

MCAM User Manual

ramona

MCAM® User Manual



Copyright 2018-2026 Ramona Optics, Inc. All rights reserved.
Durham, NC.

The information contained in this document is subject to change without notice.

DISCLAIMER

The Ramona Optics Multi-Camera Array Microscope (MCAM®) is a Gigapixel Microscope that is provided as a beta-unit for use by the user "AS-IS" and any express or implied warranties, including but not limited to, the implied warranties of merchantability and fitness for a particular purpose are disclaimed.

LICENSING

A copy of the license agreement is included along with the software package. If you have not received this agreement please contact Ramona Optics.

All third party licenses can be found within the MCAM Graphical User Interface GUI by selecting "About" in the "Help" menu.

TRADEMARKS

Ramona Optics® is a registered trademark of Ramona Optics, Inc.
All other trademarks are the sole property of their respective owners.

"Python®" and the Python® logos are trademarks or registered trademarks of the Python Software Foundation, used by Ramona Optics with permission from the Foundation.

CONTACT

Ramona Optics, Inc.
302 E Pettigrew Street
Durham, NC 27701

Phone: (919) 797-9975
General Email: info@ramonaoptics.com
Support Email: help@ramonaoptics.com
Website: ramonaoptics.com



Table of Contents

Safety Warnings and Precautions.....	8
MCAM Overview.....	9
Technical Specifications -- Kestrel.....	9
Equipment Ratings.....	13
Kestrel (2022 Revision) Features.....	14
Kestrel (2024 Revision) Features.....	15
Vireo Features.....	16
Getting Started.....	17
Recommended Hardware.....	17
Technical Support.....	19
Equipment Maintenance and Service.....	20
Servicing the MCAM.....	20
Cleaning and Decontamination.....	21
Installation.....	21
Boot Procedure.....	23
Kestrel Reflection Illumination Module (Optional).....	24
Viewing Data Acquired by the MCAM.....	28
MCAM Viewer Installer.....	28
Kestrel.....	29
Graphical User Interface (GUI).....	29
Opening and Closing the MCAM.....	29
GUI Navigation.....	30
MCAM Settings.....	33
Zebrafish Panel Settings.....	35
VIBRATION STIMULUS.....	35
Save and Export Images.....	37
Save Video or Z-Stack.....	40
Annotation Tools.....	42
Color Picker.....	42
Length Tool.....	44
Advanced Settings.....	45
MCAM Viewer.....	49
Video Playback.....	50
MCAM Data Analysis.....	51
Acquisition Considerations.....	51
Recommended Video Acquisition Parameters.....	51
Imaging Models and Pixel Binning.....	52
Movement Blur: Frame-Rate vs. Exposure.....	53



Sensor Shape for Frame-Rate Optimization.....	53
Video Compression.....	55
Creating a Keypoint File.....	55
Square Well Plate Keypoints.....	59
Compress the Video.....	60
Compression Options.....	64
Video Orientation.....	65
MCAM User Interface Activity Metric Analysis.....	66
MCAM Workflows Manual.....	66
Zebrafish Embryonic Photomotor Response Assay.....	66
MCAM Viewer Activity Metric Analysis.....	68
MCAM Viewer Activity Metric Outputs.....	69
ACTIVITY METRIC COMPOSITE.....	69
ACTIVITY PLOTS.....	70
ACTIVITY DATA (.CSV).....	70
MCAM Zebrafish Analysis.....	71
Machine Learning Tracking.....	71
Tracking Analysis Outputs.....	74
OUTPUT FILES.....	74
DATA SIZES.....	76
FROM THE MCAM GUI.....	77
FROM A PYTHON SCRIPT.....	77
Visualizations.....	78
PLOTTED TRACKS.....	78
COMPOSITE VIDEO.....	79
Assays.....	80
THIGMOTAXIS.....	80
FISH LENGTH.....	81
SKELETON ANALYSIS.....	82
TAIL BEND ANALYSIS.....	84
EYE ANALYSIS.....	87
EYE ANALYSIS IMAGING PARAMETERS.....	90
Tracking Data Filtering.....	90
ANOMALY DETECTION.....	91
DENOISING.....	92
TIME BINNING.....	94
Zebrafish Segmentation.....	95
SEGMENTATION GUI PANEL.....	96
SEGMENTATION OPTIONS.....	97
ANALYSIS OPTIONS.....	97
EXPORT SETTINGS.....	98
Analysis Computation Explained.....	99
UNITS.....	99



SPEED.....	100
Stitching Images Through MCAM Viewer.....	101
Estimated File Size and Recording Duration for Video Acquisition.....	103
System Calibration.....	105
Programming Interface and API.....	105
Accessories.....	106
Stage Inserts.....	106
Multiwell Microplate Insert - I-3020.....	106
Other Recommended Inserts.....	106
Vireo.....	107
Graphical User Interface (GUI).....	107
Opening and Closing the MCAM.....	107
GUI Navigation.....	108
MCAM Settings.....	109
Save and Export Images.....	113
MCAM Data Analysis.....	115
Background Subtraction.....	115
Cell Counting.....	117
Channel Threshold.....	120
Confluence Analysis.....	123
Macroview.....	125
Region of Interest (ROI) Extraction.....	128
Segmentation.....	131
Stack Projection.....	133
Maximum and Minimum Projections.....	134
Mean projection.....	135
Sum Projection.....	135
Standard Deviation (STD) Projection.....	136
Laplacian Focus Selection.....	137
Focus Fusion.....	138
Content aware focus projection.....	139
Stack Reduction.....	141
Transfection Efficiency Analysis.....	142
Live/Dead Analysis.....	145
Fluorescent Nuclei Segmentation Analysis.....	148
Loading your data with other software.....	154
Python®.....	154
ImageJ.....	154
Matlab®.....	154
R.....	155
Loading Image Files and EXIF Orientation.....	155



