-planes the algorithm analyzes).
2. Define the **step size** (in µm) between these focus iterations.
3. Choose the active optical **channel** used to evaluate focus.
4. Select the active **camera** and define the sensor **exposure time** for the autofocus capture frames.
5. Choose the autofocus algorithm:
    - **Maximal Intensity**: Optimal for bright fluorescent signals.
    - **Sharpness**: Optimal for dense morphological structures (e.g., brightfield/phase contrast), where the algorithm detects the highest high-frequency contrast.
6. **Add Condition**: Define precisely when the autofocus sequence triggers based on the acquisition dimensions (e.g., execute autofocus at *every position* but only *every 3 time points*).
7. Open **Advanced Parameters** to configure hardware integration:
    - Specify which mechanical device drives the autofocus routine (e.g., a fast piezo stage vs. the microscope's internal Z-drive).
    - *(Optional)* Apply **Z-offset** found by the autofocus on a mechanical device selected.
8. Test the designed autofocus protocol on a live sample before initiating the complete sequence.
9. Click **Confirm** to lock in the settings.

1.4.10 Multi-Sequence Acquisition

After configuring an isolated acquisition sequence, launch it immediately or append additional sequences to build a **Multi-Sequence Acquisition** for automated workflows.



Each sequence within the interface includes the following setup options:

1. **Rename**: Assign a name to the sequence for identification during the experiment.
2. **Toggle Image Acquisition**: Disable or enable image capture for a specific sequence by toggling the **Take Image** button (useful for setting up delays or hardware moves without saving data).
3. **Add Sequence**: Append a new, empty sequence to the multi-sequence stack.
4. **Show All**: View a summary of all prepared sequences in a bottom panel.
5. **Navigation**: Jump between sequences. The currently selected sequence highlights in orange.

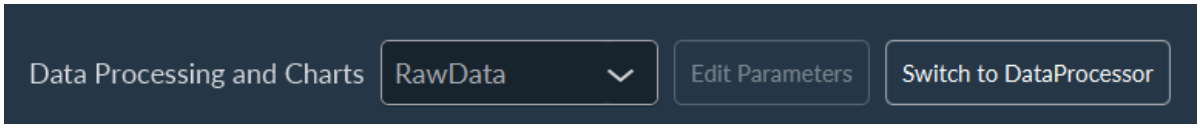
Clicking **Show All** (4) opens the **Summary Tab**, providing sequence management:

1. **Loop Count**: Define how many times the entire multi-sequence stack should loop.
2. **Intermittent Sequences** *(Optional)*: Check this box to trigger the sequence intermittently based on a selected dimension (e.g., execute this sequence only on every 3rd time point).
3. **Take Image Toggle**: Quickly enable or disable image capture for individual sequences from the summary view.
4. **Active Status**: Activate or deactivate an entire sequence by clicking its colored indicator (red indicates inactive, blue indicates active).
5. **Edit**: Click the **Pen** icon to open the configuration for that sequence.
6. **Duplicate**: Clone an existing sequence by clicking **Copy**.
7. **Delete**: Remove a sequence by clicking **Delete**.

1.4.11 Data Processor

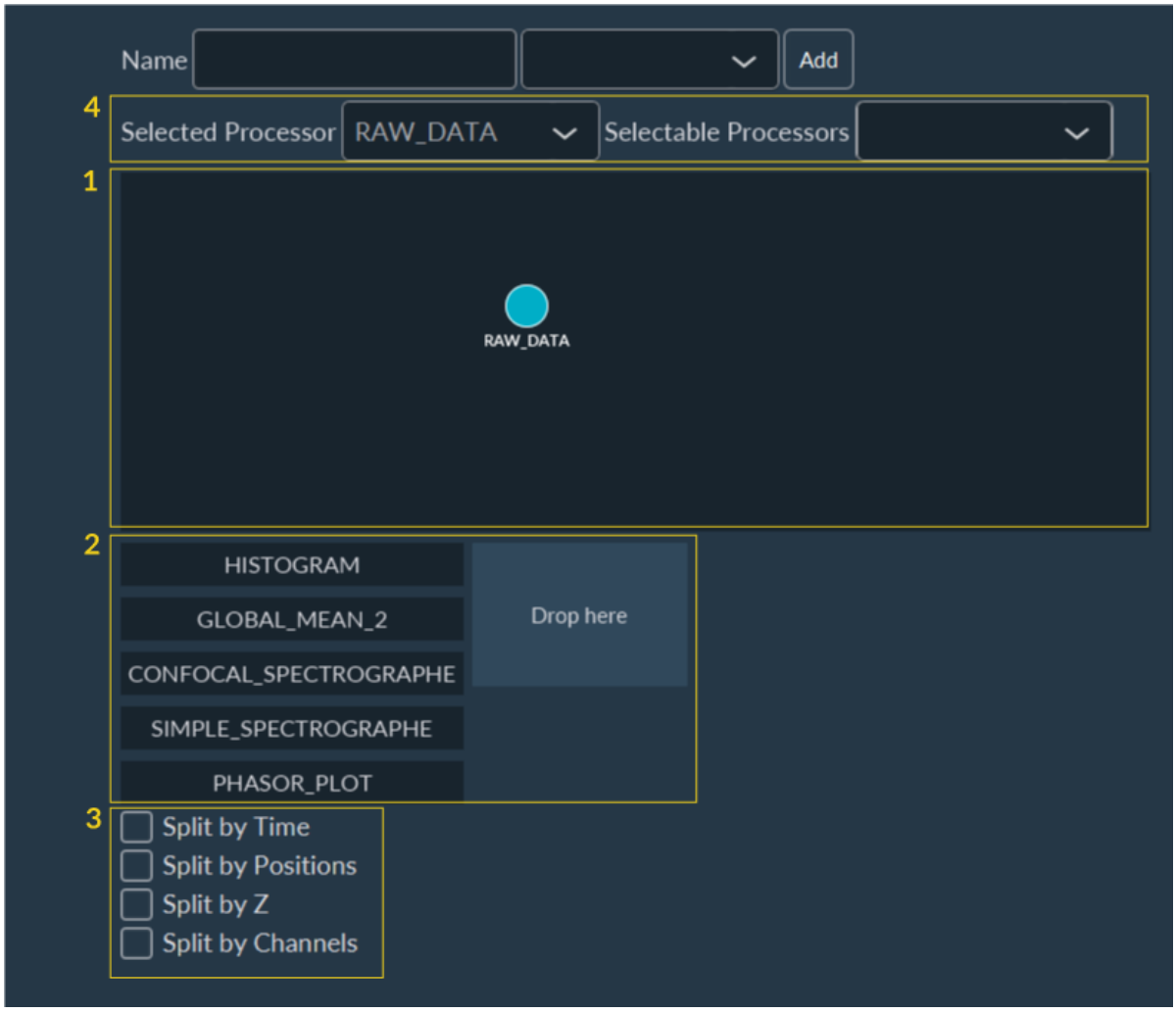
Use the **Data Processor** to customize data visualization and analysis pipelines during and after image acquisition. View raw image streams, monitor fluorescence intensity graphs, or apply post-processing.

Interface Description



In **User Mode**, access pre-configured data processing pipelines. Select a pipeline from the drop-down menu and begin the acquisition.

In **Expert Mode**, click the **Switch to Data Processor** button to build custom data processing workflows.

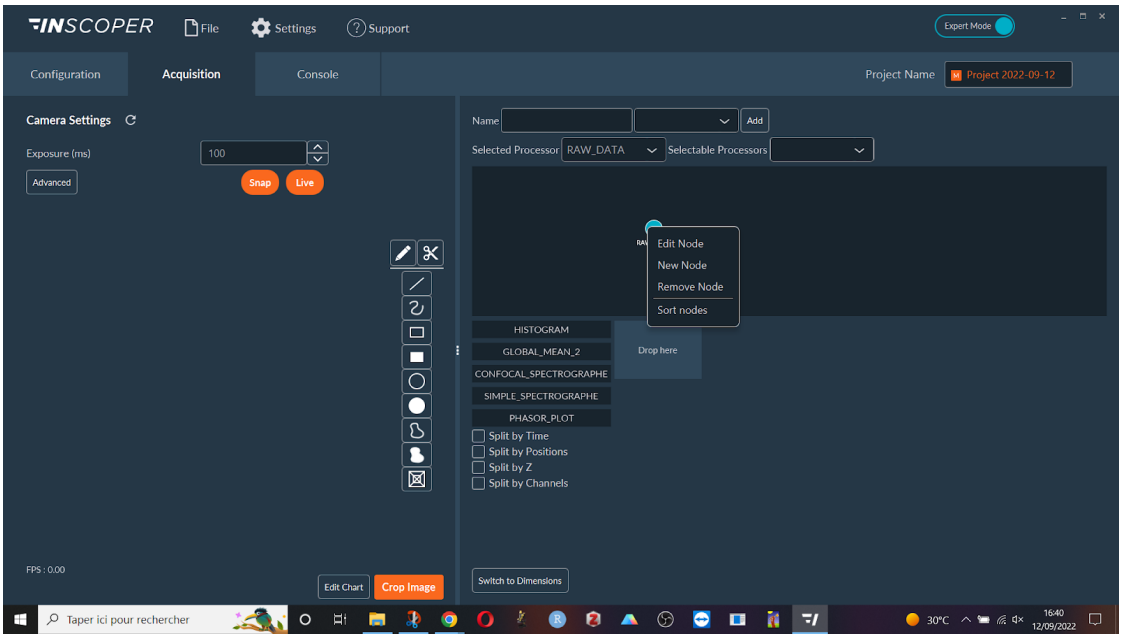


1. **Workspace**: The canvas for creating and routing data processing workflows.
2. **Visualization Tools**: Controls for customizing how data is displayed.
3. **Dimension Analysis Tools**: Tools for analyzing data separated by dimension (e.g., assessing Z-stacks or time-lapses).
4. **Image Visualization Customization**: Controls for viewing images post-processing:
    - **Selected Processor**: Choose which specific processing node's output to view live during acquisition.
    - **Selectable Processor**: Choose which node outputs will remain available for review in the visualization tab after the acquisition concludes. By default, the system selects all processing steps.

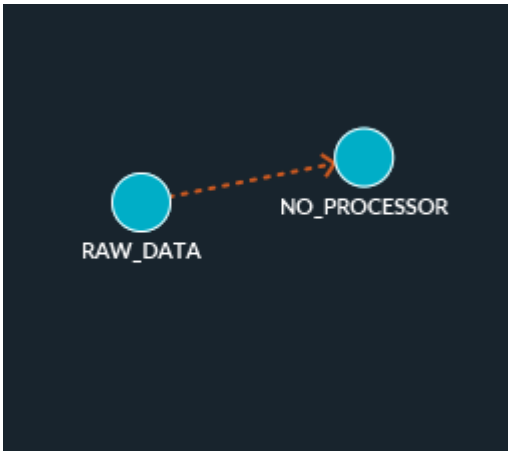
> **Note**
>
> The available processing options depend on the system's specific hardware and license configuration.

Creating a New Data Processing Workflow

1. Every workflow originates with the raw images, represented by a root node named **RawData**. To process these images, create a new node by right-clicking **RawData** and selecting **New Node**.

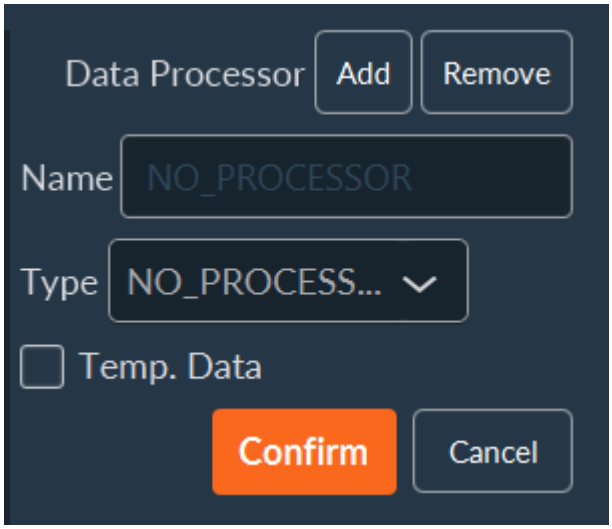


2. The system generates a new, empty node. To assign a processing operation to this node, double-click it, or right-click and select **Edit Node**.



> **Tip**
>
> For improved readability, right-click the workspace and select **Sort Nodes** to automatically arrange the pipeline. You can also manually reposition nodes by dragging and dropping them.

3. In the node properties window, use the **Type** drop-down menu to select the desired processing algorithm (e.g., Stitching, Maximum Projection, Background Subtraction).
    - Check the **Temp. Data** box to prevent saving the output of this specific step to disk. This reduces processing overhead and minimizes storage space requirements.



AVAILABLE PROCESSING OPERATIONS

Processor	Description
`NO_PROCESSOR`	Passes the data through without modifying it.
`SIMPLE_TILING`	Positions each image at its coordinates within a tiling grid.
`STITCH_TILING`	Stitches adjacent images of a tiling acquisition by blending overlap regions. *(Note: Requires* `SIMPLE_TILING` *to be applied first).*
`STANDARD_DEVIATION_ON_FLY`	Calculates pixel intensity standard deviation across an image stack.
`SHADING_CORRECTION`	Corrects uneven illumination and optical artifacts (e.g., sensor dust).
`FILTER`	Reduces noise by removing anomalous pixels (despeckling).
`TIME_MAX`	Performs a maximum intensity projection across the Time dimension.
`FOCUS_MAX`	Performs a maximum intensity projection across the Z-stack dimension.
`TIME_AVERAGE`	Calculates the average pixel intensity across the Time dimension.
`FOCUS_AVERAGE`	Calculates the average pixel intensity across the Z-stack dimension.
`CHANNEL_MULTICOLOR`	Merges images from multiple channels into a multi-color composite.
`SUBTRACT_BACKGROUND`	Subtracts the background signal to improve image contrast.
`CHANNEL_RATIO`	Calculates the ratio of intensities between channels to measure relative changes.
`MULTI_CHANNEL_MERGE`	Merges channels natively (primarily used in SPIM workflows).

Tip

You can execute all processing steps either during or after acquisition. When run during the acquisition, the processed data updates continuously and saves directly to the final output file.

Customizing Data Visualization

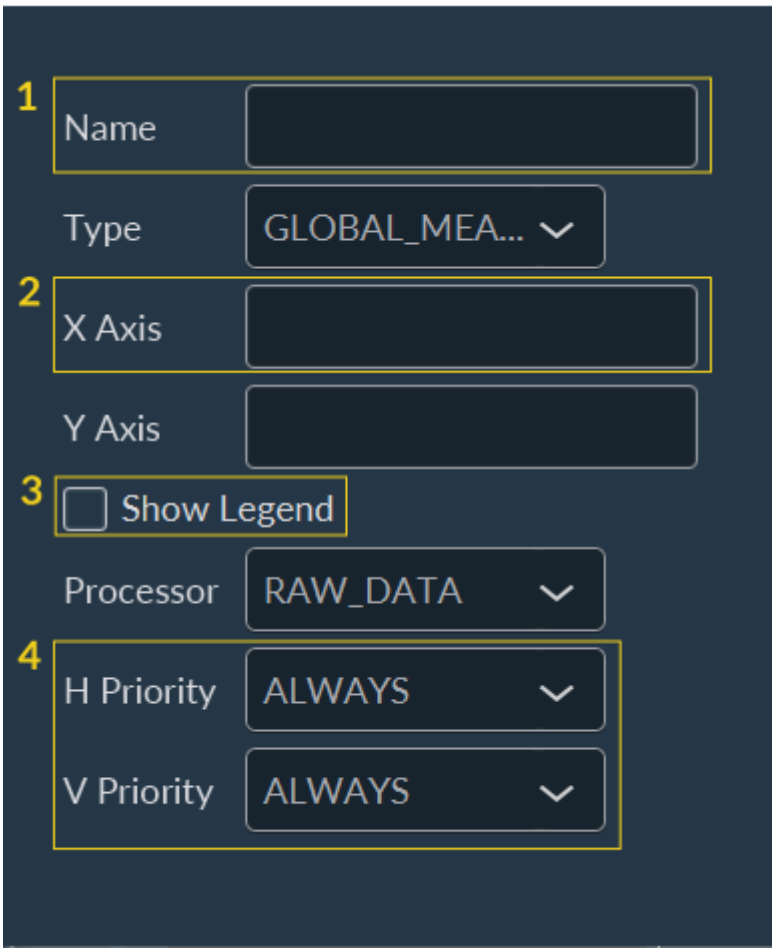
Customize the display of acquired data, such as tracking the evolution of fluorescence intensity over time.

SETTING UP THE VISUALIZATION WINDOW

1. **Select the Data Type**: Choose what metric to display during the acquisition:
    - *Histogram*: Monitors the evolution of the pixel intensity distribution.
    - *Global mean*: Monitors the average image intensity across an acquisition sequence.
    - *Global statistic*: Similar to *Global mean*, but includes additional statistical metrics (e.g., variance).
    - *Device values*: Periodically retrieves and plots telemetry from connected hardware devices.
2. **Assign to a Panel**: Drag and drop the selected data type into a view panel. Change this data type later using the panel's drop-down menu.
3. **Specify Source**: Define the data source (node) for the visualization.
4. **Customize Graph**: Adjust visual graph options, such as axis titles and curve labels.
5. **Add Graphs**: To display multiple graphs simultaneously, drag and drop additional data types into empty view panels.

GRAPH CUSTOMIZATION OPTIONS

1. Set a custom title for the graph.
2. Label the X and Y axes.
3. Toggle the visibility of curve legends.
4. Enable the **Always** option to ensure the graph remains visible throughout the experiment.



Note

Visualization customization is optional. It aids in data interpretation during acquisition and does not affect the saved acquisition data.

1.4.12 Saving Images and Metadata

The saving interface configures the destination path and data architecture for the automated sequence outputs.

Configure how the acquired images and metadata are handled:

1. Select the data storage destination:
    - **RAM**: *(Not recommended for prolonged acquisitions).* Data stored in RAM is volatile and is lost if the application closes or the system crashes. Use only for temporary test captures.
    - **Hard disk, SSD, or Server**: *(Recommended).* Writes data to a physical drive or network storage.
2. Specify the destination folder path.
3. Choose the image saving format:
    - **All in one file**: Combines the dimensional dataset (Z-stacks, Time points, Channels) into a single, multi-page image file.
    - **One TIFF per image**: Saves every captured frame as a discrete TIFF file (e.g., `Image_T01_Z03_C01.tiff`). Use this if post-processing tools require sequential files instead of multi-page containers.
4. *(Optional)* Check **.bigTiff** to support files exceeding the standard 4GB legacy TIFF limit (recommended for large volumetric datasets).

> **Tip**
>
> To configure standardized file naming conventions before initiating an automated acquisition, refer to the Display Settings guide.

1.4.13 Acquisition Designer

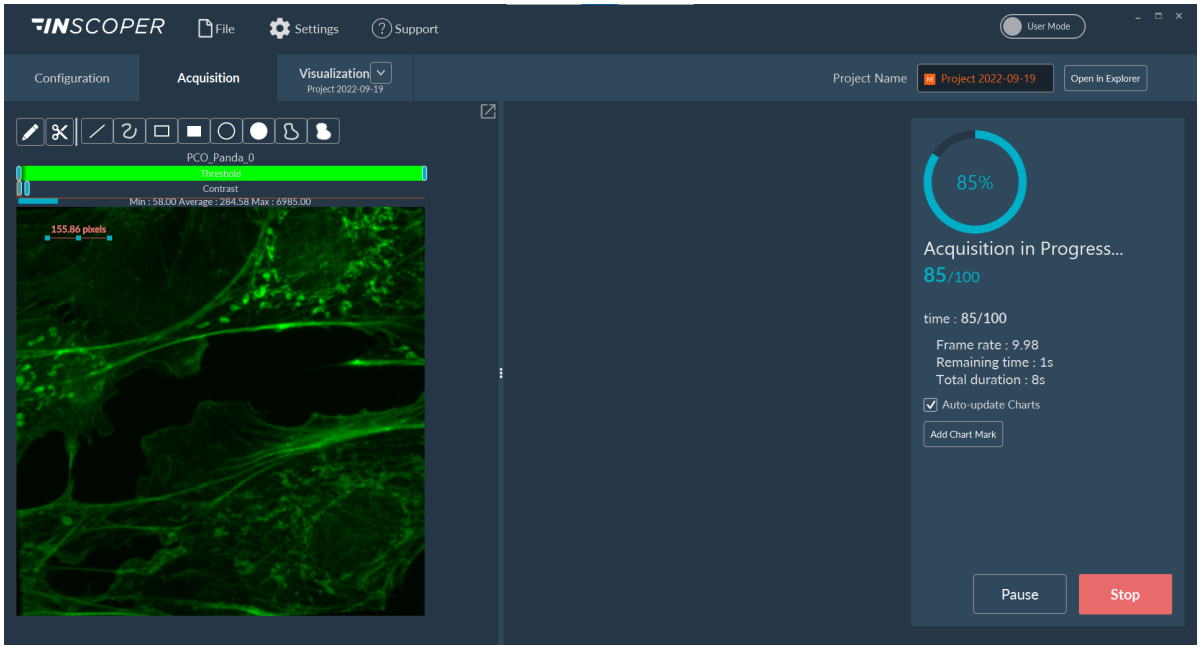
Documentation coming soon.