TIFF Files.....	156
Detailed Technical Performance Limits.....	157
Appendices.....	158
Appendix A: Maximum Recording Time.....	158
Kestrel2850 -- Well plate Imaging.....	158
Kestrel2850 -- Free-swim Imaging.....	159
Appendix B: Dataset size for extended recordings.....	160
Appendix C: System Disassembly by End User.....	161
High-Speed Data Cable Removal.....	161
PCIe Ribbon Cable Installation - Kestrel (2022 Revision).....	162
Reflection Illumination Board Installation -- Falcon (Discontinued).....	163
Appendix D: Trigger connectivity.....	165
Output Trigger.....	165
APPLICATION NOTE - VIDEO.....	165
Output Trigger for usage with Animal Tracking.....	166
Known issues.....	167
Input trigger.....	167
Appendix E: Sensor Optical Characteristics.....	167
Sensor Core Characteristics - Single Sensor.....	167
High-speed Fruit Fly Configuration.....	169
Sensor Characterization Parameters.....	170
Sensor Spectral Characteristics.....	171
Appendix F: Frame rate control and limitations.....	172
Appendix G: Thermal Monitoring.....	173
Well Plate Water Temperature Table.....	173
Kestrel Chamber Temperature Table.....	174
Appendix H: Environmental Control.....	175
Appendix I: Pixel Unit Conversion between GUI and Python.....	177
Appendix J: Optical Specs of Objective Lenses: high-speed Fruit Fly.....	179
Appendix K: MCAM Configuration.....	180
Appendix L: Transmission Illumination Module.....	181
Appendix M: Fluorescence Illumination Spectra.....	183
Kestrel.....	183
UV LED - 385 nm Excitation.....	183
ROYAL BLUE - 440 nm Excitation.....	184
AMBER - 590 nm Excitation.....	185
Vireo.....	186
UV LED - 385 nm Excitation.....	186
ROYAL BLUE - 440 nm Excitation.....	187
Green - 530 nm Excitation.....	188
Red - 633 nm Excitation.....	189
Appendix N: Imaging Orientation.....	190
Other Questions.....	193



Troubleshooting Guide.....	194
Did Not Detect MCAM.....	194
Did not detect a subcomponent.....	195
Limited Recording Duration.....	196
Cold Boot.....	198
Connection issues.....	199
Ensuring a good connection to the Multi-Camera Array.....	200
Fluorescence Illumination Modules Turned Off or Not Turning On.....	201



Safety Warnings and Precautions

The MCAM system ("MCAM") should always be used in accordance with the guidelines of this manual to avoid risk of personal injury and/or damage to the instrument. **If the MCAM is used in a manner not consistent with the manner specified by Ramona Optics, Inc. in this manual, protection provided by the equipment may be impaired.**

Note the following safety warnings:



The MCAM is heavy (Kestrel: ~30 kg/68 lbs; Vireo: ~91 kg/200 lbs). Use caution when lifting. Use your legs and core muscles to lift the equipment, not your back. Keep the load close to your body during the lift. Maintain a firm grip on the load with both hands.



Given the weight of the MCAM, for stability, position all hardware on a stable surface prior to use.



Prior to moving, cleaning, or performing maintenance on the MCAM, always power off and disconnect power from hardware and disconnect the MCAM and MCAM Workstation.



The MCAM is an electronic system. Do not touch any MCAM hardware with wet hands.



Ventilation is required for proper function of the MCAM. Make sure to leave a minimum of two inches or 5 cm of space around all sides of the MCAM as well as the MCAM Workstation.



The MCAM's X, Y, and Z stages can move quickly, causing pinch hazards. To avoid pinch hazard, ensure hands are outside the MCAM imaging chamber before operating the moving MCAM stages using the provided software.



The LEDs on the MCAM Fluorescence Illumination Module can become very hot after use. Do NOT touch any part of MCAM Fluorescence Illumination Module except for Quick Release Handles (see MCAM User Manual).



The MCAM is an electronic device. Disconnect power prior to moving, cleaning, or performing maintenance.

MCAM Overview

The Multi-Camera Array Microscope (MCAM) is a form of Gigapixel Microscope. This new class of high-throughput imaging system enables Gigapixel Microscopy and consists of 6 main components: data acquisition electronics, image forming optics, an illumination unit, a motion control unit, the graphical user interface (GUI), and MCAM software (including machine learning analysis and the software development kit).

Ramona offers two products that leverage its MCAM technology, the Kestrel™ and the Vireo™:

- The Kestrel consists of 24 or 48x video microscopes and is the ultimate imaging system for small model organisms. It transforms small model organism research with high-throughput behavioral, morphological, and fluorescence analysis of entire plates at video rates.
- The Vireo consists of 24x video microscopes and is the world’s fastest live-cell imaging system. It revolutionizes 4D live-cell imaging, delivering unparalleled speed, precision, and multi-channel fluorescence for studying cells and organoids. It is compatible with laboratory scheduling software, enabling seamless integration into automated workflows.

The software and firmware coordinate these optical sensors so that they operate in the desired configuration. The illumination unit enables one to control both the spectral and the angular profile of the illumination. The GUI is provided to give users a quick way to navigate many of the functions of the MCAM. The software development kit (SDK) provides finer grained control over the MCAM's functionality, enabling more advanced acquisition through Python®.