1.5 Visualization & Analysis

Inscoper I.S. provides tools for visualizing data during and after acquisition.

1.5.1 Visualization During Acquisition

The **Visualization** tab allows for real-time monitoring of the acquisition sequence.



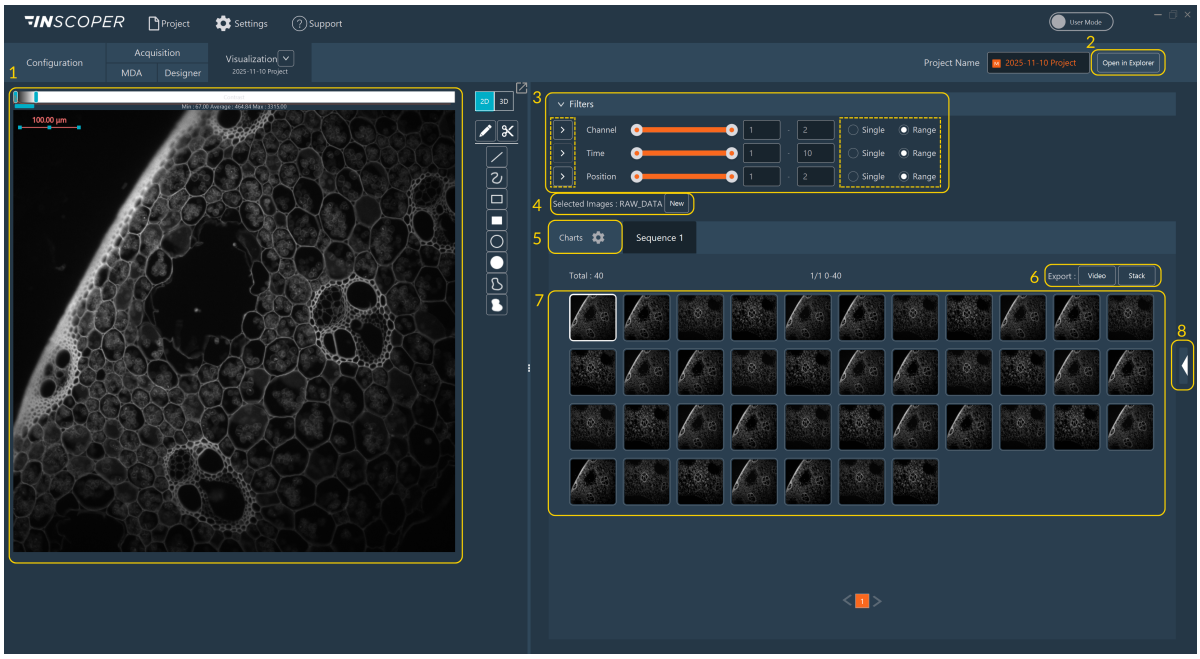
The interface is divided into three functional areas:

1. **Left Panel:** Displays the live, incoming images from the current acquisition sequence.
2. **Center Panel:** Renders corresponding graphical data (e.g., intensity evolution or statistical plots), depending on the configured Data Processor pipeline.
3. **Right Panel:** Provides acquisition progress metrics and system controls to pause or abort the active sequence.

Large Area Tiling

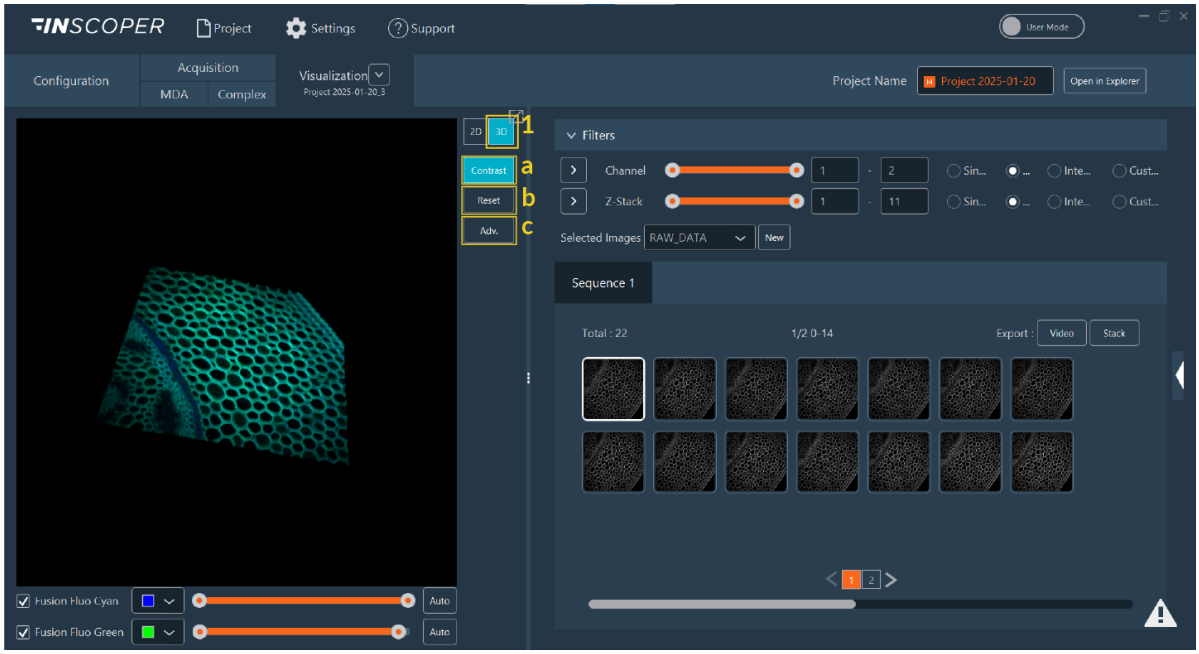
During extensive tiling acquisitions, the left panel dynamically updates to display the full macroscopic mosaic as individual tiles are acquired and aggregated into the global view.

1.5.2 Visualization After Acquisition

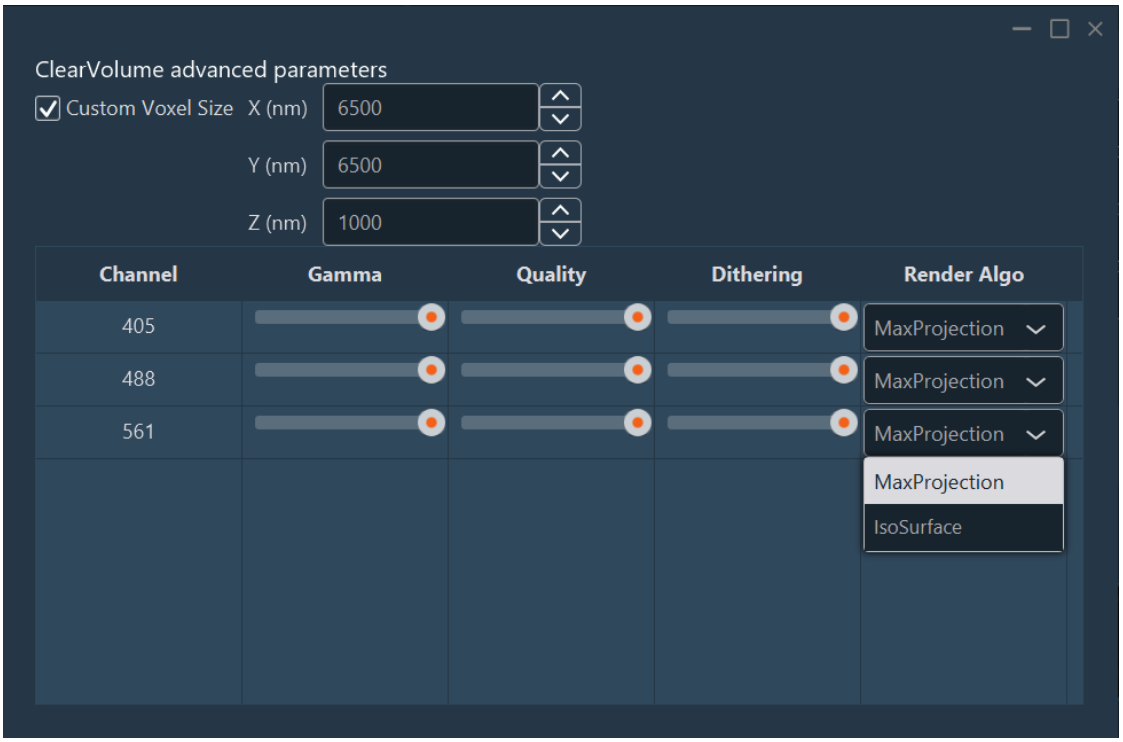


1. **Image Canvas**: View images in the central viewer.
2. **Open Directory**: Navigate directly to the local dataset storage path (applicable only if data is saved to disk).
3. **Filters**: Use dimension-based filters to search your dataset. Select individual images or full groups. Click the **Play** button adjacent to a dimension name to animate the sequence.
4. **Data Processing**: Select output images for post-acquisition processing.
5. **Graphics Mode**: Switch viewing modes to visualize quantitative data charts.
6. **Export Options**: Save the currently viewed sequence as an MP4 **Video** or a TIFF **Stack**.
7. **Thumbnail Gallery**: Browse miniature representations of all acquired images.
8. **Metadata Access**: Reveal the hardware and experimental parameters tagged to the images.

3D Viewer



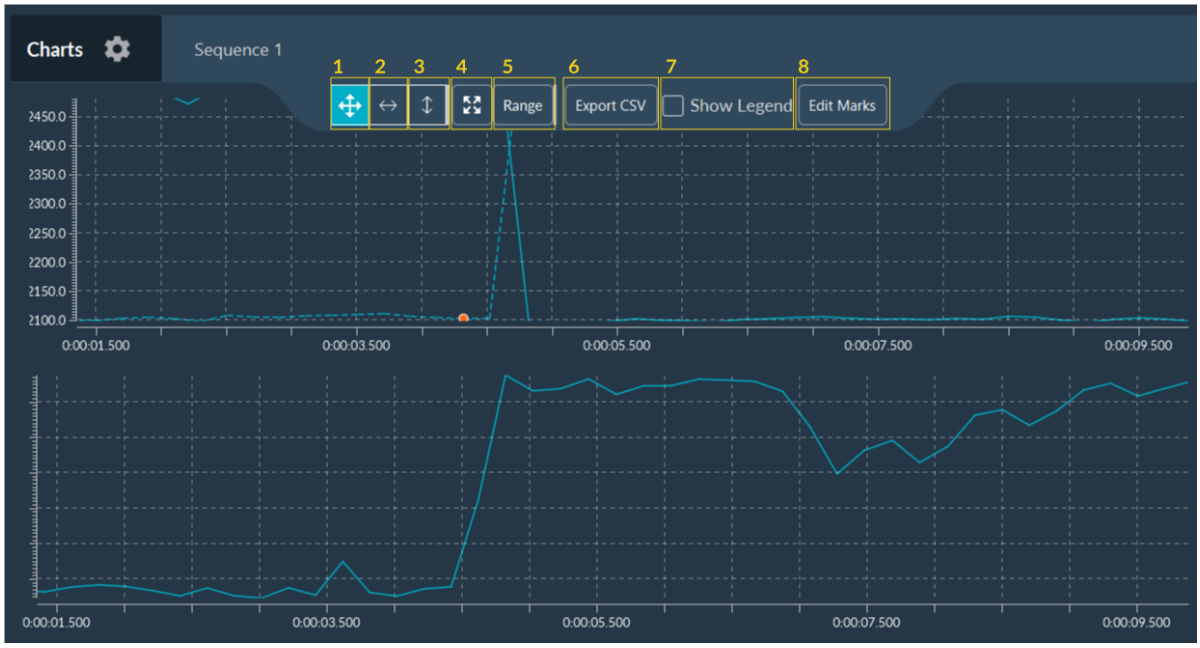
1. Click the **3D** button to launch the volumetric viewer.
2. 3D visualization canvas.
3. Adjust rendering options and parameters:
    1. **Contrast**: Display the LUT and adjust the contrast bar below the 3D view.
    2. **Reset**: Restore the default 3D camera view.
    3. **Advanced**: Access detailed rendering parameters per channel:
        - **Voxel Size** (nm).
        - **Gamma**: Apply a non-linear gamma curve to adjust contrast.
        - **Quality**: Adjust rendering quality for navigation.
        - **Dithering**: Reduce banding/aliasing artifacts.
        - **Render Algorithms**:
            - **MaxProjection**: Displays only the voxels with the maximum intensity encountered along the viewing ray.
            - **IsoSurface**: Renders a 3D surface connecting points of equal intensity value.



1.5.3 Interacting with Graphics

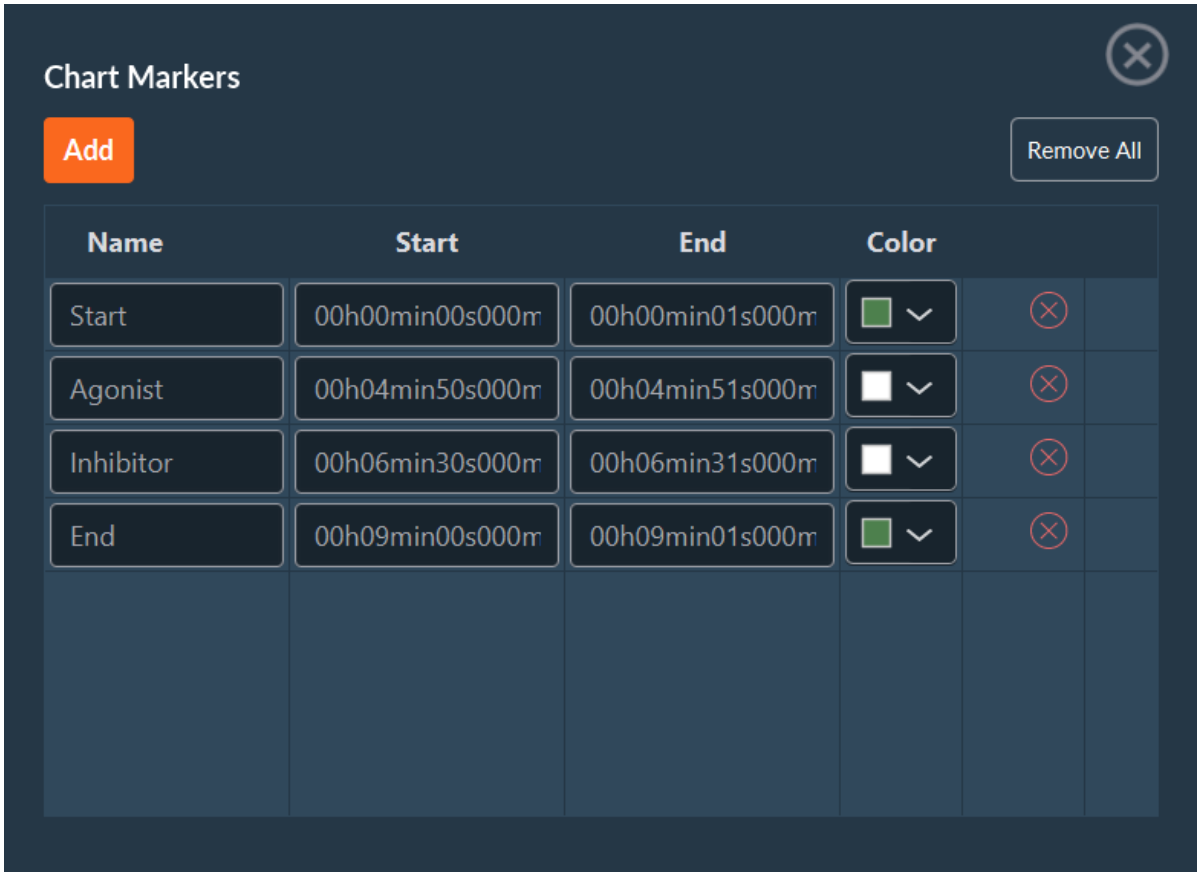
Interact with charts by adjusting their appearance, adding time markers, or exporting raw data. Hover over a graph to access its toolbars.

- **Pan**: Click and drag the mouse wheel (middle click) to pan across the graph.
- **Zoom**: Scroll the mouse wheel to zoom in and out. Click and drag to select a region to magnify.
- **Jump to Image**: Left-click a data point on the graph to view the corresponding image.
- **Reset View**: Right-click within the graph to restore the default axis scaling.



Graph Tools

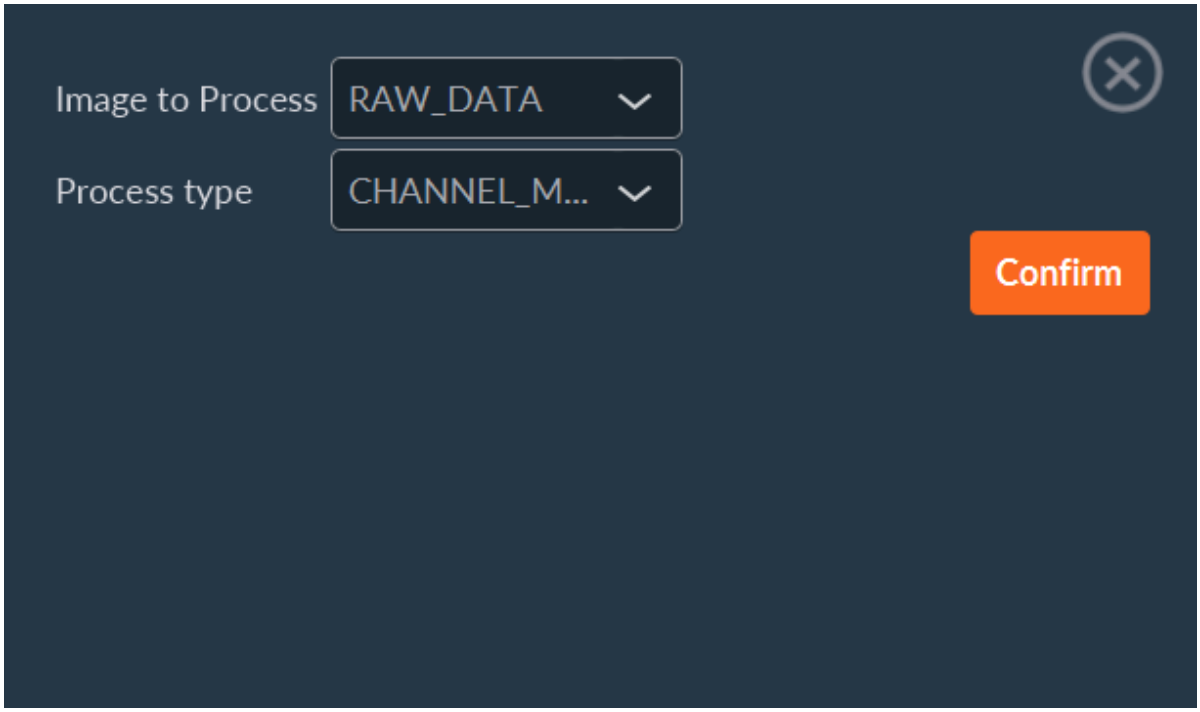
1. Enable the XY zoom mode
2. Enable the X zoom mode
3. Enable the Y zoom mode
4. Reset zoom or enable auto-ranging.
5. Manually override the axis ranges.
6. Export the raw graph data to a `.csv` file.
7. Toggle curve legends.
8. **Add Markers**: Annotate the timeline with event markers. Markers can be exported to a `.csv` file.



In this window you can add information about your experiment as markers. These events, which are fully customizable, can be associated with the acquisition itself (start, pause, end), external events (addition of an inhibitor, medium supplementation), or others. These markers can be saved in a .csv file and reused at any time.