Technical Specifications -- Kestrel

ENVIRONMENTAL CONDITIONS	Operating Temperature:	15 - 28 °C with ventilation
	Storage Temperature:	5 - 40 °C
	The temperature of the MCAM should not be below that of the dew point to avoid condensation on the electronics.	
SYSTEM SPECIFICATIONS	MCAM Dimensions:	360 x 360 x 460 mm (Depth x Width x Height)
	Mass (weight):	30 kg (68 lbs)
	Electrical outlets:	Up to three (3) outlets for the MCAM, Computer, and Monitor Display.

COMPUTER DIMENSIONS
AND SPECIFICATIONS

Option 1:	Desktop Computer
Dimensions:	500 x 395 x 525 mm (Depth x Width x Height)
Mass (weight):	14 kg (30 lbs)
Shipping weight:	32 kg (70 lbs)
Nominal power consumption:	400 W
Maximum power consumption:	850 W
Nominal power draw during usage:	300 W to 500 W
Option 2:	Workstation Laptop
Mass (weight):	3 kg (6.65 lbs)
Dimensions:	300 x 400 x 30 mm (Depth x Width x Height) 17 Inch Monitor diagonal
Nominal power consumption:	100 W
Maximum power consumption:	300 W
Nominal power draw during usage:	150 W
Option 3:	Network attached computer
Dimensions:	Integrated into MCAM
Nominal power consumption:	40 W
Maximum power consumption:	100 W

TRANSMISSION
ILLUMINATION
WAVELENGTHS

Nominal power draw during usage:	80 W
Visible:	Red -- 620 nm, Green -- 518 nm, Blue -- 470 nm
Infrared:	850 nm

Technical Specifications -- Vireo

ENVIRONMENTAL
CONDITIONS

Operating Temperature:	15 - 28 °C with ventilation
Storage Temperature:	5 - 40 °C
	The temperature of the MCAM should not be below that of the dew point to avoid condensation on the electronics.

SYSTEM
SPECIFICATIONS

MCAM Dimensions:	529 x 469 x 600 mm (Depth x Width x Height)
	Note: The sample tray will extend 120 mm from the front of the case.
Mass (weight):	91 kg (200 lbs)
Electrical outlets:	Up to three (3) outlets for the MCAM, Computer, and Monitor Display.

COMPUTER DIMENSIONS
AND SPECIFICATIONS

Desktop Computer	
Dimensions:	427 x 210 x 474 mm (Depth x Width x Height)
Mass (weight):	18 kg (40 lbs)
Shipping weight:	32 kg (70 lbs)
Nominal power consumption:	300 W
Maximum power consumption:	850 W

TRANSMISSION
ILLUMINATION
WAVELENGTHS
(BRIGHTFIELD)

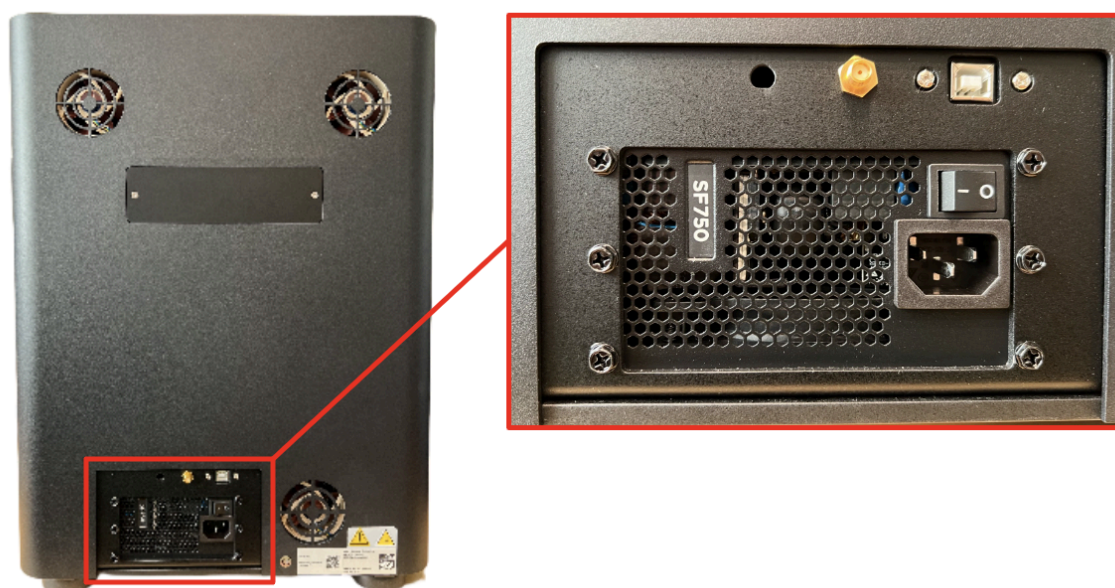
TRANSMISSION
ILLUMINATION
WAVELENGTHS
(FLUORESCENCE)

Nominal power draw during usage:	400 W to 800 W
Visible:	Green – 530nm
Visible:	Red -- 640 nm, Green -- 530 nm, Blue -- 440 nm
UV:	385 nm

Equipment Ratings

The MCAM is rated for use in the following conditions:

- Indoor use
- Altitude up to 2000 m
- Operating temperature 15-28 °C
- Storage Temperature 5 - 40 °C
- Maximum relative humidity 80% for temperatures up to 31 °C decreasing linearly to 50% relative humidity at 40 °C
- MAINS supply voltage fluctuations up to +/-10% of the nominal voltage
- Short duration overvoltages occurring on the MAINS supply
- Only non-conductive pollution (solid, liquid, or gaseous foreign matter that may produce a reduction of dielectric strength or surface resistivity), except that occasionally a temporary conductivity caused by condensation is expected



Kestrel (2022 Revision) Features

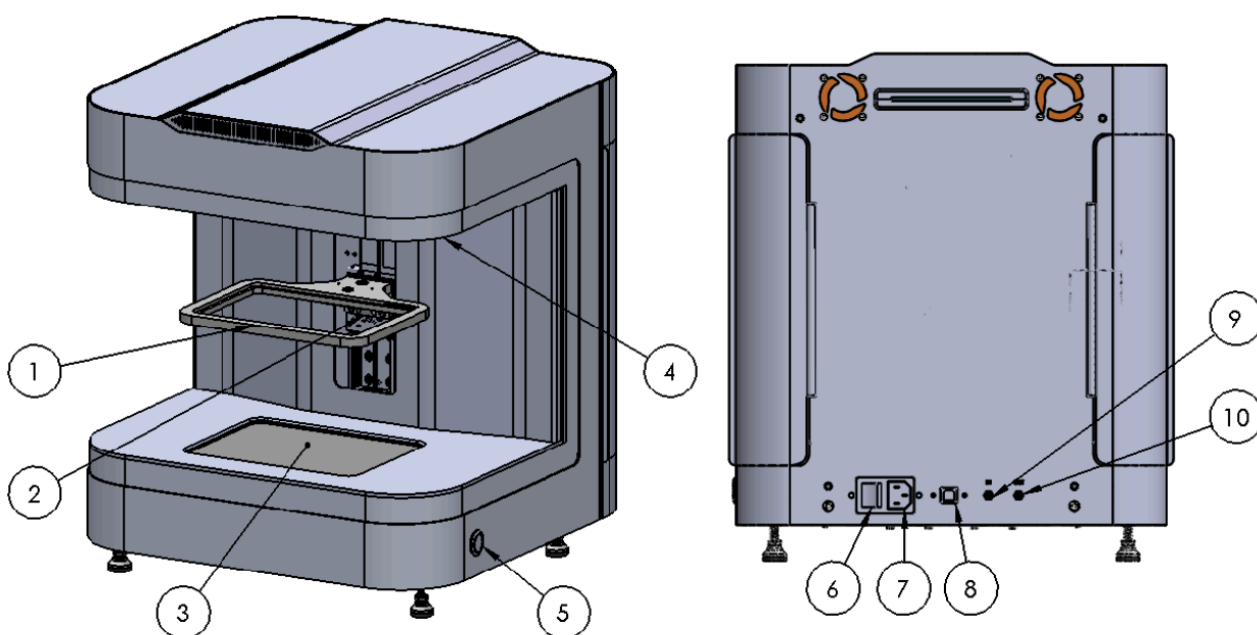


Fig 1: MCAM Isometric View (Left) and Rear View (Right) with key features highlighted.

- | | |
|----|----------------------------------|
| 1. | Stage |
| 2. | Z-axis motor |
| 3. | Transmission Illumination Source |
| 4. | Reflection Illumination Source |
| 5. | Power Indicator Light |

- | | |
|-----|------------------|
| 6. | Power Switch |
| 7. | Power Socket |
| 8. | USB-B Port |
| 9. | Trigger In Port |
| 10. | Trigger Out Port |

Kestrel (2024 Revision) Features

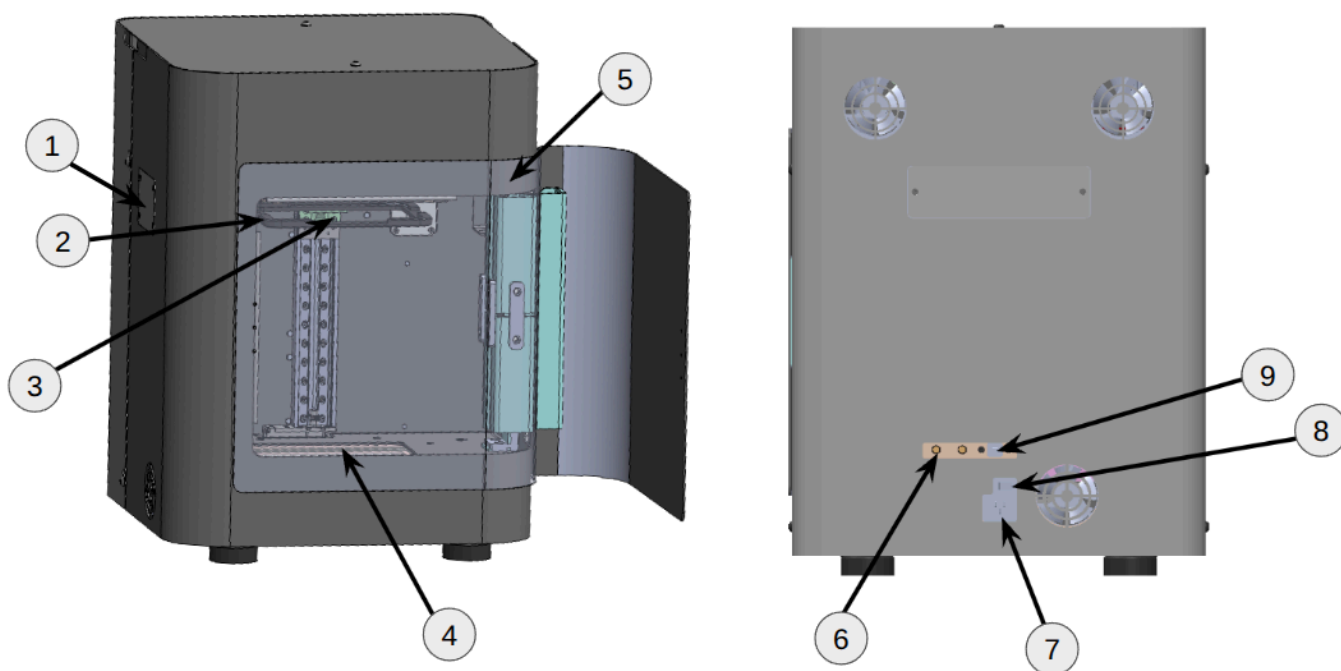


Fig 2: Kestrel Isometric View (Left) and Rear View (Right) with key features highlighted.