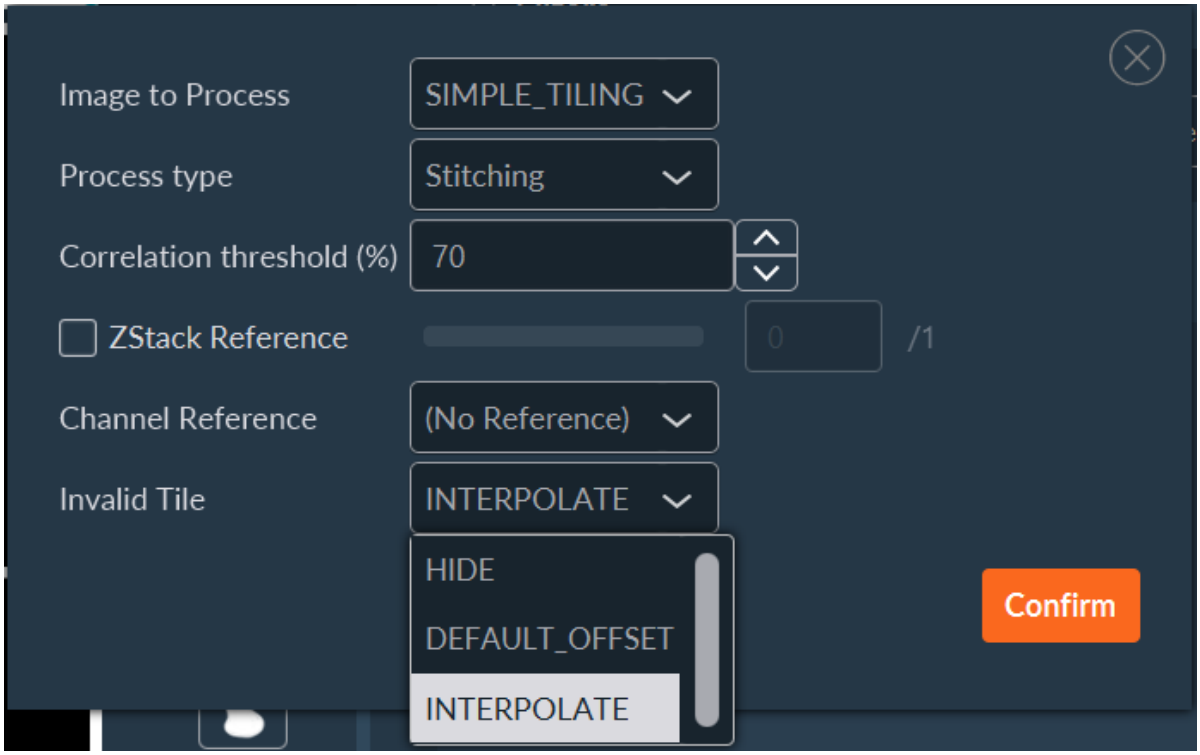
1.5.4 Image Processing

Apply post-acquisition data processing from the Visualization tab.



1. Select the target sequence from the image drop-down list.
2. Choose the desired data processor.
3. Click **Confirm** to execute.

Example: Tiling Processing



1. Select the source sequence.
2. Select the **Tiling** process type.
3. Set the cross-correlation threshold (percentage) for image stitching.
4. *(Optional)* Apply a Z-Stack reference, specifying which focal plane should drive the 2D stitching calculation.
5. *(Optional)* Select a specific reference channel to calculate the stitching coordinates.
6. Define the fallback behavior for uncorrelated (**invalid**) tiles:
    - **HIDE**: Discard the tile from the final image.
    - **DEFAULT_OFFSET**: Place the tile based solely on its recorded mechanical stage coordinates.
    - **INTERPOLATE**: Interpolate the tile's position based on the calculated offsets of surrounding valid tiles.
7. Click **Confirm**.

1.5.5 Data Export

Export sequences as MP4 videos or multi-dimensional Tiff stacks via the Export menu (Toolbar Item 6).

Video Export

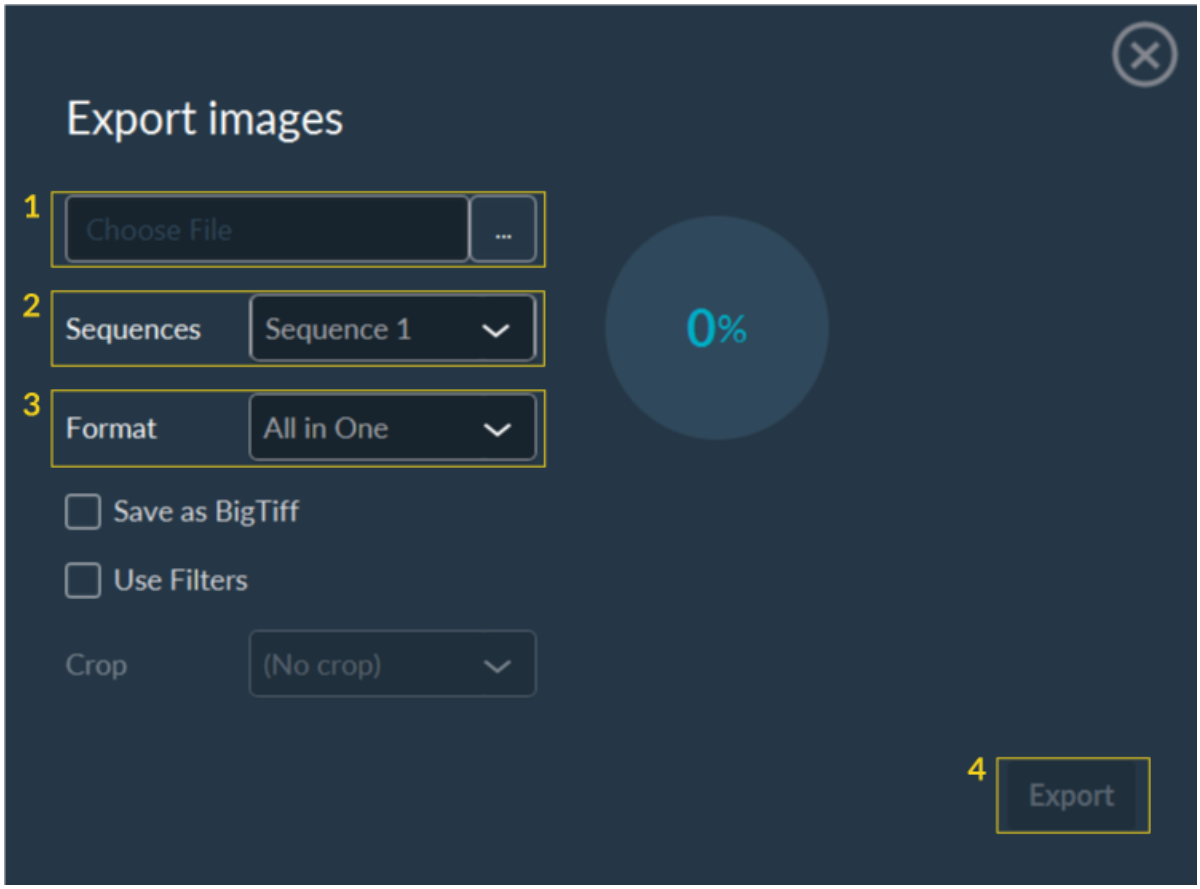
Use dimension filters to isolate the image sequence, then click **Video** from the Export dropdown.

To export a video:

1. Select the destination file path.
2. Verify the sequence to export.
3. Select the video format/codec.
4. Click **Export**.

Stack Export

Use dimension filters to isolate images, then click **Stack** from the Export dropdown to generate a multidimensional TIFF.

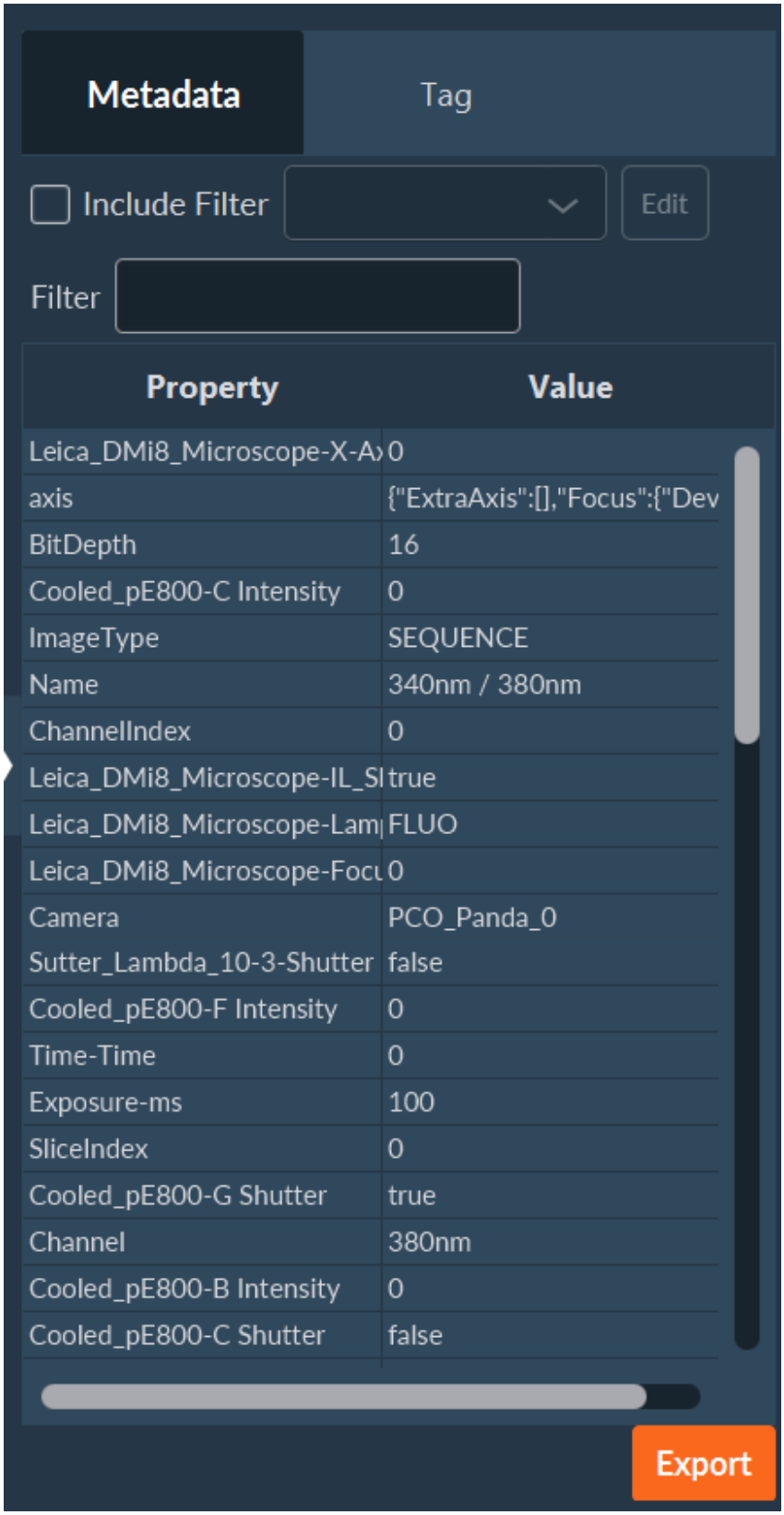


To export a stack:

1. Select the destination file path.
2. Verify the sequence to export.
3. Select the data format.
4. *(Optional)* Check **Save as BigTiff** if the expected file size exceeds 4GB.
5. *(Optional)* Check **Use Filters** to apply your visualization filters to the export.
6. Click **Export**.

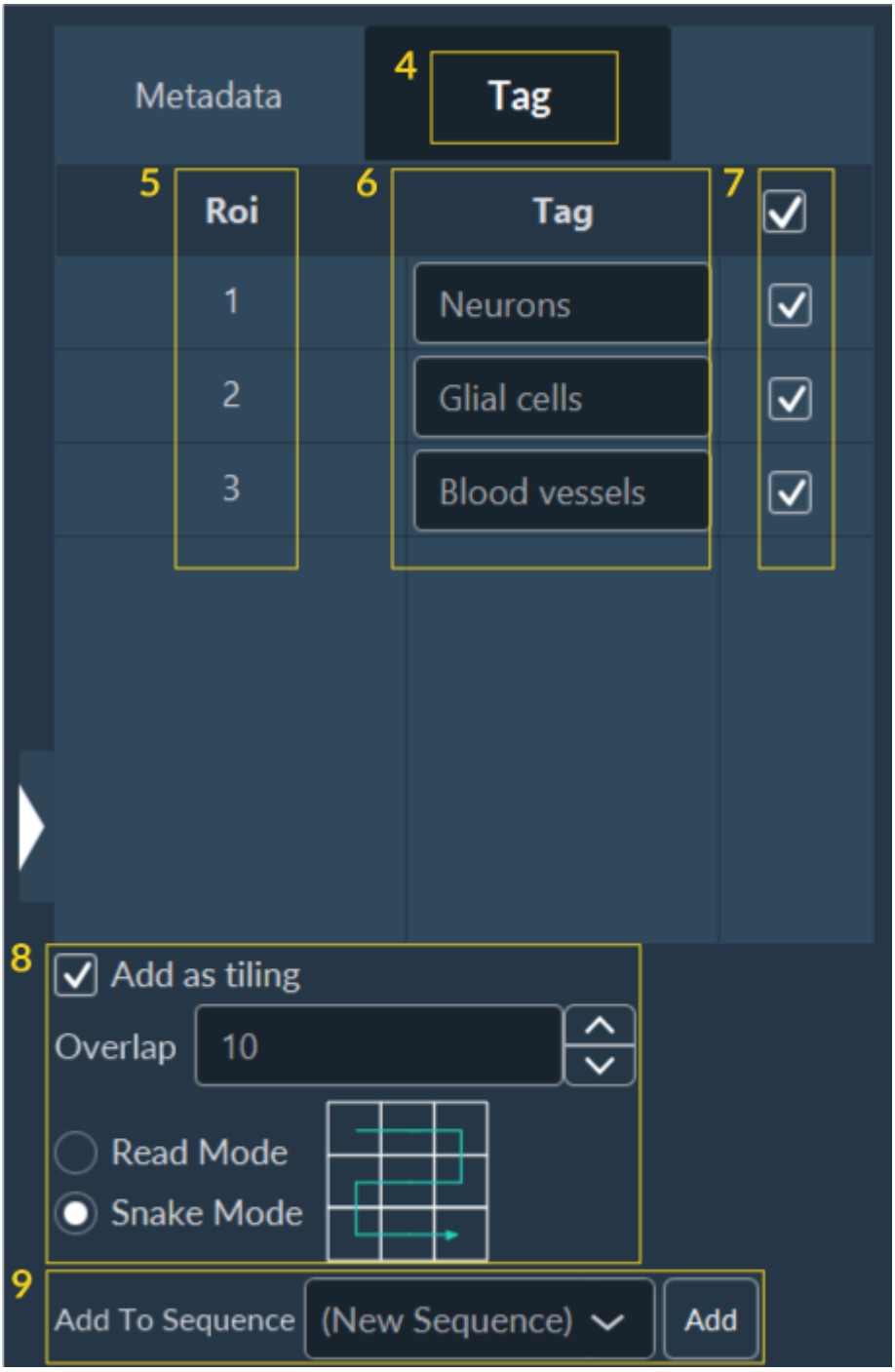
Metadata Access

On the right side of the window there is a white triangle. You can click on it to access all the metadata. In this tab, you can access all the metadata, including the camera, light source or microscope settings; a search bar and some filters are available to facilitate the search for some specific parameters. This list can also be exported, if necessary, by clicking on the Export button located in the lower right part of the screen. All metadata are bio-format compatible.



1.5.6 Semi-automated feedback microscopy feature

The software supports feedback microscopy workflows. For example, you can acquire a low-magnification overview scan, tag regions of interest, and export those coordinates into a new, high-magnification acquisition sequence.



To use this feature:

1. Acquire an overview image (e.g., a **Tiling** scan).
2. Use the ROI tools to draw regions around structures of interest.
3. In the **Visualization** tab, click the white triangle on the right edge to expand the side panel.
4. Navigate to the **Tag** sub-tab.
5. This lists all drawn ROIs. You can manage them directly from this list.
6. *(Optional)* Assign descriptive names/tags to each ROI.
7. Select the ROIs you wish to target.
8. If the new targets require their own sub-tilings, configure those spatial parameters here.
9. Choose to inject these new coordinates into a fresh sequence or append them to an existing one.

Smart Microscopy

Use this workflow for:

1. A low-magnification pre-scan followed by targeted high-resolution imaging.
2. A brightfield pre-scan, applying fluorescence imaging to identified structures to reduce global phototoxicity.

1.6 Add-ons

1.6.1 Custom Python Scripting

Introduction to Python Scripting

The **Python Scripting** module bridges the gap between automated image acquisition and custom computer vision algorithms. By executing Python scripts after each image capture event, it enables smart microscopy: dynamic, adaptive workflows that use computational analysis to control microscope hardware.

This feature is constructed upon two core components:

1. THE GRAPHICAL INTERFACE MODULE

> **Licensing Requirement**
>
> The **Custom Scripting** software license option is required to use this module.

Located natively within the Inscoper I.S. GUI, the module serves as the bridge between your custom scripts and the acquisition engine. It is found within the **multi-dimensional acquisition (MDA)** tab, immediately below the standard sequence settings.

- **Automated Settings UI**: Upon importing a Python script, the module parses the code to identify all functions (potential entry points). Parameters defined in your Python functions are mapped to graphical controls (e.g., `bool` arguments generate toggle switches, `str` become text fields, enumerations become dropdown lists).
    - The **Image Provider** function dictates which acquired images are passed to the Python processing function at any given time point.
    - The **Processing Function** executes the analysis (e.g., computer vision, intensity measurements) and returns the updated values for microscope hardware parameters.
- **Sequence Execution**: The system engine triggers these functions as callbacks following each image capture. The *Image Provider* is invoked first to assess if analysis is needed; if confirmed, the *Processing Function* runs, adjusting designated hardware parameters, or pushing figures to the interface.

2. THE DEVELOPMENT TOOLKIT (`inscoper-scripts`)

Provided as a local package, this toolkit serves as the foundation for your custom script development. It is organized into three categories:

- **Application Collection**: A suite of ready-to-use smart microscopy master applications, including modules for:
    - Mechanical tracking
    - Single-cell population screening/tracking
    - Autofocus
    - FRAP workflows
- **Structural Templates**: Simplified application examples that serve as starting points for generating custom scripts.
- **Utility Sub-modules**: A suite of helper libraries containing predefined image providers, ROI objects, and mathematical functions for tracking, measurement, segmentation, and filtering. These utilities simplify interacting with the core software environment, allowing you to display figures in the GUI, manipulate metadata dictionaries, and generate analytical plots.

Calling Scripts in Inscoper I.S.

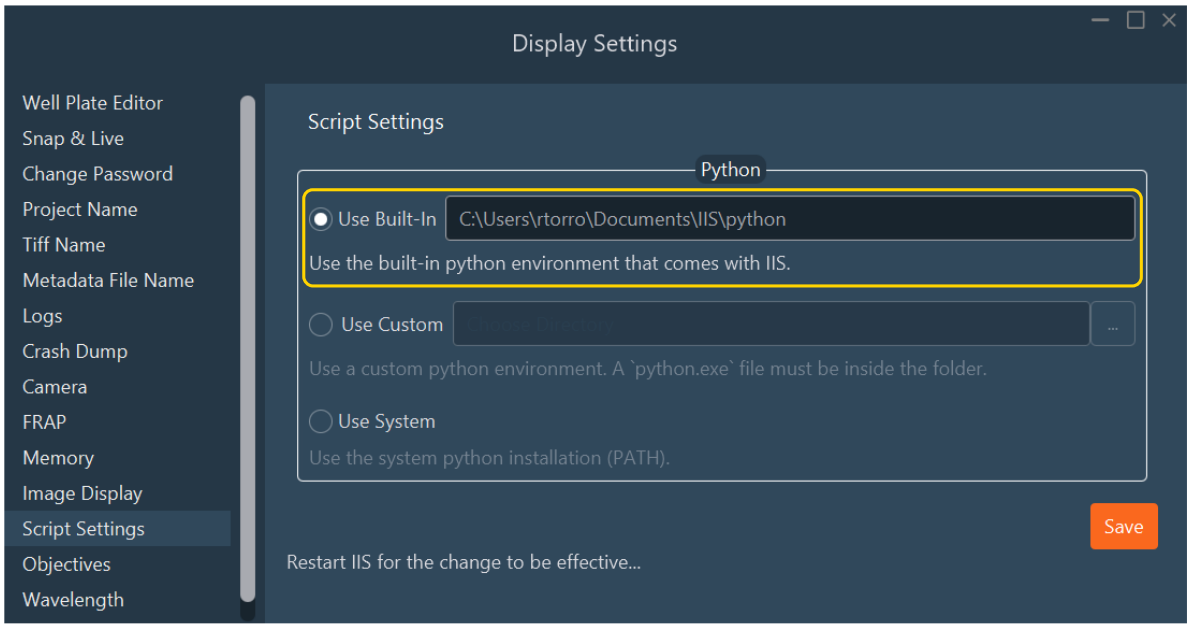
The Python Scripting module enables the execution of custom Python algorithms within an MDA sequence.