1. Cable passthrough for user provided stimuli
2. Stage
3. Z-axis motor
4. Transmission Illumination Source
5. Reflection Illumination Source

6. Trigger Out Port
7. Power Socket
8. Power Switch
9. USB-B Port



Vireo Features

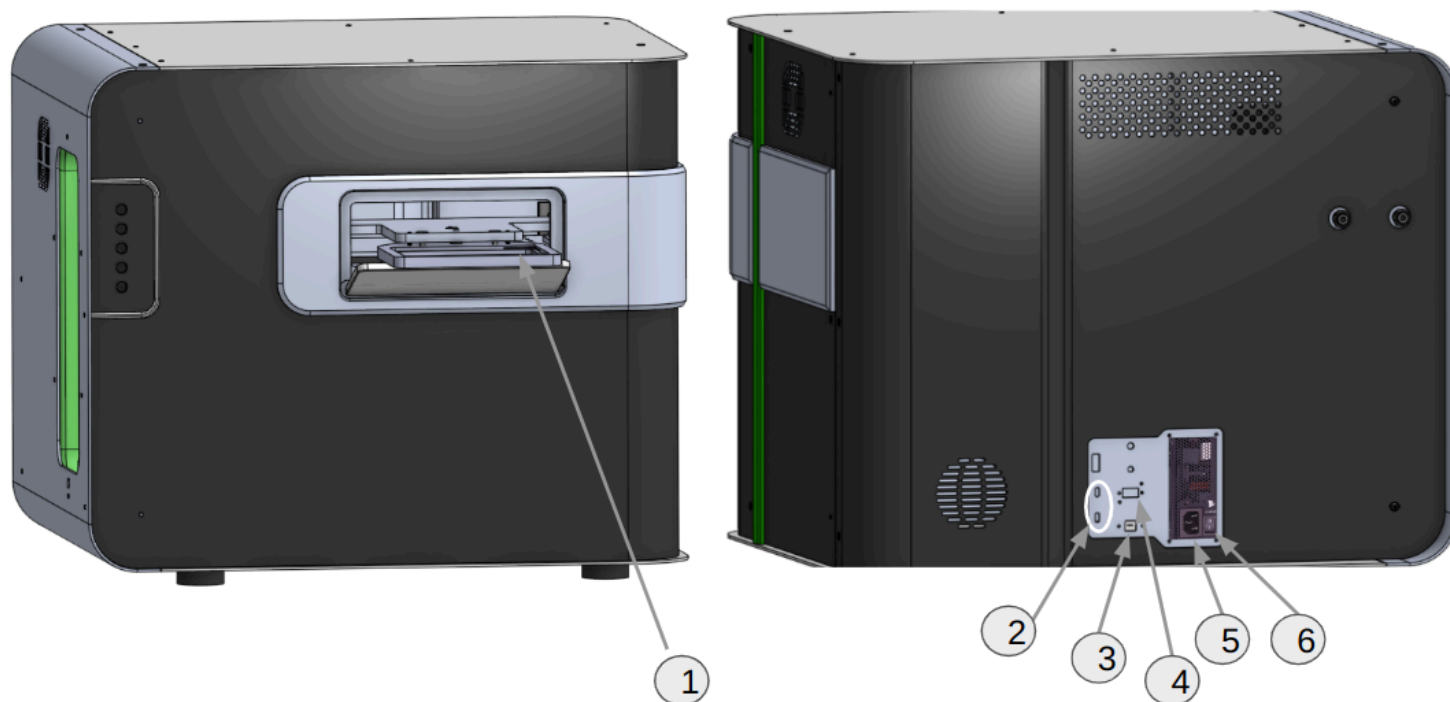


Fig 3: Vireo™ Isometric View (Left) and Rear View (Right) with key features highlighted.

1. Plate Nest
2. Thunderbolt Port
3. USB-B Port
4. RS232 Port
5. Power Socket
6. Power Switch

Getting Started

Recommended Hardware

The following hardware items are included with the MCAM:

ITEM	QTY
Multi-Camera Array Microscope (MCAM)	1
MCAM Workstation Note: Linux based workstation to ensure optimal system functionality. For more information on system specifications, please refer to Appendix A.	1
Computer Monitor, either: - Dell 24 Monitor - P2422H - Full HD 1080p, IPS Technology, ComfortView Plus Technology (see on Amazon), OR - Dell 27 Monitor - P2722H - Full HD 1080p, IPS Technology, 8 ms Response Time (see on Amazon)	1
Mouse	1
Keyboard	1
Display Cable (HDMI or DisplayPort)	1
Universal Power Cable (3-prong NEMA, rated for 100-240VAC, 10A-5A, 47-63 Hz)	3
Note: Universal Power Cables (3-prong NEMA, rated for 100-240VAC, 10A-5A, 47-63 Hz) are rated for MCAM and MCAM Workstation power supplies. Do not connect other cable types to the MCAM.	
Note: To ensure proper grounding, only Universal Power Cables (3-prong NEMA, rated for 100-240VAC, 10A-5A, 47-63 Hz) can be used with the MCAM. Do not use other cable types with the MCAM.	
For Kestrel: - USB Cable (USB-A to USB-B) - Mini SAS Cables - Transmission Illumination Diffuser	1 1 1