To ensure compatibility, scripts must adhere to the templates provided in the `inscoper-scripts` local package. You may use the included sample scripts for testing.

PYTHON ENVIRONMENT SETUP

Before executing any script, you must explicitly specify which Python environment the software will utilize for its calculations.

1. Navigate to **Settings** > **Display Settings**.
2. Select **Script Settings** from the left-hand navigation pane.
3. Choose your preferred execution environment:
    - **Use Built-In (recommended)**: Utilizes the stable, integrated Python environment bundled securely with the software installation. This environment comes pre-loaded with standard (non-GPU) data science and essential image analysis packages.
    - **Use Custom**: Point the software to a specific, user-managed local Python executable (`python.exe`). Note: This advanced route explicitly requires the **jep** library (see warning below).
    - **Use System**: Automatically detects and utilizes the global Python installation currently defined in your operating system's PATH variables.
4. **Restart Inscoper I.S.** to fully instantiate the environment changes.



Jep Package Required for Custom Environments

If you elect to use a custom environment (such as a specific Conda environment), the **jep** (Java Embedded Python) package is strictly required to bridge the software architectures.

Install it by activating your target environment and running:

```
pip install jep
```

Expanding the Built-In Environment

If your custom script requires specialized statistical or biological libraries not included in the standard distribution:

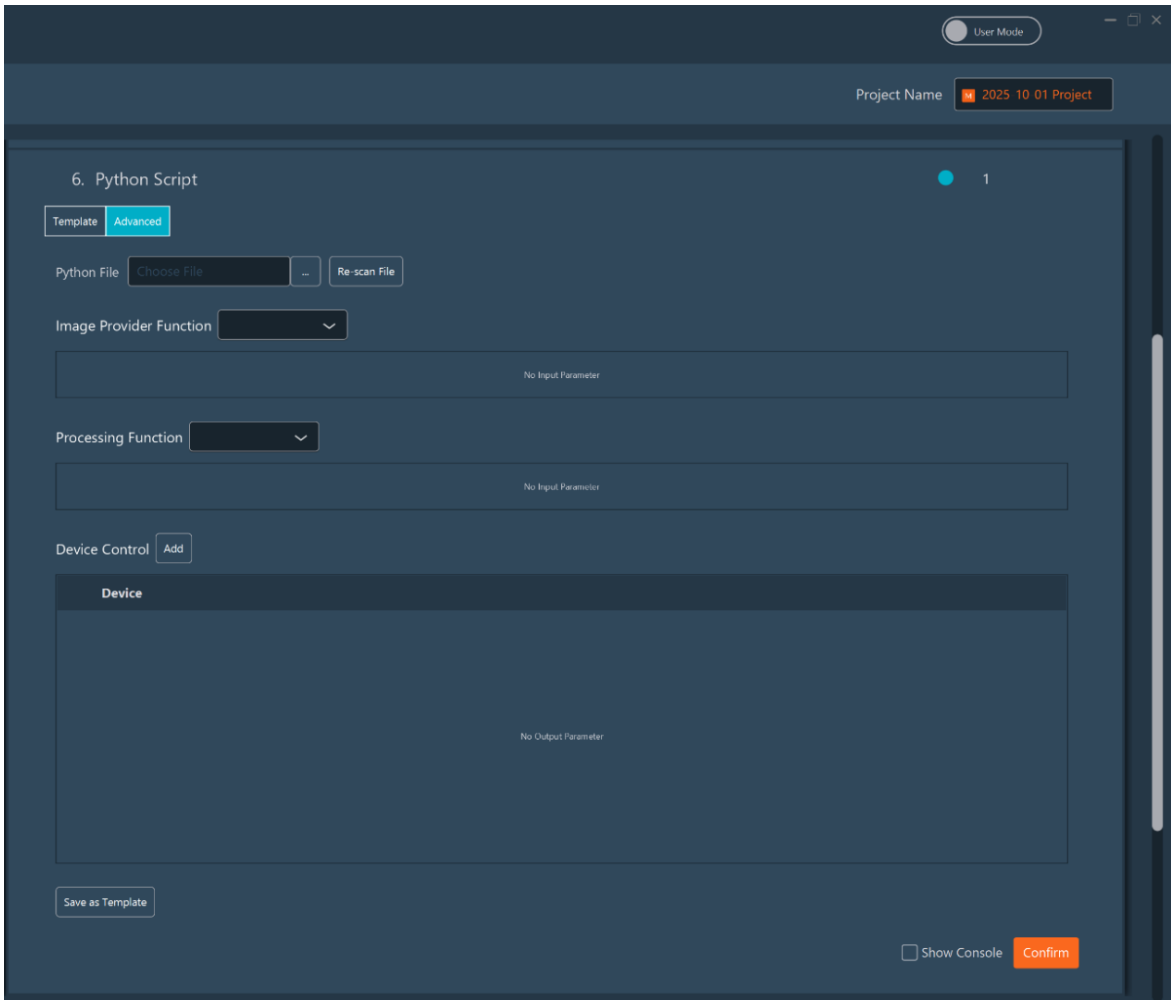
1. Open a command line terminal within the **I.I.S. installation directory**.
2. Navigate to the nested `python` subdirectory.
3. Execute the standard pip installation command:

```
./python.exe -m pip install <package_name>
```

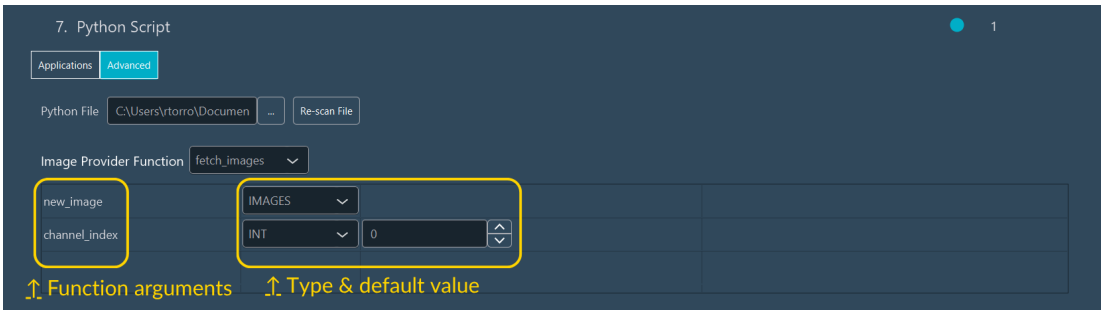
IMPORTING AND MAPPING A SCRIPT APPLICATION

Once your Python interpreter is set, you can load your script into the MDA sequence definition.

1. Navigate to the central **Acquisition** tab and select the **MDA** (Multi-Dimensional Acquisition) sub-tab.
2. Expand the **Python Script** module panel and toggle the **Advanced** switch to enable this mode.
3. Click the **"..."** button adjacent to **Re-scan File**, browse to the `inscoper-scripts` local package, and select your specific `.py` script.

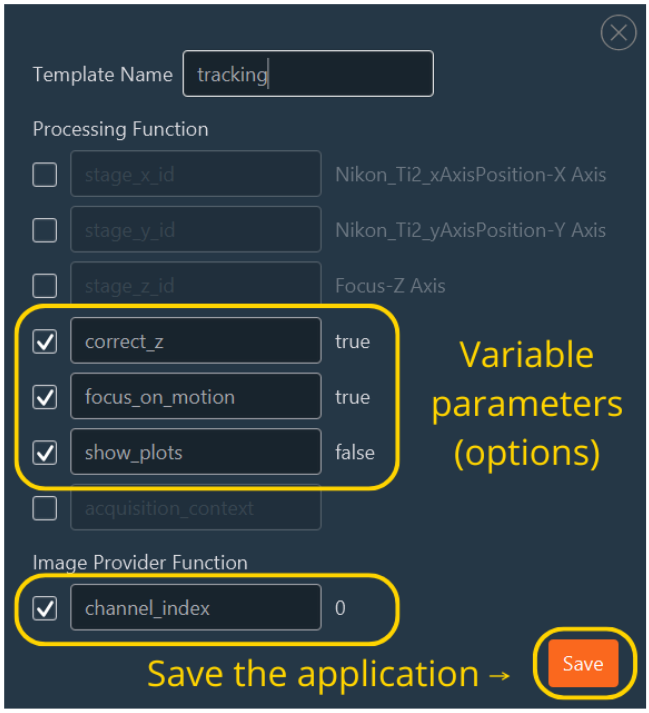


1. **Map Python functions to I.I.S. parameters.** Define which function handles image fetching (image provider) and which handles heavy-computations (processing), then map the functions' arguments to the proper type and value.
    1. **Step A (image provider)**: Select the script function responsible for fetching images (e.g., `fetch_images`).
        - **Map Arguments**: Map the script variable `new_image` to the `IMAGES` data type. This allows the software to pass the most recently acquired image to your script.
        - Map `channel_index` to `INT` and assign a default value (e.g., `0`). These channel indices correspond to their order within the Multi-channels dimension.



    1. **Step B (processing function)**: Select the function responsible for data analysis (e.g., `correlation_tracking`).
        - **Map Arguments**: Map the `images` variable to the `IMAGES` type. This variable receives the images curated by the Image Provider function.
        - Map hardware identifiers (e.g., `stage_x_id`, `stage_y_id`, `stage_z_id`) to the `SUBDEVICEID` type and set default values matching your hardware.
        - Map logical switches (e.g., `correct_z`, `focus_on_motion`, `show_plots`) to `BOOL` and define initial states.
        - Map `acquisition_context` to `IIS_CONTEXT`. This provides access to internal software APIs, such as rendering `matplotlib` figures in the UI (see how to use the IIS Context Engine).
    2. **Step C (device control)**: Declare any hardware sub-devices that will be updated by your processing function's outputs (it should match symmetrically with the return values of the processing function, e.g. if the processing function returns a tuple of new 'x', 'y', and 'z' values, then you should declare the X, Y, and Z sub-devices here). If your function only logs data without modifying hardware, leave this blank.

2. **Save as a Reusable Template**
    1. Once mapping is complete, click **Save as Application Template**.
    2. In the dialog box, select which parameters should remain editable by the user.
    3. Unselected parameters will be hardcoded to their default values and hidden from the UI.



Configuring the UI for Script Visualization

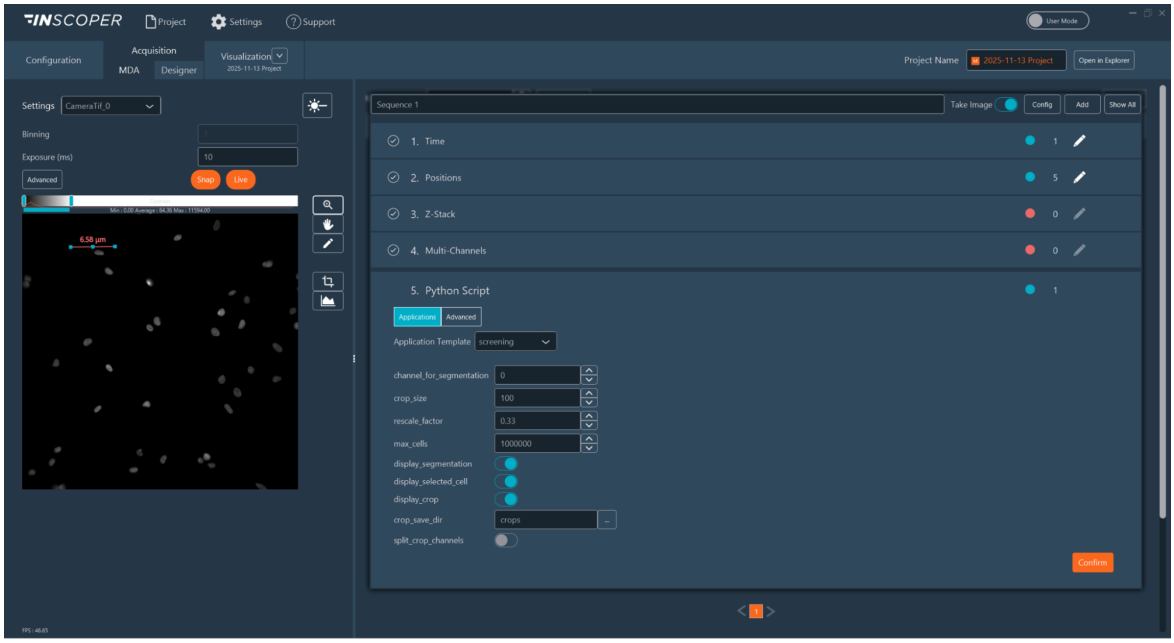
If your Python script is designed to output interactive visual graphs (via `matplotlib`) or text logs to the console, you must configure a data processor to receive this data.

1. Enable **Expert Mode** using the top-right interface toggle.
2. Within the **Data Processing and Charts** operational section, click **Switch to Data Processor**.
3. **Enable Console Logging**: Drag the **Feedback Console** module from the library into the right-hand panel workspace. This active console captures and displays all standard Python `print()` logging outputs.
4. **Enable Graphical Output**: If your script mathematically computes and pushes a figure using `display_figure(..., processor_name='name')`:
    - Drag a **Feedback Chart** module into the right-hand panel workspace.
    - **Crucial**: You must manually rename this chart element within the UI to exactly match the target `processor_name` string defined within your underlying Python script.
5. Save this complete charting layout with a descriptive, logical name and click **Add**.
6. Click **Switch to Dimensions** to return to the standard sequence configuration window.

EXECUTING THE SCRIPTED EXPERIMENT

Executing a templated script is designed for operational simplicity.

1. Navigate to the **Application** tab.
2. Select your saved template from the dropdown list.
3. Adjust any exposed control parameters.
4. Verify that the **Data Processing and Charts** menu selection matches the layout required by the script.
5. Click **Start Acquisition** to initiate the sequence.



Writing Custom Scripts

This module facilitates the execution of Python script functions during acquisition. For the scripts to work, they must adhere to the templates provided in the `inscoper-scripts` repository.

IMAGE PROVIDER FUNCTION ARCHITECTURE

The **image provider** function acts as your script's entry point. The acquisition engine invokes this function **after every image capture event**. Its responsibilities are:

1. **Execution Gatekeeping**: Evaluating whether the processing function should run (e.g., waiting until a Z-stack is completed).
2. **Image Selection**: Deciding which images to fetch from memory and pass to the processing function (e.g., passing the full Z-stack array versus isolating the final 2D frame).

Signature

Arguments

- `new_image` (`images`): The function expects a primary argument defined with type `images`. It receives the **last image acquired** by the hardware. This object contains the metadata required to identify the microscope's multidimensional state (Time, Position, Channel, and Z-slice indices).
- **Optional Parameter Arguments**: You can define additional parameters (such as `channel_index`, `focus_option`) which the user will configure within the GUI.

Return Value

The Image Provider function is structurally required to return a `List[Dict[str, int]]`.

- **Empty List `[]` or `None`**: Returning an empty array signals the engine to **SKIP** the processing function for the current iteration.
- **Populated List of Dictionaries**: Instructs the system to **EXECUTE** the processing function. Each dictionary specifies an image to fetch from memory.
  - **Empty Dictionary `{}`**: Fetches the **current image** (indices are inferred from the `new_image` object).
  - **Specific Indices** (e.g., `{'focusIndex': 3}`): Fetches an image from a specific coordinate (e.g., Z-slice 3). Any dimension index not explicitly declared (e.g., `timeIndex`, `positionIndex`) is copied from the current `new_image`.

Architectural Dimension	Dimension Key (`index`)	Dimension Size Attribute (`sizeDim`)
Time (t)	`timeIndex`	`timeSize`
Position (XY)	`positionIndex`	`positionSize`
Channel (λ)	`channelIndex`	`channelSize`
Focus (Z)	`focusIndex`	`focusSize`
Lifetime (FLIM)	`flimIndex`	`flimSize`

Code Example: Fetching an Entire Z-stack

This example checks if the current image is the absolute final slice of a Z-stack. If so, it collects all prior slices so that the Imaging Software can dispatch this list of images to the processing function; if not the final slice, it does nothing and awaits the next capture. Note that the `positionIndex` and `timeIndex` are inferred from the `new_image` object.

```
from inscoper_scripts.utils import extract_metadata
import numpy as np
```

```python
from typing import List, Dict

def fetch_zstack(new_image: "images", channel_index: int = 0) -> List[Dict[str, int]]:
    """
    Collect the entire volumetric Z-stack for the current spatial position at the logically defined
target channel.
    """
    # 1. Analytically extract the metadata dictionary from the current image object
    meta = extract_metadata(new_image)

    current_z = meta.get('focusIndex', 0)
    total_z = meta.get('focusSize', 1)

    # 2. Evaluate physical conditions: Only authorize execution at the final mechanical Z slice
    if current_z != (total_z - 1):
        return [] # Return empty array to skip computationally expensive processing

    # 3. Construct chronological selection: All volumetric slices (0 to total_z-1)
    # We deliberately only specify 'focusIndex' and 'channelIndex'.
    # 'timeIndex' and 'positionIndex' are reliably inferred from the current image state.
    image_selection = []
    for z in range(total_z):
        image_selection.append({
            'focusIndex': z,
            'channelIndex': channel_index
        })

    return image_selection
```

Code Example: Fetching the Singular Current Image

This represents the simplest case. Returning a list containing a single empty parameter dictionary instructs the engine to send solely the current image to the processing function.

```python
def fetch_current_image(new_image: "images") -> List[Dict[str, int]]:
    return [{}]
```

Code Example: Fetching the Last Two Distinct Timepoints

This specific pattern retrieves the image from both the previous (`t-1`) and the current (`t`) timepoints. This is useful for tracking algorithms.

```python
def fetch_last_two_timepoints(new_image: "images") -> List[Dict[str, int]]:
    meta = extract_metadata(new_image)
    current_t = meta.get('timeIndex', 0)

    # Logical Gate: We require a minimum of 2 discrete timepoints (Index 0 and Index 1) to proceed
    if current_t < 1:
        return []

    return [
        {'timeIndex': current_t - 1}, # Request previous chronological timepoint
        {'timeIndex': current_t}      # Request current, active timepoint
    ]
```

PROCESSING FUNCTION ARCHITECTURE

Within the data processing function, the primary argument must be designated as `images`. This variable receives the list of image arrays returned by the image provider, serving as the dataset for your algorithms.

SUPPORTED ARGUMENT TYPES AND UI AUTOMATION

Standard Python Type	Automatically Generated Graphical Interface	Functional Description
`str`		A standard string parameter, rendered identically as a text input field.
`int`		An integer value, rendered as a numeric spin box with a fixed increment constraint of 1.
`float`		A floating-point value, rendered as a precise numeric spin box with a standardized increment of 0.001.
`bool`		A binary boolean value, rendered as a visual toggle switch.
`Path` or `str` with `/` or `\\`		A path directly to a file directory, rendered as a system directory chooser widget.
`Literal["val1", "val2", "val3"]`		A value constrained by an enumeration, rendered as an exclusive dropdown list.
`List[Literal["val1", "val2", "val3"]]`		A list of values constrained by an enumeration, rendered as a multi-selection dropdown list.
`images`		Represents one or more image arrays provided by the Imaging Software (the last acquired image for the image provider function, or a curated list of images for the processing function).
`SubDeviceId`		A sub-device name (used as a key to read its value in the metadata), rendered as a dropdown list of all available sub-devices.
`channel`		A channel name, rendered as a dropdown list of all available channels.
`IIS_CONTEXT`		A context object, allowing the script to interact with the software environment (variable transfer, method execution).

Initial default logic values can be initialized directly inside the function signature header, adhering to standard Python typing conventions: e.g., `var1: float = 0.5` or `List[Literal["val1", "val2", "val3"]] = ["val2"]`.

UTILIZING THE IMAGING SOFTWARE CONTEXT ENGINE (`IIS_CONTEXT`)

The `IIS_CONTEXT` exposes methods allowing your script to interact with the software environment.

`getCameras()`

Retrieves a list of active cameras.

- **Return Signature**: `List[str]` - An array list containing valid camera identifier strings.

Functional Example:

```
cameras = IIS_CONTEXT.getCameras()
# Programmatic return evaluation: ["Camera1", "Camera2"]
```

`getProjectPath()`

Retrieves the path of the currently active project directory.

- **Return Signature**: `str` - The string representing the active project path.

Functional Example:

```
path = IIS_CONTEXT.getProjectPath()
# Programmatic return evaluation: "D:/Data/Project_001"
```

`displayImage(id, img)`

Sends an image array to render directly into a designated data processor within the visualization dashboard.

- **Parameters**:
  - `id` (`str`): An identifier matching the target data processor.
  - `img` (`NDArray`): The raw image data payload to view.

`setVariable(id, value)`

Caches a variable within the application's shared memory architecture. This function is useful to pass global variables during the acquisition loop.

- **Parameters**:
  - `id` (`str`): The string key utilized to identify the cached variable.
  - `value` (`Object`): The data value intended to be stored.

Functional Example:

```
IIS_CONTEXT.setVariable("tracking_counter", 1)
```

`getVariable(id)`

Retrieves a variable previously cached utilizing the `setVariable` method.

- **Parameters**:
  - `id` (`str`): The string key corresponding to the desired variable.
- **Return Signature**: `object` - The currently cached value.

Functional Example:

```
count = IIS_CONTEXT.getVariable("tracking_counter")
```

`fireOnDemand(cameraName, roi, channelName, powerMap, repetition)`

Triggers a FRAP photo-activation sequence along the defined region of interest (ROI) using the specified channel and power parameters.

- **Parameters**:
  - `cameraName` (`str`): The name of the camera to be used for the photo-activation sequence (this string must exactly match one of the returned values from the `getCameras()` execution).
  - `roi` (`Object`): The geometric region of interest. It functionally requires an ROI programming instance (e.g., `RectangleROI`) or an extended array list containing such precise instances imported from the system's `inscoper_scripts.utils.rois` core module. If an array list is structurally provided, the system sequentially, mechanically photoactivates each ROI.
  - `channelName` (`str`): The definitive channel configuration nomenclature to utilize.
  - `powerMap` (`Object`): The physical hardware Illumination/Power settings formatted algorithmically as a Python dictionary structure (e.g., `{"MyLaserSubDevice": 100}`).
  - `repetition` (`Object`): The numerical repetition parameters (e.g., configuring `10` forces exactly 10 consecutive hardware cycles).

Functional Example:

```
from inscoper_scripts.utils.rois import RectangleROI

# Objectively define structural parameters
cam = "Camera1"
my_roi = RectangleROI(0, 0, 512, 512, fill=False)
channel = "GFP_Activation"
power = {"488nm_Laser": 50.0}
reps = 1

# Issue command to mechanically trigger sequence acquisition
IIS_CONTEXT.fireOnDemand(cam, my_roi, channel, power, reps)
```

Python Script Library Reference

This document provides a comprehensive technical reference for the pre-built Python scripts available within the `inscoper_scripts` foundational library. It explicitly details the required entry points, utilized data processors, primary data outputs, and generated metadata for each core application.

CORE SMART MICROSCOPY APPLICATIONS

Automated Screening Scripts

Single-cell Screening (AI-Driven)

- **Script Name**: `single_cell_screening.py`
- **Architectural Entry Points**:
  - Image Provider: `fetch_images`
  - Processing Logic: `detect_single_cells`
- **Required Data Processors**: `segmentation`, `crop`
- **Functional Description**: Automatically screens and segments individual cells within an acquisition sequence. It utilizes a deep learning segmentation algorithm (Cellpose) to detect isolated single cells from a designated fluorescence channel, generates targeted single-cell crops, and exports them directly to disk. The UI dynamically displays single-cell geometric measurements.
- **Disk Output**: `crops/*.json`, `crops/*.tif`
- **Injected Metadata**: `NBR_OF_CELLS`, `MEAN_CELL_AREA`, `MEDIAN_CELL_AREA`, `ROI_POINTS`

Single-cell Screening (Thresholding)

- **Script Name**: `single_cell_screening_no_ai.py`
- **Architectural Entry Points**:
  - Image Provider: `fetch_images`
  - Processing Logic: `detect_single_cells`, `run_detect_single_cells_async`
- **Required Data Processors**: `segmentation`, `crop`
- **Functional Description**: Operationally equivalent to `single_cell_screening.py`, but utilizes classical image processing techniques (intensity thresholding, watershed transforms) rather than deep learning for cellular segmentation. This approach is faster but potentially less robust in confluent or complex scenarios.
- **Disk Output**: `crops/*.json`, `crops/*.tif`
- **Injected Metadata**: `NBR_OF_CELLS`, `MEAN_CELL_AREA`, `MEDIAN_CELL_AREA`, `ROI_POINTS`

Cellular Tracking Workflows

Single-cell Population Tracking

- **Script Name**: `cell_population_tracking.py`
- **Architectural Entry Points**:
  - Image Provider: `fetch_images`
  - Processing Logic: `cell_population_tracking`
- **Required Data Processors**: `segmentation`, `tracks`, `timeseries`
- **Functional Description**: Dynamically tracks a population of cells over time throughout a 2D or 3D sequence. It performs frame-by-frame segmentation (using Cellpose or thresholding), links identified cells between frames to construct kinetic tracks, and computes running ensemble metrics (mean/median velocities or intensities) for the population. It features heuristics for tracking temporarily lost cells and compensating for global stage drift.
- **Disk Output**: `tracks.csv` (if explicitly enabled in context logic)
- **Injected Metadata**: `NBR_OF_CELLS`

Mechanical Hardware Tracking

Correlative Hardware Tracking of a Target Object

- **Script Name**: `correlation_tracking_withZ.py`
- **Architectural Entry Points**:
    - Image Provider: `fetch_images`
    - Processing Logic: `correlation_tracking`
- **Required Data Processors**: `motion_difference`, `motion_mask`
- **Functional Description**: Tracks a targeted sample object in 3D space using phase correlation. It computes the XYZ optical shift between the current and preceding timepoints, generating real-time stage position updates to maintain the sample centered within the field of view. It supports corrective Z-axis positioning via determination of the optimal focus plane and can deploy estimation for sub-pixel shift calculations.
- **Disk Output**: `tracks.csv` (if a valid data export path is actively defined)
- **Injected Metadata**: Calculated vector shift values are written to runtime logs/console.

Photomanipulation Integration

Active FRAP / Photo-activation

- **Script Name**: `roi_manipulation.py`
- **Architectural Entry Points**:
    - Image Provider: `fetch_images`
    - Processing Logic: `compute_roi`
- **Required Data Processors**: `image`, `roi`
- **Functional Description**: Demonstrates how to programmatically define, generate, and manipulate spatial Regions of Interest (ROIs) on an incoming image array. It creates geometric shapes (Rectangles, Circles, Lines) and can trigger a "fire-on-demand" targeted laser action on the ROI geometry, useful for developing automated FRAP or photo-activation workflows.
- **Disk Output**: None
- **Injected Metadata**: None

Focus Management

Algorithmic Autofocus

- **Script Name**: `autofocus.py`
- **Architectural Entry Points**:
    - Image Provider: `fetch_images`
    - Processing Logic: `auto_focus`
- **Required Data Processors**: `focus` (implicit)
- **Functional Description**: Calculates the optimal Z-focus parameter from a streaming Z-stack acquisition. It analyzes relative image sharpness (contrast gradients) across the Z-slices and returns the Z-position of the sharpest focal plane. This allows for closed-loop software autofocus during time-lapse experiments without relying on hardware focus devices.
- **Disk Output**: None
- **Injected Metadata**: The calculated optimal absolute Z position is exported to standard logs.

Image Optimization

Dynamic Background Sampling

- **Script Name**: `background_model.py`
- **Architectural Entry Points**:
    - Image Provider: `fetch_images`
    - Processing Logic: `correct_background`
- **Required Data Processors**: `background`, `normalized`
- **Functional Description**: Performs background correction. At the initial timepoint t_0, it constructs a background illumination model (e.g., computing the median intensity projection across multiple fields of view). For subsequent timepoints, it applies this model to flat-field correct the incoming raw images. It can be optionally configured to mask out distinct cells/objects before computing the background model.
- **Disk Output**: `background.tif` (The generated master background model)
- **Injected Metadata**: None

STRUCTURAL SCRIPTS & UTILITIES

Developmental Templates

Comprehensive Demo Workflow

- **Script Name**: `demo.py`
- **Architectural Entry Points**:
    - Image Provider: `fetch_images`
    - Processing Logic: `find_most_eccentric_cell`
- **Required Data Processors**: `segmentation`, `crop`
- **Functional Description**: A demonstration script showcasing an integrated workflow: initial background subtraction, subsequent Cellpose neural network segmentation, geometric feature extraction (identifying the most eccentric cell), and stage movement targeting that cell. It serves as an example of combining API features into a cohesive sequence.
- **Disk Output**: `background.tif`, `crops/*.tif`, `crops/*.json`
- **Injected Metadata**: `NBR_OF_CELLS`

Architecture Verification

- **Script Name**: `test_python_env.py`
- **Architectural Entry Points**:
    - Image Provider: `fetch_images`
    - Processing Logic: `test_versions`
- **Required Data Processors**: None
- **Functional Description**: A diagnostic script used to verify the Python environment configuration. It identifies and prints the installed TensorFlow library version to the console. Useful for asserting that neural network dependencies are installed and accessible by the software.
- **Disk Output**: None
- **Injected Metadata**: None

Explicit Type Hints for UI Mapping

This critical section enumerates the uniquely specialized type hints available for explicitly defining your script parameters. Utilizing these exact types ensures the software's user interface automatically generates and maps the correct input widgets.

- `"images"`: Represents the memory-buffered list of input images ready to be processed by the script engine.
- `"IIS_CONTEXT"`: Represents the overarching operational acquisition context object. This crucial object grants direct access to the live microscope hardware state and programmatic execution methods (e.g., `fireOnDemand()`).
- `"SubDeviceId"`: Exact string representations corresponding to the physical IDs of specific hardware sub-devices (e.g., "Stage X", "Laser 488nm"). The interpreted UI will safely provide a constrained dropdown menu containing only available, connected devices.
- `"channel"`: Represents a channel configuration object.
- `Literal["val1", "val2"]`: Automatically creates a constrained dropdown selection menu populated exclusively with the explicitly specified string options.
- `List[Literal[...]]`: Automatically creates a multi-select, checkable list containing the specified enumeration options.
- `Path`: Represents a standard system file/directory path. The interpreted UI automatically provides a robust graphical file/directory picker dialog.
- `bool`: Generates a standard boolean checkbox or toggle switch (True/False).
- `int`, `float`, `Number`: Generates robust numeric input fields.

> **Reference Examples**
>
> For concrete, structural examples illustrating the usage of every data type mapping, examine the included `input_types.py` foundational script file.

Advanced: Skipping Processing Execution

To optimize computational overhead and deliberately skip processing for a specific acquisition event (e.g., executing analysis *only* on the absolute last optical slice of a thick Z-stack, or only at highly specific chronological timepoints), the `fetch_images` operational function must return an definitively empty list: `[]`.

When `fetch_images` is mathematically evaluated to return `[]`, the overarching system architecture completely bypasses invoking the main, computationally heavy processing function (e.g., `detect_single_cells`).

Example Logic Structure (`_image_provider_functions.py`):

```python
def fetch_zstack(new_image, channel_index: int = 0) -> List[Dict]:
    # ... preliminary metadata extraction ...
    current_focus = meta_dict.get('focusIndex', 0)
    size_z = meta_dict.get('focusSize', 1)

    # Architectural Gate: Only process if we have mechanically reached the final Z-slice
    if current_focus != (size_z - 1):
        return [] # Return empty list, bypassing processing

    # ... otherwise construct and return the required image list ...
```

This specific design pattern guarantees that highly expensive computational evaluations (like neural network inference) run exclusively when the strictly necessary data (e.g., the fully acquired, complete Z-stack) is fully available in system memory.

1.6.2 Photomanipulation (ScanFRAP)

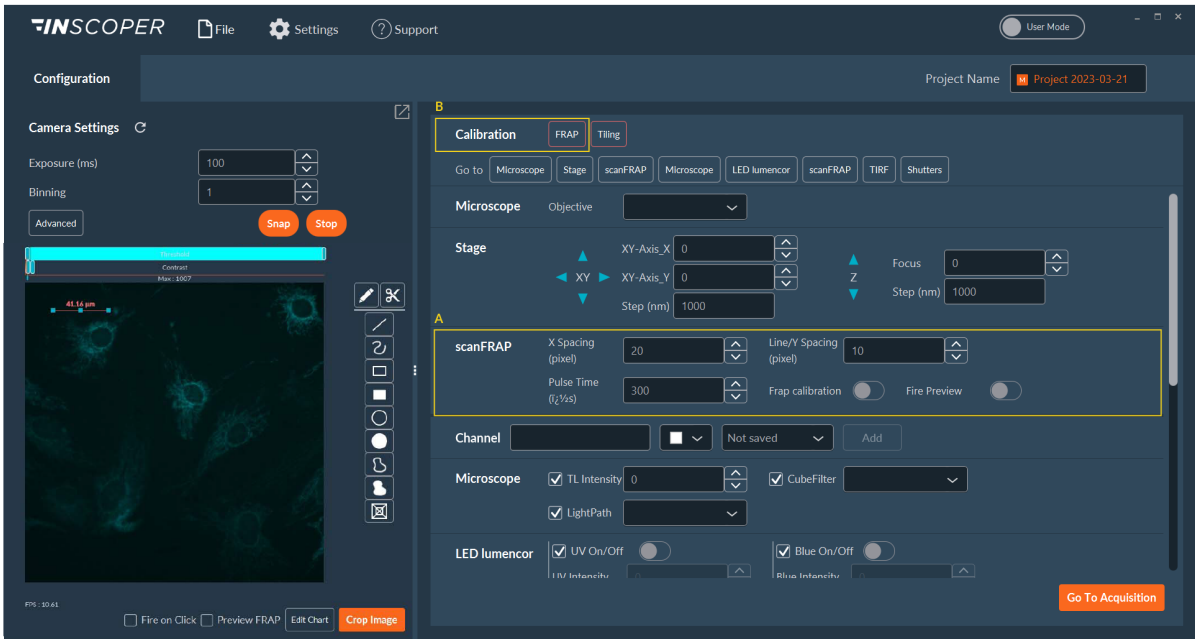
The **scanFRAP** add-on enables targeted laser-based photomanipulation (such as FRAP, photoactivation, or optogenetics) integrated into your imaging sequences.

Configuration & Calibration

GLOBAL CONFIGURATION

Section (A) of the **scanFRAP** Configuration tab controls the geometric point density during active photomanipulation sweeps.

- **X and Y spacing**: Defines the laser spot density used to fill *solid geometric shapes* (e.g., circles, rectangles).
- **Line/Y spacing**: Defines the laser spot density used exclusively for *lines or unfilled shape boundaries*.
- **Fire Preview**: Activates the manipulation laser within currently selected ROIs to allow safe, empirical testing of your power configurations.

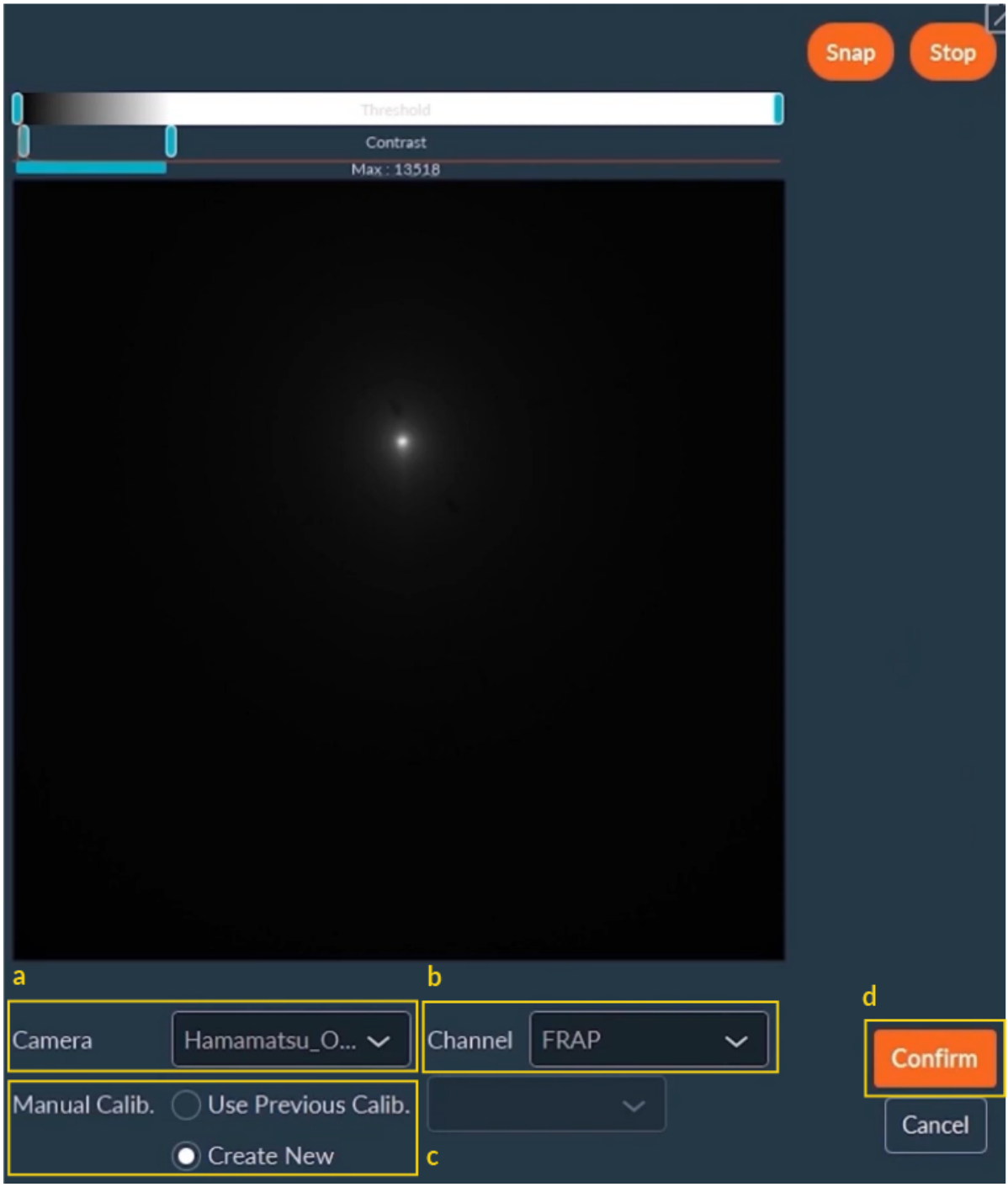


> **Note**
>
> The **Fire Preview** function requires manually drawn ROIs and the correct manipulation channel to be actively selected.

SPATIAL CALIBRATION

scanFRAP requires spatial calibration to ensure targeting accuracy. Recalibrate the galvanometer mirrors before each experimental session.

Navigate to the generic calibration menu (B) and click **FRAP**.



- **a**. Select the currently active imaging camera.
- **b**. Select the pre-configured laser channel designated for photomanipulation.
- **c**. Choose an existing archived calibration profile or initiate a new one.
- **d**. Click **Confirm** to route the optical settings.

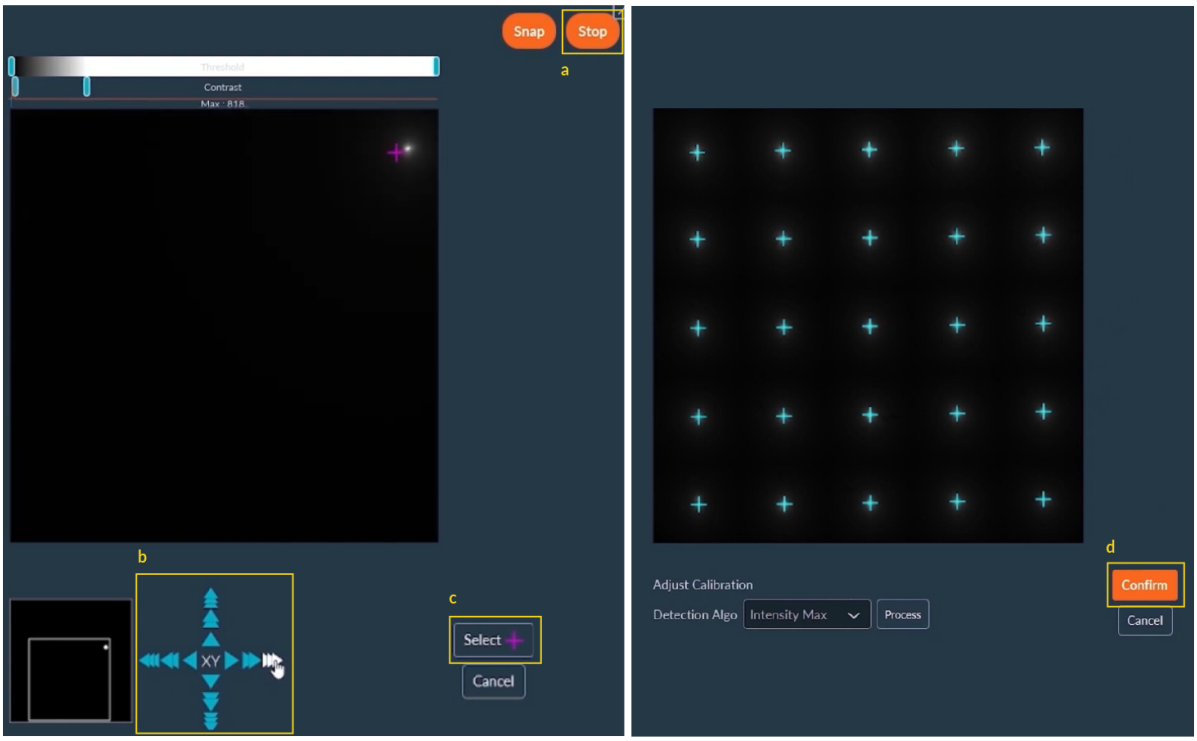
> **Tip**
>
> Always perform your initial or routine spatial calibration utilizing a *fluorescent reference slide* (e.g., standard plastic/glass). This guarantees a strong, uniform signal matrix without risking irreversible photobleaching of your actual, sensitive biological sample.