• Universal Stage Insert	1
• Tokai Hit ThermoPlate TPI--SQX Insert (Optional)	1
For Vireo:	
• USB Cable (USB-B to USB-C)	1
• Thunderbolt Cable	1
• Universal Stage Insert	1
• Fluorescence Illumination module and emission filter tray (Optional)	1
• Okolab Stage-top Incubator (Optional)	1

The following software is included with the MCAM:

ITEM	DESCRIPTION
MCAM User Interface	User interface for controlling the MCAM Unit and capturing data.
MCAM Viewer	Viewing software for data captured on the MCAM. For optimal performance, it is recommended the MCAM Viewer be run on a computer with the following specs: • Ubuntu 24.04 or Windows 10 • A modern Intel i7, Xeon processor, or Ryzen (from the last 3 years) • 16 GB of RAM (64 GB recommended) • An internal solid state drive for data storage • A dedicated graphics card Using an external hard drive can significantly reduce data transfer rates and system performance.
Software Package License	Ramona Optics Software Authentication Token

MCAM software is continuously updated. For information on changes and update instructions, please refer to the changelog: <https://docs.ramonaoptics.com/changelog.html>



Technical Support

For technical and/or service issues please contact Ramona Optics, Inc., at help@ramonaoptics.com.

Please feel free to contact the Ramona team to ask any other questions not resolved by this manual or to request additional material regarding the MCAM. We can be reached by email at info@ramonaoptics.com and by phone at +1 (919) 797-9975.

Website: www.ramonaoptics.com

Address: 302 E Pettigrew Street
Durham, NC 27701



Equipment Maintenance and Service

Servicing the MCAM

For the safety of MCAM hardware, only Ramona service technicians should service the MCAM. Users should not attempt to service their MCAMs.

To avoid risk of shock or burns, as well as damage to the equipment, Ramona service technicians should confirm that both the MCAM and the MCAM workstation are powered off before servicing.

To verify the safety state of the equipment after servicing, Ramona service technicians should successfully complete functionality testing, exercising MCAM components as a user would. Functionality testing will include the exercising of any system stages and illumination components, and capturing video and image data in the formats commonly utilized by users.

Cleaning and Decontamination

Instructions for surface cleaning and decontamination (can be performed by any users of the MCAM) are as follows.

STEP	ACTION
1	Confirm that the power button at the back of the unit is in the OFF position (0).
2	Confirm that the MCAM Workstation is powered off.
3	Prepare a fresh 70% ethanol disinfectant solution and pour it into a spray bottle.
4	Spray all exterior surfaces of MCAM with ethanol disinfectant solution. Spray any handles on MCAM with ethanol disinfectant solution.
5	Let disinfectant sit on sprayed surfaces for 20 minutes. Surfaces should appear wet for the full 20 minutes. Re-wet surfaces that begin to dry before the end of this time period.
6	After waiting for 20 minutes, use paper towels to wipe down all sprayed surfaces and discard towels as biohazard waste.

Cleaning and decontamination of the MCAM interior should only be performed by a Ramona technician. If equipment requires interior cleaning or decontamination (for example, if there appears to be dust on optical lenses), please contact **technical support** at Ramona Optics: help@ramonaoptics.com



Installation

A Ramona technician will often perform the MCAM installation, but others can do so following the below instructions and contact the Ramona team if any questions arise.

The safety of any system incorporating an MCAM is the responsibility of the assembler of that system.

STEP	ACTION
1	Position the MCAM, MCAM Workstation, Computer Monitor, Keyboard, and Mouse on a stable surface with minimal vibration. Do not position MCAM components in a manner that makes power buttons difficult to access.
2	Confirm that the power button at the back of the unit is in the OFF position (0).
3	Confirm that the MCAM Workstation is powered off.
4	Plug the MCAM, MCAM Workstation, and Computer Monitor into standard 120 V/220 V wall power outlets using the Universal Power Cables (3-prong NEMA, rated for 100-240VAC, 10A-5A, 47-63 Hz). Note: Universal Power Cables (3-prong NEMA, rated for 100-240 VAC, 10A-5A, 47-63 Hz) are rated for MCAM and MCAM Workstation power supplies. Do not connect other cable types to the MCAM. Note: To ensure proper grounding, only Universal Power Cables (3-prong NEMA, rated for 100-240 VAC, 10A-5A, 47-63 Hz) can be used with the MCAM. Do not use other cable types with the MCAM.
5	Connect MCAM Workstation and Computer Monitor with Display Cable (HDMI or DisplayPort).
6	For the Kestrel: I. Connect MCAM and MCAM Workstation with one (1) USB Cable (USB-A to USB-B). II. Connect MCAM and MCAM Workstation with Mini SAS Cables. For the Vireo: I. Connect MCAM and MCAM station with a Thunderbolt cable.



II. Plug in the USB-B-to-USB-C cable to the USB-B port and the Thunderbolt port at the back of the MCAM.	
7	Plug in the dongle to connect keyboard and mouse to the computer. Plug in the USB-to-USB-C cable for keyboard/mouse recharge.
8	

Boot Procedure

Once the system has been assembled, start the MCAM in the following order:

In the following order:

Power on the monitor.

Power on the MCAM unit.

Power on the computer.

a. The default username and password are:

i. Username: **ramona**

ii. Password: **gigapixel**

Note: The MCAM must be powered on before the workstation computer to open and use the MCAM software. If the computer is on and MCAM is off, please turn on the MCAM and then reboot the computer prior to use. If the system is not powered on in this order, the MCAM will return an error message upon opening and will not function.

Once the system has been started for the first time, it is possible to turn off the MCAM while leaving the computer on. It will simply have to be turned on again prior to using the MCAM software.



Kestrel Reflection Illumination Module (Optional)

The Kestrel Reflection Illumination Module allows the user to illuminate samples from above. To set up the Kestrel Reflection Illumination Module within the MCAM, complete the following steps.