> **Warning**
>
> Maintain optimal Z-focus throughout this process. Laser targeting accuracy depends on a sharp focal plane.

New calibration

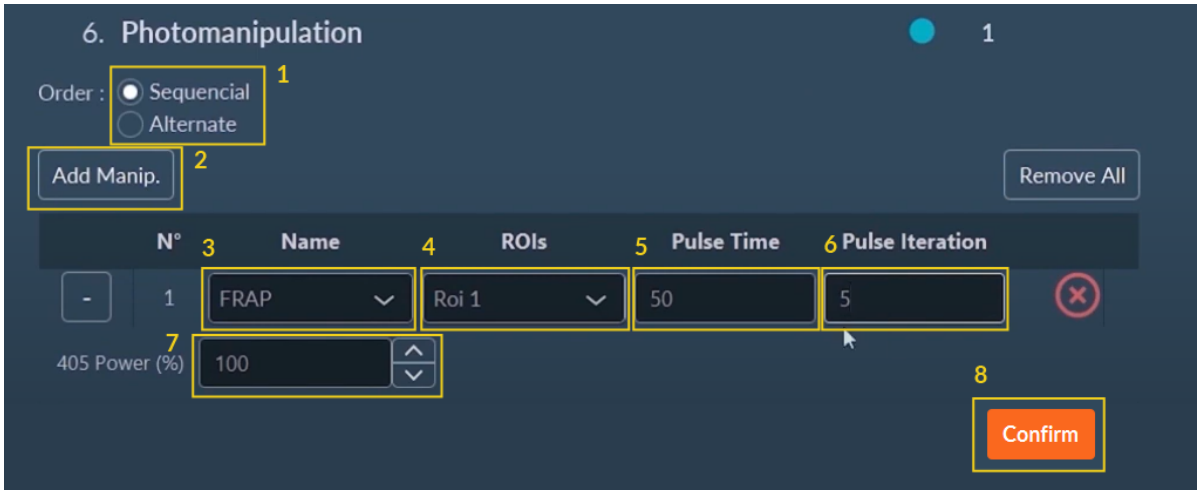
- **a**. Activate **Live** mode to visualize the reference slide.

- **b**. Use the virtual joystick to align the laser pointer with the software's virtual crosshair.
- **c**. Click **Select**. Repeat this alignment process three times at different XY coordinates across the field of view.
- **d**. The software will generate a verification pattern. Validate its geometric regularity (e.g., straight lines, sharp corners), then click **Confirm** to accept the calibration matrix.



Sequence Acquisition

scanFRAP procedures are incorporated into your sequence via the **Photomanipulation dimension**.



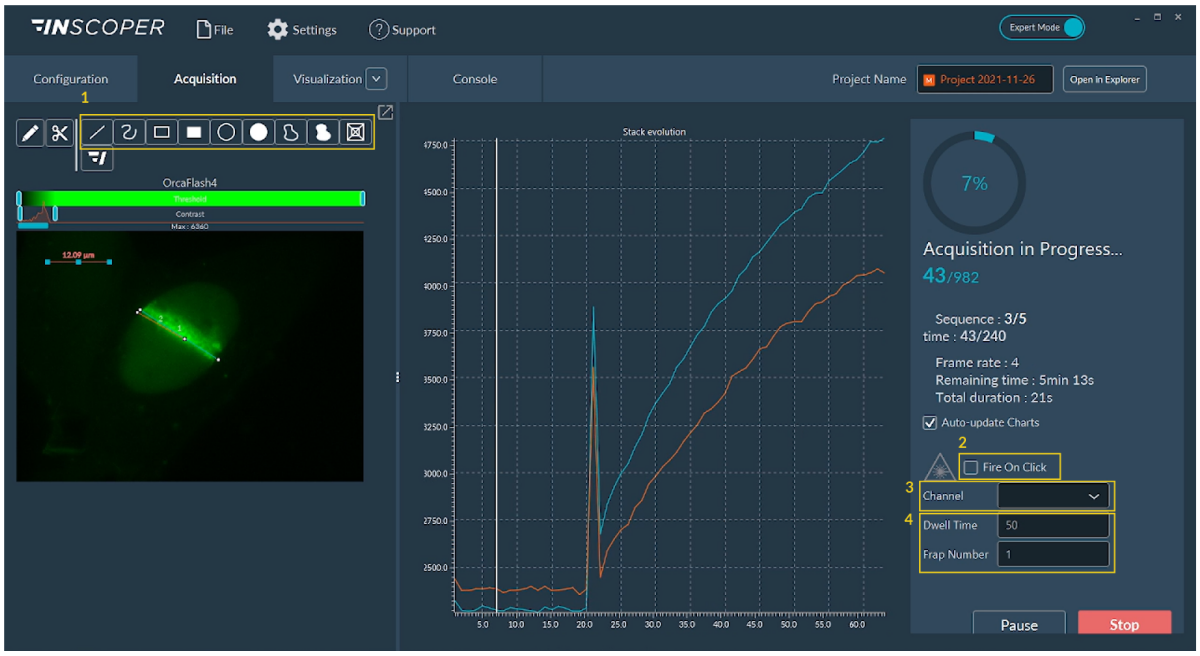
To configure the **Photomanipulation** dimension:

1. Select the scanning mode: Scan assigned ROIs one at a time (**Sequential**) or interleave the scanning paths between them (**Alternate**).
2. Click **Add Manip.** to instantiate a new photomanipulation event block.
3. Select the pre-configured laser channel assigned to drive the photomanipulation.
4. Select and activate all target ROIs on the sample preview.
5. Adjust the total **pulse time** (the time required for the galvanometers to move from the previous coordinates plus the programmed exposure dwell time).
6. Define the total number of consecutive exposure iterations.
7. Adjust the laser output power (percentage).
8. Click **Confirm** to save the sequence parameters.

> **Note**
>
> Before initiating a complex photomanipulation experiment, empirically ensure that the scanFRAP galvo module is acutely calibrated.

Live Acquisition Overrides

During an actively running time-lapse acquisition sequence, you can monitor experimental progress in real-time, reviewing both live images and plotted raw intensity data for each designated ROI. The unique **Fire On Click** feature allows you to selectively photomanipulate entirely new, spontaneous areas on the fly without interrupting the sequence timeline.

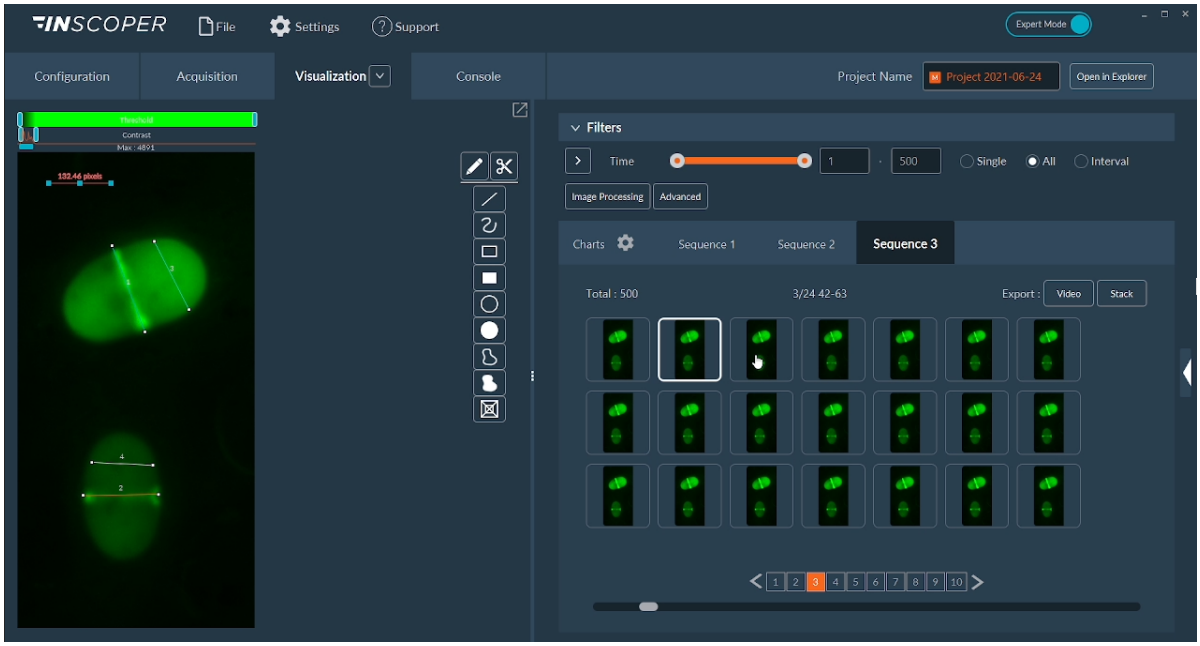


To deploy **Fire On Click**:

1. Draw a new target ROI on the live, incoming image stream.
2. Check the **Fire On Click** enablement box.
3. Verify the accurately calibrated manipulation channel is actively selected.
4. Adjust the temporal **Dwell Time** (pulse duration per pixel) and the **FRAP number** (total physical iterations).
5. Click directly on the newly active ROI in the viewer to immediately demand and execute the photomanipulation sweep.

Data Visualization

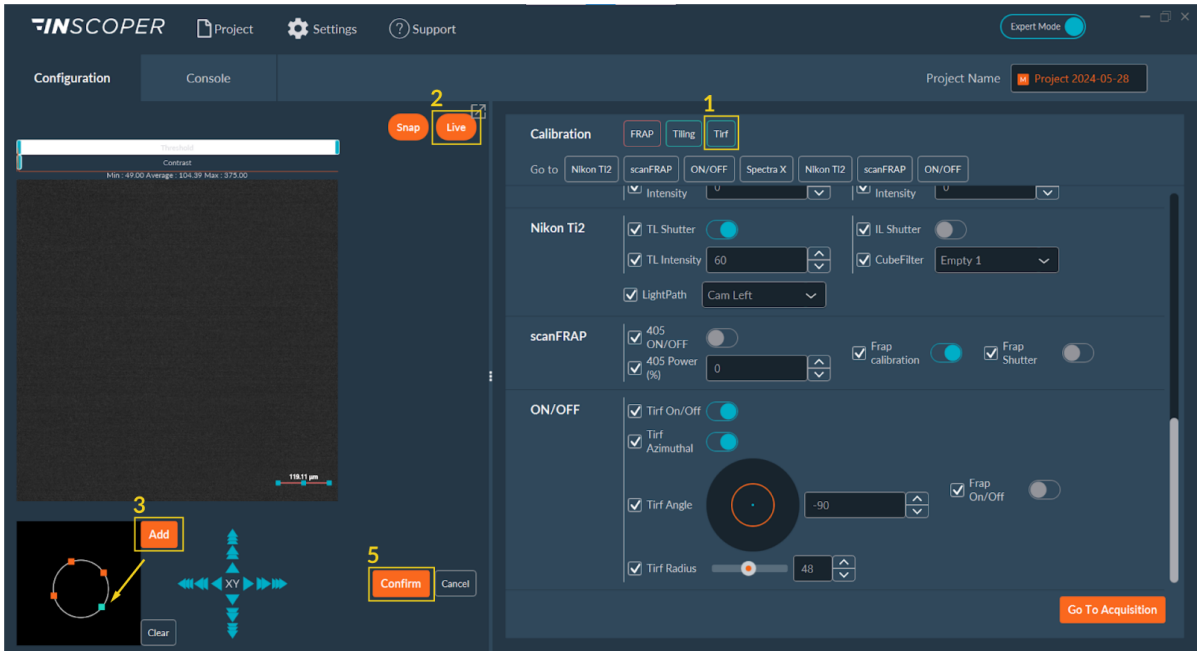
The **Visualization** tab displays the sequence imaging results alongside graphs tracking the fluorescence intensity over time within all defined ROIs.



1.6.3 TIRF

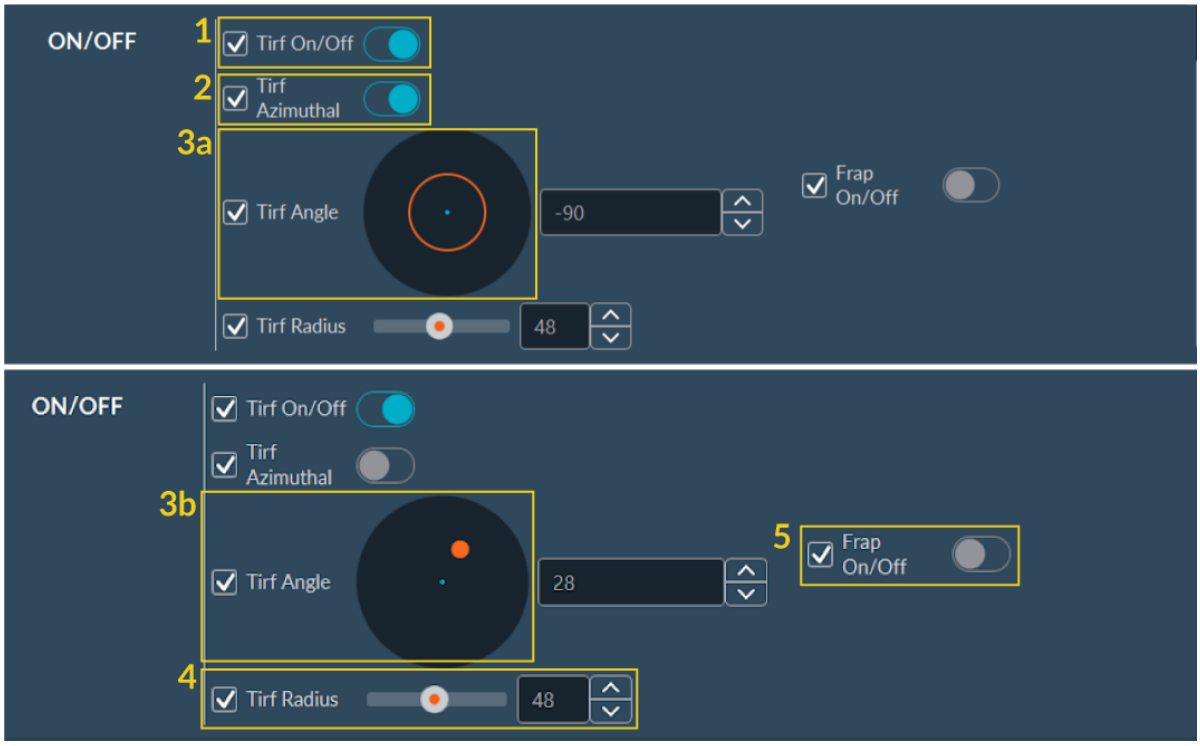
Total Internal Reflection Fluorescence

Optical Calibration



1. Navigate to the Calibration section and click **TIRF**.
2. Enable the **Live** camera feed to observe the sample.
3. Click **Add** to insert a calibration point into the schematic field of view. This blue control point corresponds to the TIRF laser position. Drag this point radially to adjust the laser incidence angle until you achieve critical angle TIRF illumination (evidenced by the loss of out-of-focus background fluorescence).
4. Repeat the coordinate placement process to define additional calibration positions if needed for different sample topologies or wavelengths.
5. Click **Confirm** to save the spatial calibration matrix.
6. Repeat this angular calibration protocol for each discrete laser wavelength intended for TIRF imaging, as critical angles vary with excitation wavelength.

Configuration



1. **Activate TIRF**: Toggle the switch to enable or disable TIRF mode hardware control.
2. **TIRF Azimuthal**: Toggle **TIRF Azimuthal** to enable circular illumination, minimizing interference fringes and uneven illumination.
3. **Option**:
    - a. *Activated*: The interface displays a diagram of the circular laser scanning pattern at the currently set TIRF angle.
    - b. *Deactivated*: The interface displays a single laser position (orange dot). Specify its coordinates using the **TIRF Angle** parameter field.
4. **TIRF Radius**: Adjust the spatial radius to set the laser incidence angle at the objective lens aperture. This control determines the illumination regime: Widefield (Epi), Highly Inclined and Laminated Optical Sheet (HILO), or TIRF.
5. **FRAP**: Toggle the corresponding switch to activate the targeted photobleaching (FRAP) function.

CREATING A TIRF CHANNEL

Save the spatial parameters (**Azimuthal mode**, **Angle**, and **Radius**) to the designated TIRF channel preset in the Configuration tab to properly execute automated TIRF imaging.

Sequence Acquisition

Select the saved TIRF channel to perform a TIRF acquisition during the multi-dimensional sequence.

1.6.4 liveDRIM

Documentation for the liveDRIM feature is currently under development and will be available soon.

1.6.5 liveSR

Documentation for this feature is coming soon (expected in a future release).

1.6.6 liveRATIO

The **liveRATIO** add-on enables real-time ratiometric imaging, allowing you to quantify the ratio between two emission wavelengths over time (e.g., FRET, calcium imaging). The `Ratiometric imaging` data processor drives this function.

Hardware Configuration

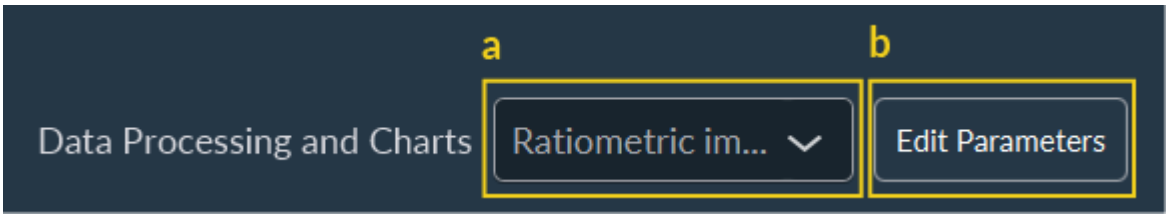
For physical hardware and baseline channel setup, please refer to the standard Configuration section.

Sequence Acquisition

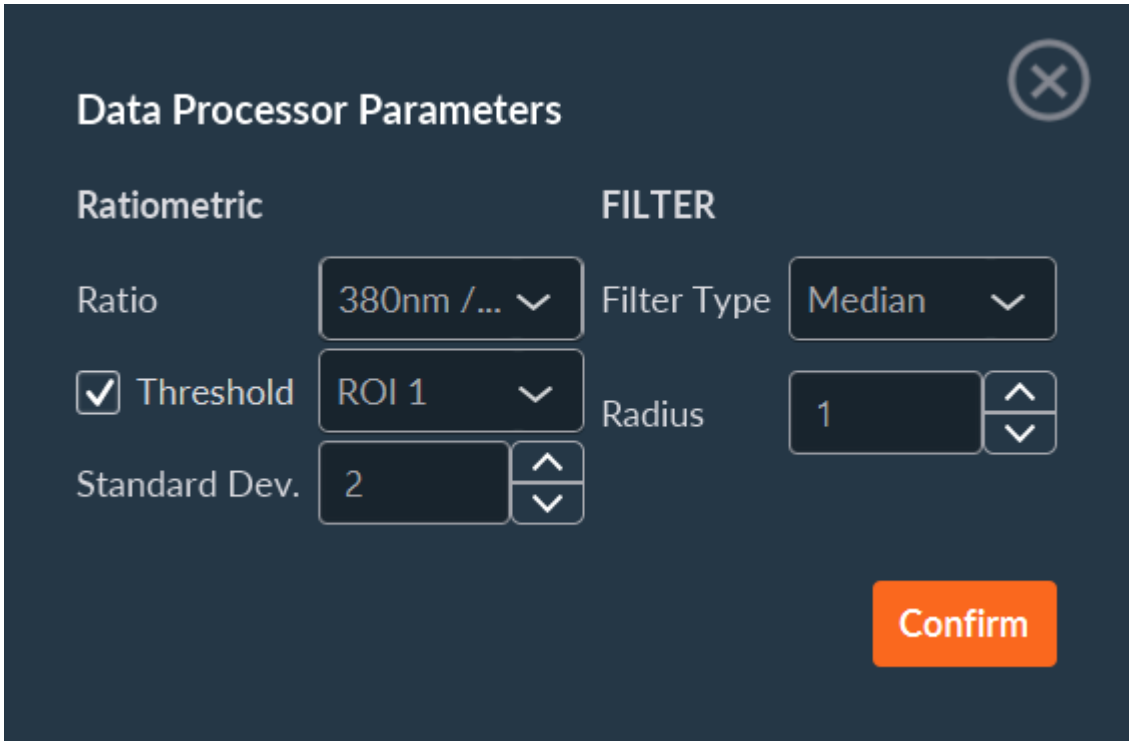
1. WAVELENGTH SETUP

1. Navigate to the **Multi-Channels** dimension within the **Acquisition** tab.
2. Configure two or more distinct optical wavelengths for your chronological sequence.

2. DATA PROCESSING CONFIGURATION

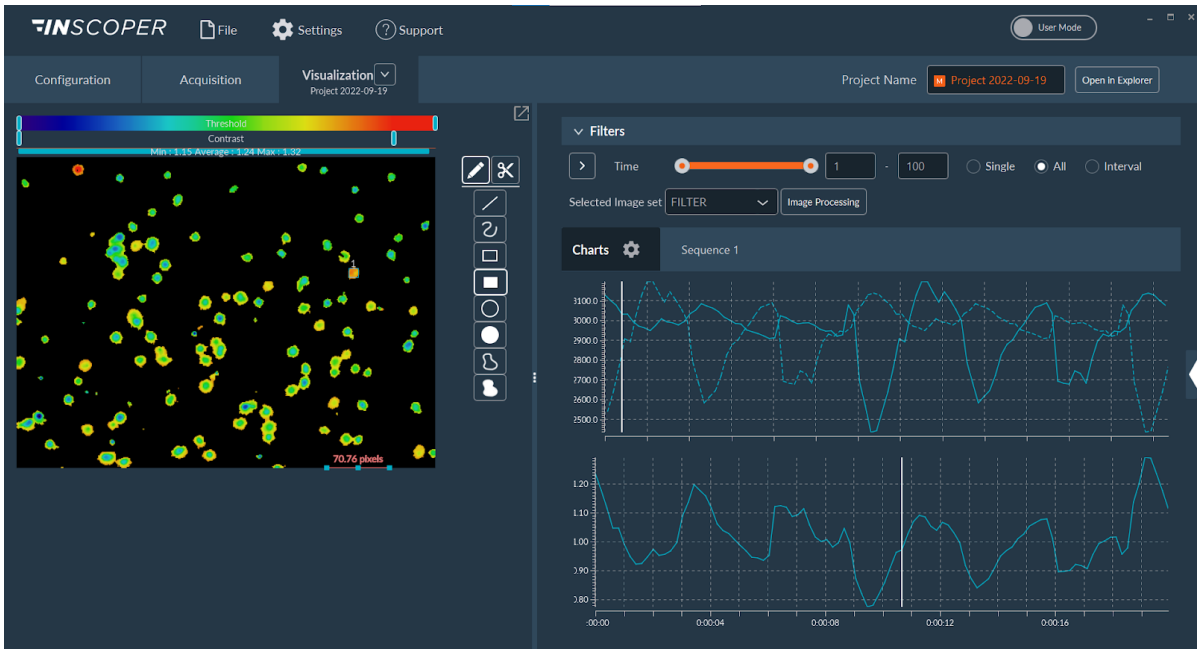


- a. Choose **Ratiometric imaging_Despeckle** from the **Data processing and Chart** drop-down menu.
- b. Click **Edit Parameters** to open the module's settings.
- c. Specify which channels form the **numerator** and **denominator** for your calculated ratio.
- d. *(Optional)* Enable background noise removal:
  - If enabled, draw a Region of Interest (ROI) in an area devoid of signal. This ROI serves as the background reference.
  - Specify the standard deviation multiplier for the background subtraction threshold calculation.
- e. Select the spatial filtering method (median or average filter).
- f. Define the radius (in pixels) for the spatial filter.
- g. Click **Confirm** to save your processing parameters.



Data Visualization

The **Visualization** tab displays both the raw sensor intensity data and the calculated ratiometric data for any drawn ROIs. These data plots update in real-time whenever an ROI is created, adjusted, or removed.



1.6.7 Confocal MAICO

The Confocal MAICO provides a solution for laser scanning confocal microscopy, designed for integration with microscopes from any manufacturer. It is used for routine fluorescence imaging, functioning as a dedicated benchtop system or as a primary instrument in a core facility.

System Startup

Before launching the Inscoper Software, you must **turn on and calibrate the confocal MEMS unit**:

1. Turn the physical Power key to **On**. The system LED transitions from solid orange to a blinking green light.
2. Wait 5 to 10 minutes for the unit to reach thermal stabilization. The **Laser Calibration** button in the startup interface turns green once the system is ready.
3. Press the **Laser Calibration** button to initiate the automated hardware calibration.

> **Important**
>
> You must open the microscope's optical shutter for this step to succeed.

4. The green calibration light turns off upon successful completion.
5. You may now safely launch the main Inscoper Software.

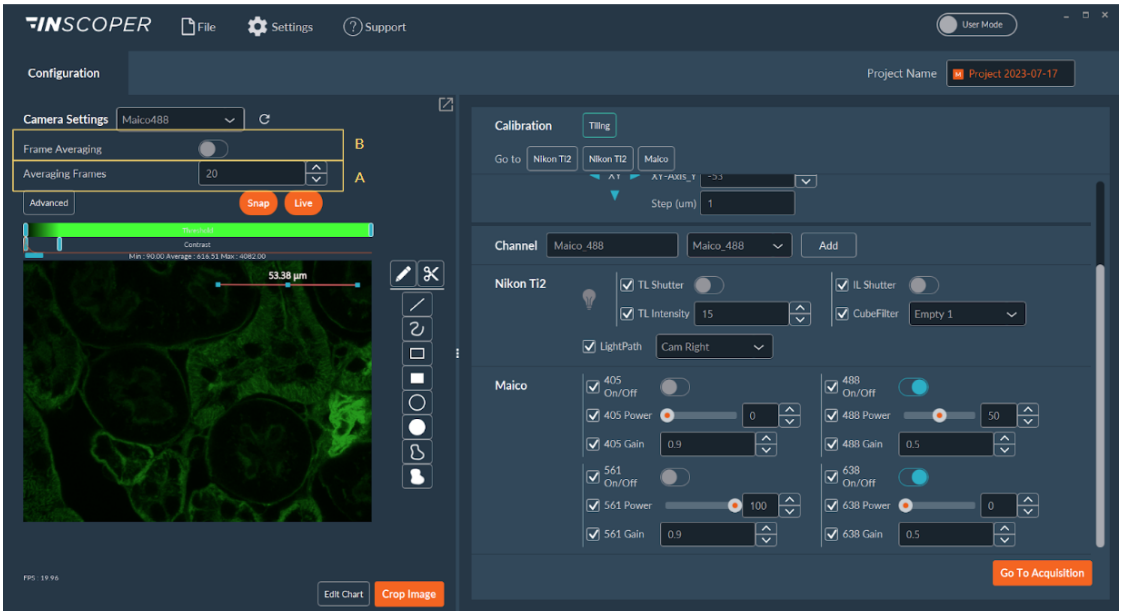
Channel Configuration

The **Configuration** tab provides direct access to the laser and detector hardware parameters, allowing you to finely tune laser excitation power and detector amplification (gain).

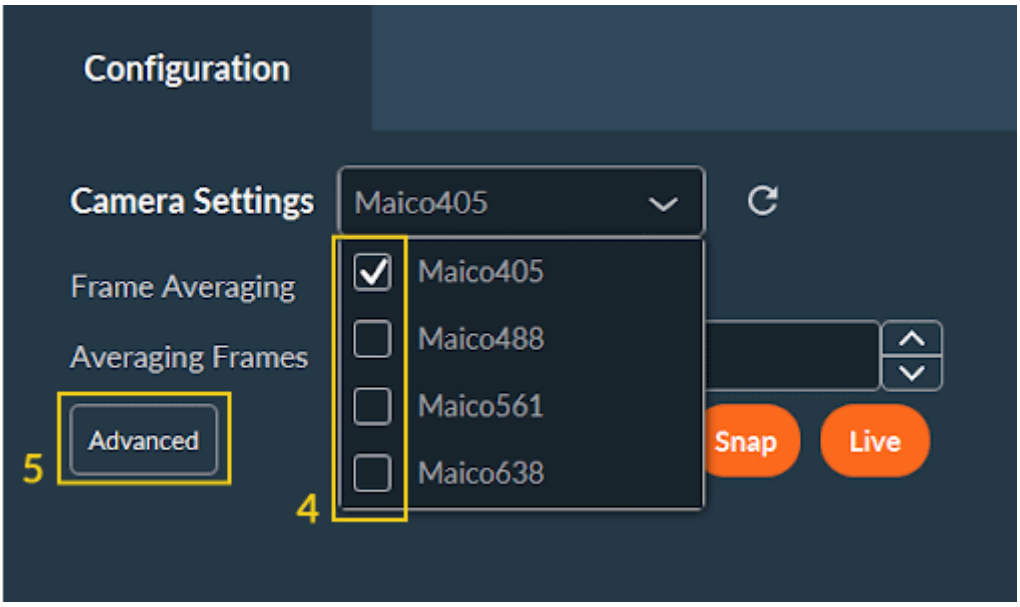
- Click **Add** to apply parameter changes temporarily for the current live experiment session.
- Switch to **Expert Mode** and click **Save** to create a new, reusable channel preset or to store your current settings permanently to the database.

SETTING UP A MAICO CHANNEL

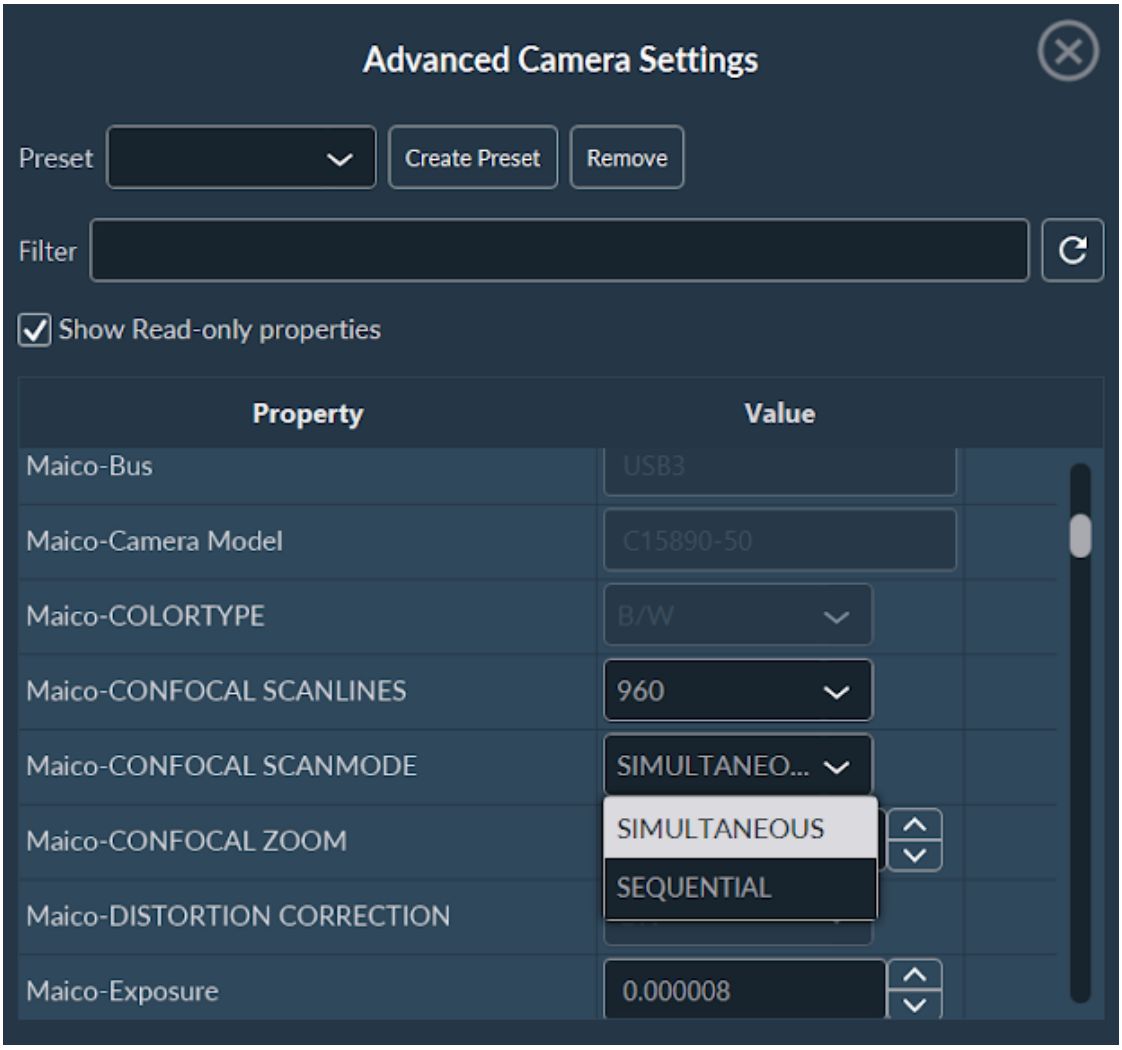
1. For multicolor imaging sequences, create a **single channel preset that encompasses all required laser lines**, even if you perform sequential (line-by-line) acquisition later.
2. Adjust the excitation intensity (% power) and detector gain for each wavelength.
3. **Frame Averaging**: Enable frame averaging across all active detectors to improve the image Signal-to-Noise Ratio (SNR):
    1. Set the number of consecutive frames to average.
    2. Toggle the averaging feature on.



4. **Detection View**: Select the detection channels you want to monitor by checking their respective boxes in the interface.



5. **Advanced Options**: Access detailed scanning hardware settings by clicking **Advanced**:
    - **Scan line**: Select the number of vertical scan lines (960, 480, or 240). Lower line counts significantly increase the frame rate but reduce spatial resolution.
    - **Scan Mode**: Choose between Sequential (minimizes spectral bleed-through) or Simultaneous (maximizes acquisition speed) excitation.
    - **Zoom**: Choose between optical Zoom 1 or 2.



6. Click **Go to Acquisition** to transition to the sequence builder.

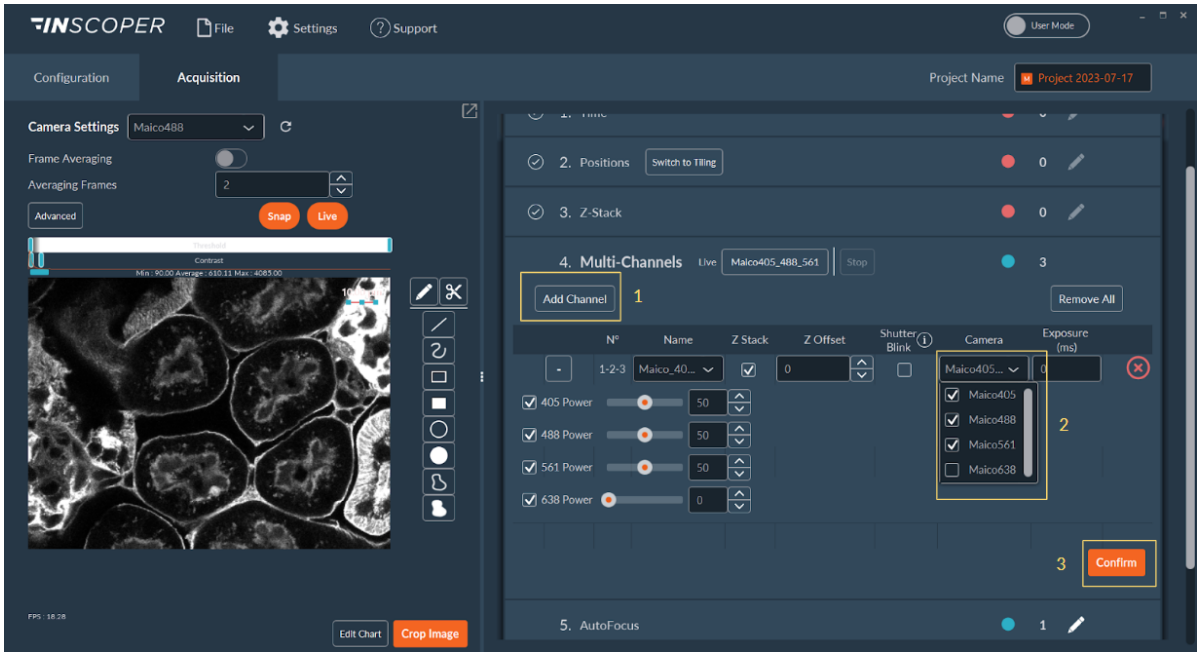
Sequence Acquisition

Configure your experimental sequences in the **Acquisition** tab, following the standard procedures outlined in the global User Guide.

> **Warning**
>
> For MAICO confocal acquisitions, the only procedural deviation from standard widefield imaging occurs within the **Multi-Channels** dimension setup.

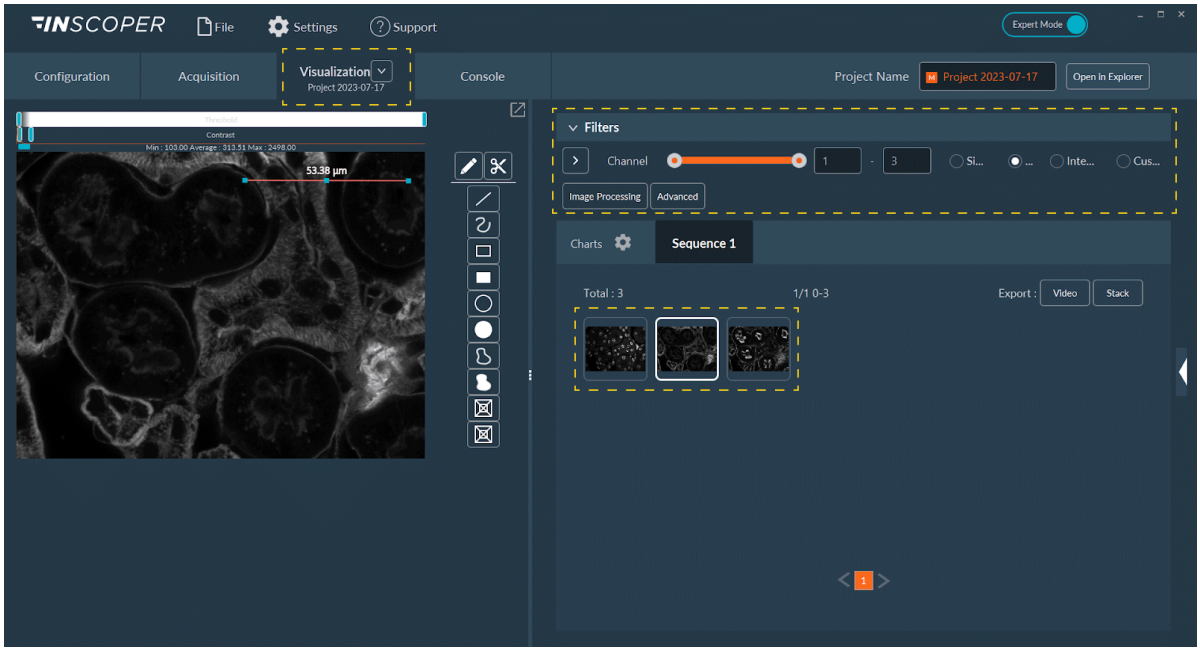
THE MULTI-CHANNELS DIMENSION

1. Click **Add Channel** to insert your previously configured multi-laser MAICO channel preset into the sequence.
2. Check the box corresponding to your desired confocal detection mode.
3. Click **Confirm** to route the configuration.



Data Visualization

Immediately following the acquisition, the **Visualization Tab** becomes active. You can browse, filter, and review your confocal image stacks by their assigned dimensions (e.g., Z, T, C).



Tip

If your system includes an active *Microvolution* license, you can perform advanced 3D deconvolution directly within Inscoper I.S. by clicking the **Image Processing** button.