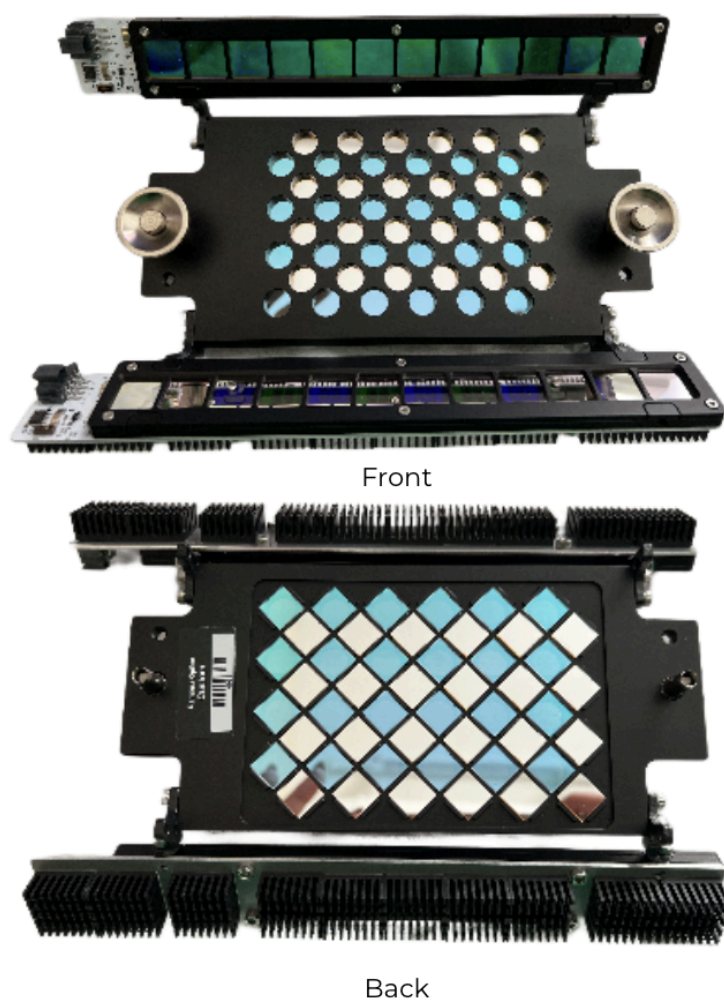


Fig 4: Kestrel Reflection Illumination Module



Fig 5: MCAM chamber upper interior surface with lens block

STEP	ACTION
------	--------

- | | |
|---|---|
| 1 | Remove the Kestrel Reflection Illumination Module from its packaging, taking care to only touch it by the quick release handles on either end of the module. |
|---|---|

Note: To protect the hardware of the **Kestrel Reflection Illumination Module**, never touch any part of the module except the quick release handles.

Note: Surfaces of **Kestrel Reflection Illumination Module** can become hot after use. To avoid risk of burns, use caution when touching module after use. Do not touch any part of module except for quick release handles.

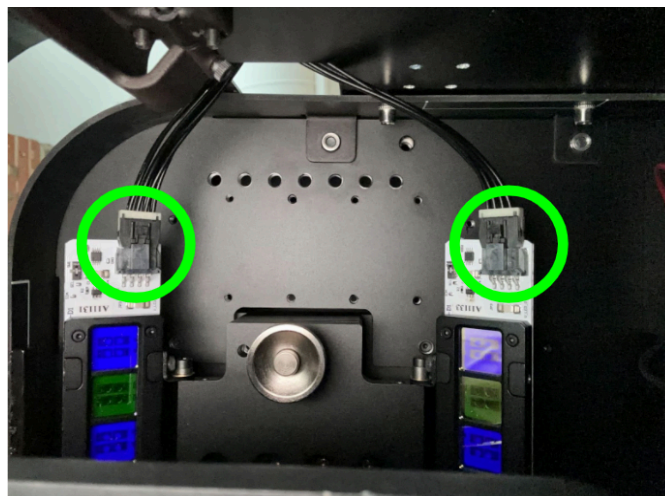


- 2 Simultaneously pressing down on the button on each of the **Kestrel Reflection Illumination Module's** 2 quick release handles, insert the module's pins into the sockets on either side of the MCAM lens block on the upper interior surface of the MCAM chamber.

Make sure to insert the **Kestrel Reflection Illumination Module** so the end with the cabling is near the **Kestrel Reflection Illumination Module Ports** on the right hand side of the MCAM

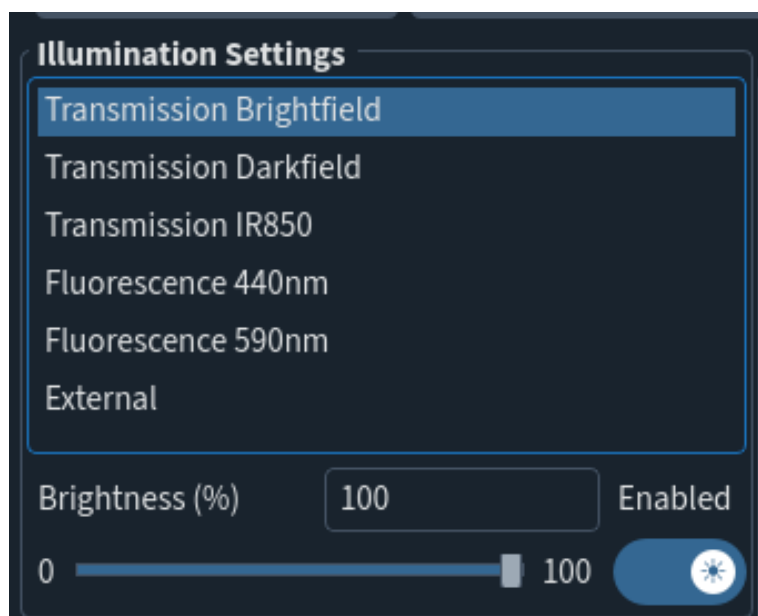


- 3 Releasing the buttons on the **Kestrel Reflection Illumination Module** quick release handles, connect the cable on the module's back rail to the upper **Kestrel Reflection Illumination Module Port**. Connect the cable on the module's front rail to the lower Kestrel Reflection Illumination Module Port.