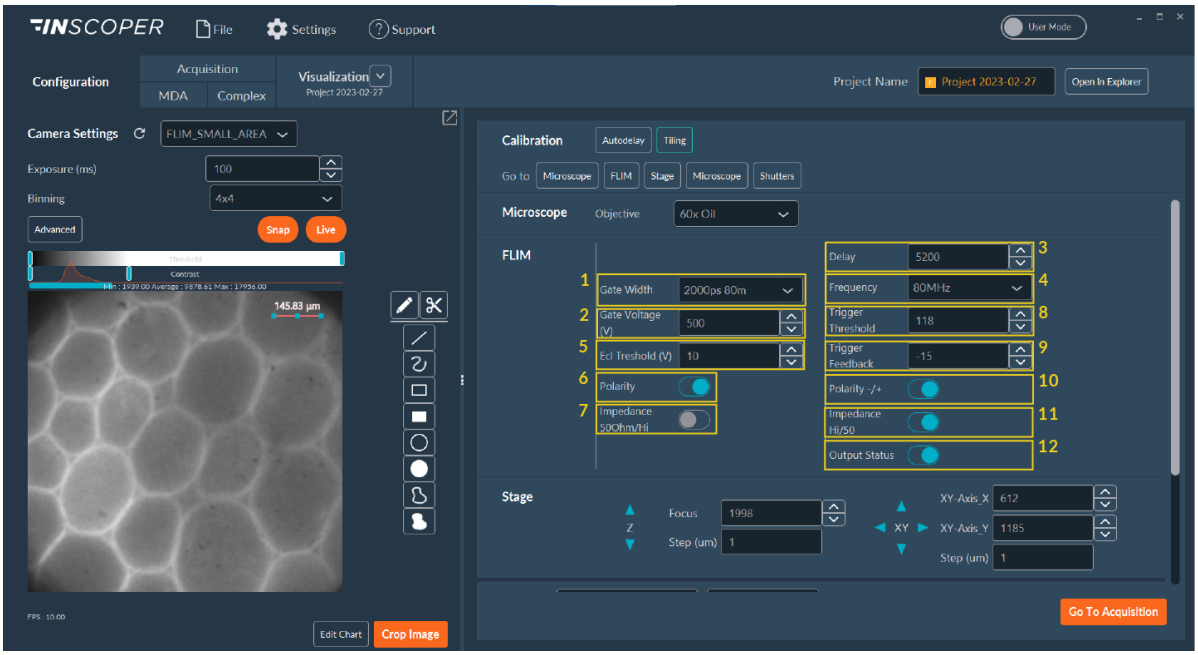
1.6.8 Time Gated Domain FLIM (fastFLIM)

fastFLIM is a hardware and software solution designed to measure mean fluorescence lifetime using a widefield, camera-based microscope architecture. fastFLIM measurements integrate seamlessly with other imaging dimensions, including **Multi-positions**, **Tiling mosaics**, **Multi-channel overlays**, and **Z-stacks**.

Unlike traditional point-scanning FLIM (e.g., TCSPC), which is typically slow, **fastFLIM** is a **camera-based FLIM technique** that employs a time-domain approach combined with a time-gated image intensifier. This captures the lifetime of the entire field of view simultaneously (or rapidly via spinning disk), providing significantly higher acquisition speeds and minimal phototoxicity. This makes it highly effective for observing rapid biological dynamics and biosensors in living cells.

Hardware Configuration

The **Configuration** tab provides direct access to the critical FLIM timing and gating parameters:

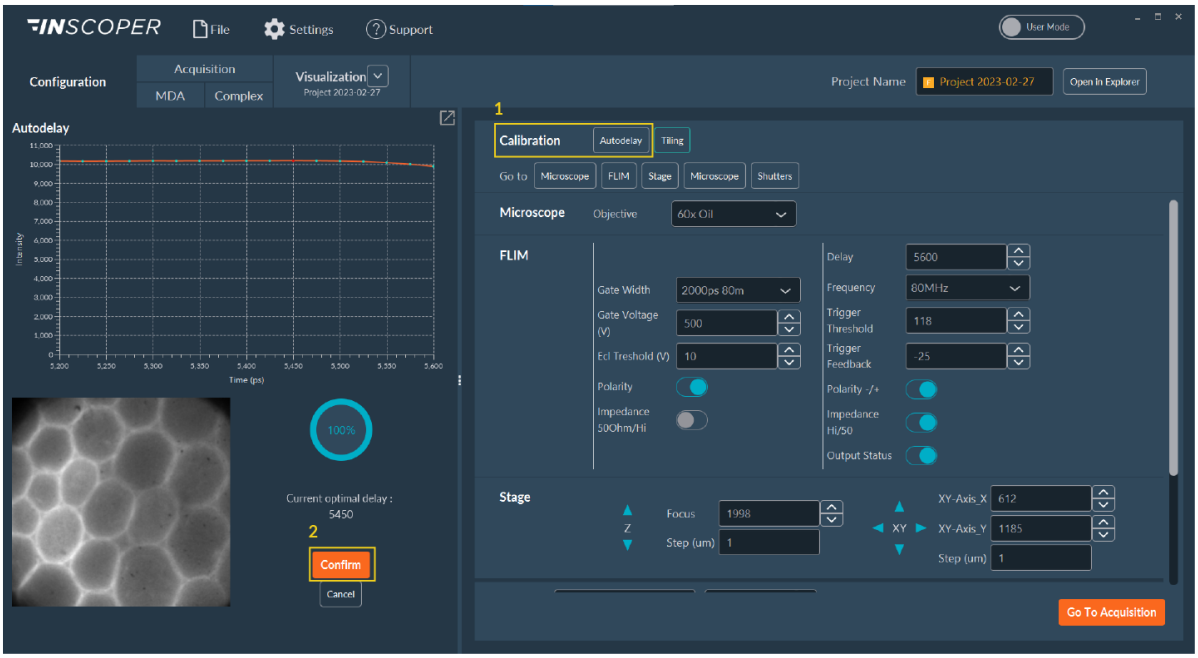


1. **Gate Width**: Specifies the absolute duration of the temporal detection window (gate).
2. **Gate Voltage**: Sets the multi-channel Plate (MCP) potential voltage, which directly controls the internal signal amplification (gain) of the intensifier.
3. **Delay**: The precise temporal offset (in picoseconds) applied relative to the pulsed laser synchronization signal.
4. **Frequency**: The operational repetition rate of the central delay generator.
5. **ECL Threshold**: Sets the electrical noise floor threshold for the input synchronization signal routed to the intensifier.
6. **Polarity**: Configures the detector logic to correctly identify the leading or trailing edge of the delay generator's synchronization pulse.
7. **Impedance**: Adjusts electrical impedance matching to ensure artifact-free signal detection.
8. **Trigger Threshold**: Sets the electrical noise floor threshold for the input synchronization signal routed to the delay generator.
9. **Trigger Feedback**: (Specific to legacy delay generator models). Displays a diagnostic value verifying lock synchronization between the delay generator and the pulsed laser. Optimal operational values hover near 0, safely within a [-100, +100] operational range.
10. **Polarity (-/+)**: Configures the delay generator logic to correctly detect the laser's physical synchronization pulse.
11. **Impedance (Hi/50Ω)**: Adjusts electrical impedance matching specifically for the laser synchronization line.
12. **Output Status**: Manual override for the optical shutter control.

> **Note**
>
> The visibility and editability of specific parameters adapt dynamically based on the specific intensifier and gating hardware model recognized by the system.

AUTODELAY PULSE CALIBRATION

> **Important**
>
> Before initiating a FLIM experimental sequence, execute an **Autodelay** calibration. This step synchronizes the pulsed laser excitation with the initial temporal gate opening, ensuring accurate photon capture and reliable lifetime calculations.



1. Click **Autodelay** to launch the automated hardware calibration protocol.
2. Wait for the software algorithm to calculate the optimal temporal offset. Once displayed, click **Confirm**. This calibrated picosecond value will automatically populate the master **Delay** field in the Configuration tab.

Sequence Acquisition

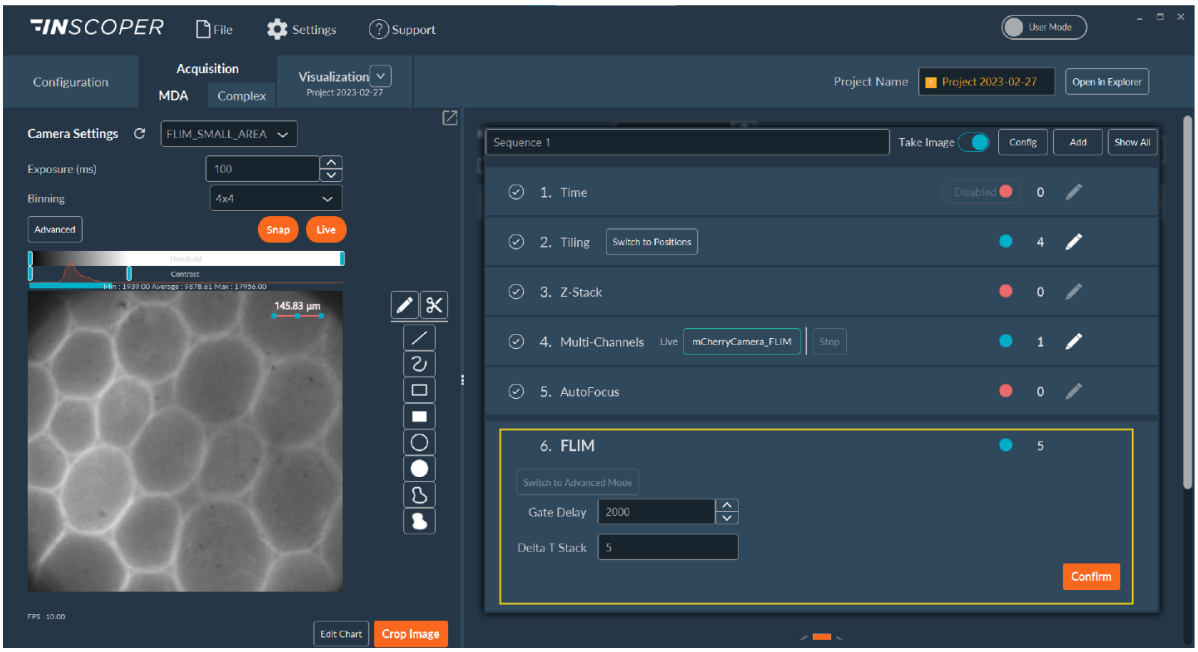
With the hardware configured and calibrated, proceed to the **Acquisition** tab to build the imaging sequence.

Configure your imaging channels natively using the **Multi-channel** dimension, ensuring you explicitly define both your standard widefield fluorescence channels and your dedicated FLIM measurement channels.

Within the dedicated **FLIM dimension** panel, the interface behavior adapts computationally based on the number of active FLIM channels designated:

SIMPLE MODE

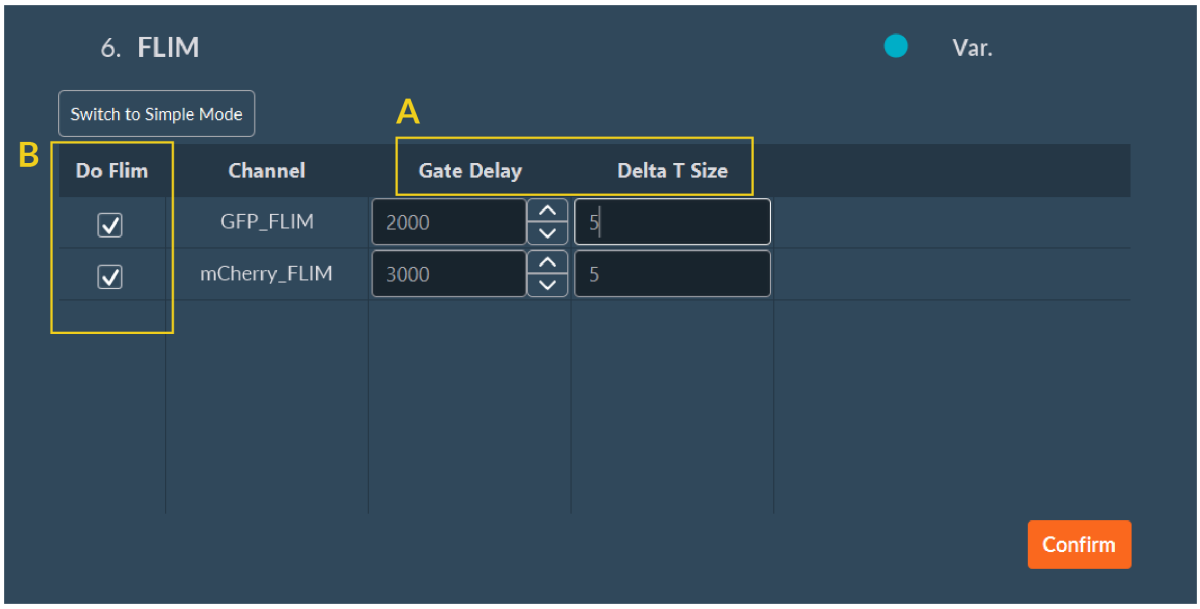
This mode activates automatically when only **a single FLIM channel** is present in the sequence.



- **Gate Delay**: Defines the incremental temporal shift (interval) between successive measurement gates across the decay curve.
- **Delta T stack**: Defines the total number of sequential temporal images (discrete time points) to acquire for the complete FLIM measurement curve.

ADVANCED MODE

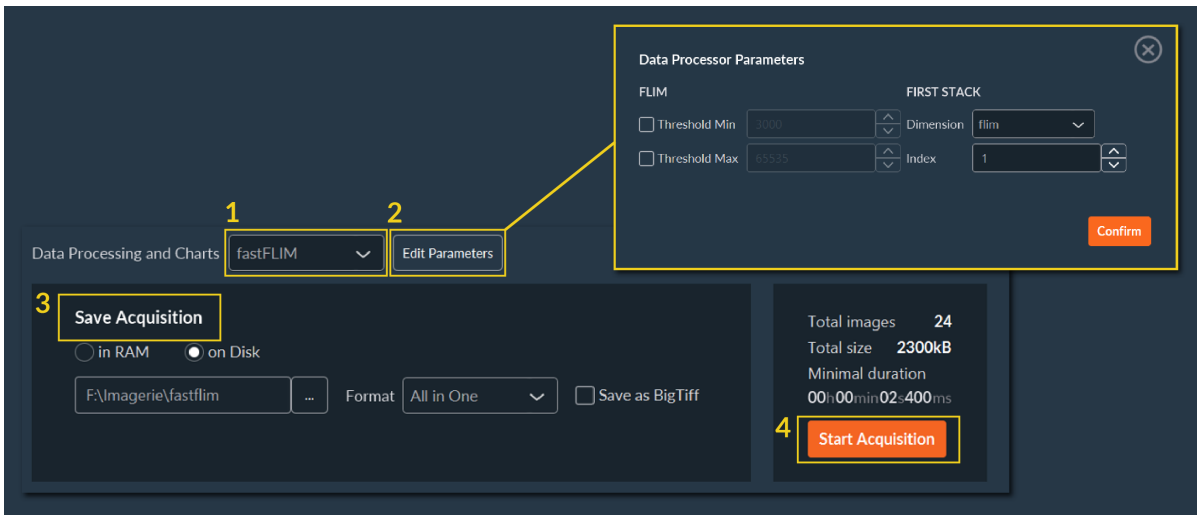
This mode activates automatically when **multiple FLIM channels** are configured in the sequence, granting granular control over complex multi-channel lifetime measurements.



- **A**: Define unique, independent **Gate Delay** and **Delta T Size** parameters tailored computationally for **each specific channel**.
- **B**: Toggle active FLIM measurement execution on or off for individual channels by utilizing the **Do Flim** checkboxes.

FINALIZING ACQUISITION

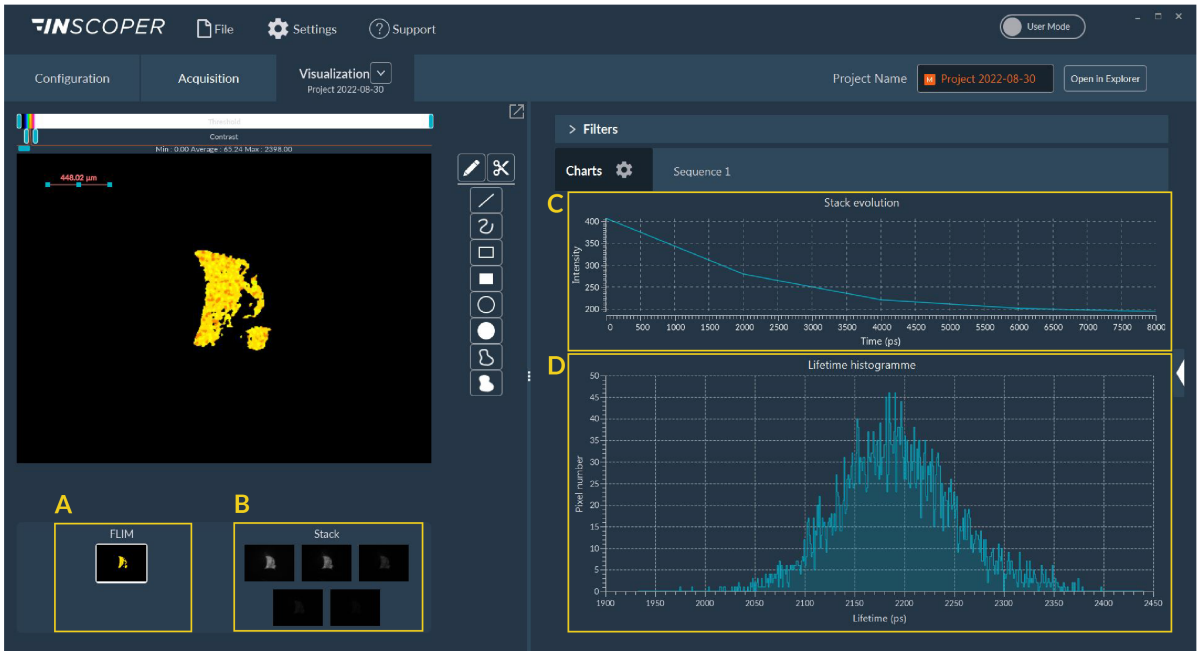
Once your temporal FLIM parameters are locked, click **Confirm** on the dimension module. To finalize the data routing workflow:



1. Navigate to the Save menu. Select **fastFLIM** from the **Data Processing and Charts** computation drop-down menu.
2. Click **Edit Parameters** to define the absolute minimum and maximum **Intensity Thresholds** (photon counts) required for valid pixel-wise FLIM calculation. Click **Confirm**. (Pixels below this threshold are excluded from the decay fit).
3. Specify your absolute destination save folder path.
4. Click **Start Acquisition** to initiate the sequence.

Data Visualization

The **Visualization tab** provides FLIM analytical results computed at the end of the acquisition. It displays calculated FLIM maps alongside raw sensor intensity, integrated lifetime distributions, and point-wise intensity decay curves.



- **A**: The pseudo-colored, computed FLIM image map.
- **B**: The raw, time-gated intensity images utilized for the decay calculation.
- **C**: The plotted intensity decay curve for the selected pixel or ROI.
- **D**: The global pixel lifetime distribution histogram.

All generated multi-dimensional images and statistical charts are saved automatically to your specified destination folder upon compute completion.

1.7 Legal

1.7.1 FCC/IC certification

Any changes or modifications to this equipment not expressly approved by Inscoper may cause harmful interference and void the FCC authorization to operate this equipment.

This equipment has been tested and found to comply with the limits for a Class A digital device, pursuant to Part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment. This equipment generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with the instruction manual, may cause harmful interference to radio communications. Operation of this equipment in a residential area is likely to cause harmful interference in which case the user will be required to correct the interference at his own expense.

This device must be professionally installed.

1.7.2 Copyright

The copyright in this document and the associated drawings are the property of INSCOPER and all rights are reserved. This document and the associated drawings are issued on the condition that they are not copied, reprinted, or reproduced, and that their contents are not disclosed, except in connection with the authorized use of the system.

The publication of information in this document does not imply freedom from any patent or proprietary right of Inscoper or any third party.

INSCOPER and the INSCOPER logo are trademarks of INSCOPER Company (INSCOPER SAS - 3771 boulevard des Alliés, 35510 Cesson-Sévigné - France). INSCOPER includes technology covered by the following patents:

- US Patent No. US10330911,
- EP Patent No. EP3123149,
- FR Patent No. FR3019324,

This product is updated periodically, and revisions will be incorporated into new editions of the user documentation.

1.7.3 Disclaimer

The information contained in this manual is provided on an “as is” basis, without any warranties, conditions or representations of any kind, whether express, implied, statutory or otherwise, including, but not limited to, any warranties of merchantability, non-infringement or fitness for a particular purpose.

In no event shall Inscoper be liable for any loss, or for any direct, indirect, special, incidental, consequential, or other damages, regardless of cause, whether arising in contract, tort, or otherwise in connection with the use of the information provided herein.