- 4 Now, plug the USB cable from the control box to the workstation and connect the provided AC power cable between the control box and an AC power outlet. In the MCAM GUI software, Fluorescence is located in the left panel with the rest of the acquisition controls. Select the excitation wavelength to use and type or select the power with the slider. Use the enable slider to turn the LEDs on or off.



Driving the LEDs for extended periods of time and at high power levels could cause the LEDs to overheat. This will result in the LEDs turning off automatically. It is recommended that the Fluorescence illuminations be turned off when not in use and, when in use, the power be kept below 50% whenever possible to reduce the likelihood of overheating. (See [Fluorescence Illumination Modules Turned Off or Not Turning On](#) Troubleshooting section for more details).

If the illumination modes from the Reflection modules don't appear, refer to the troubleshooting guide "[Did not Detect Subcomponent](#)" section

The Kestrel Reflection Illumination Modules may be used as reflection illumination modules if they are equipped with white or IR 850 nm LEDs.



Viewing Data Acquired by the MCAM

The images acquired by the MCAM can be viewed in a variety of ways:

PER-MICRO CAMERA IMAGES

Data can be exported on a per-micro camera level allowing them to be viewed and analyzed in isolation. A metadata file includes information about the MCAM that was used to acquire the images in a machine readable format.

STITCHED IMAGE OF THE ENTIRE FIELD-OF-VIEW (FOV)

Data can be exported using the GUI into a single stitched image. This is most appropriate for publication quality images. Unfortunately, many image viewers struggle to load images more than 100 megapixels in size, limiting the use of this image format.

NC OF THE ORIGINAL DATA

The standard NC files that are used throughout the MCAM software retain the original data. This data can be loaded into other applications such as Python and Matlab, but requires users to apply many of the corrections applied in the MCAM Viewer application. For more information about the dataset, please refer to our documentation found online: https://docs.ramonaoptics.com/python_metadata.hl

MCAM Viewer Installer

All required software has been installed on the provided MCAM workstation prior to delivery.

Kestrel

Graphical User Interface (GUI)

Opening and Closing the MCAM

After installing the acquisition software, and turning on the MCAM, the Graphical User Interface (GUI) can be started by clicking on the Ramona Optics application launcher on the sidebar. By default, the MCAM will attempt to connect to each of three components:

- The reflection illumination board,
- The transmission illumination board,
- The stage motor.

If any of these components are unavailable, the user can choose to not connect these components by deselecting their boxes in "Select Components" under the "Advanced" menu on the toolbar. This will automatically disable the ability to change the settings for these components.

The MCAM can be closed using the "Exit" button from the File menu, or by exiting the GUI with the red "X" button in the upper right corner. Either method will stop all ongoing operations and safely close the MCAM.

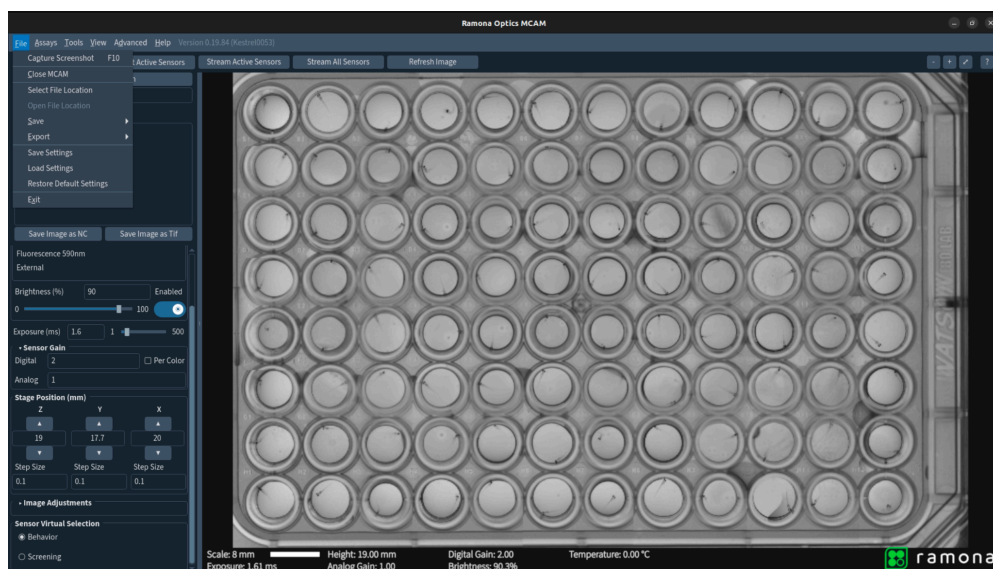


Fig 6: GUI start screen, with the "File" menu open and highlighted.

GUI Navigation

The user can zoom and pan in the GUI to view any section of the streamed or acquired images. To zoom, either scroll up and down on the mouse wheel, or move the mouse while holding the right click mouse button. To pan, move the mouse in any direction while holding down the left click button.

At the top left of the window (Fig 7, #1) the current file location is displayed. This is the location where all files will be saved.

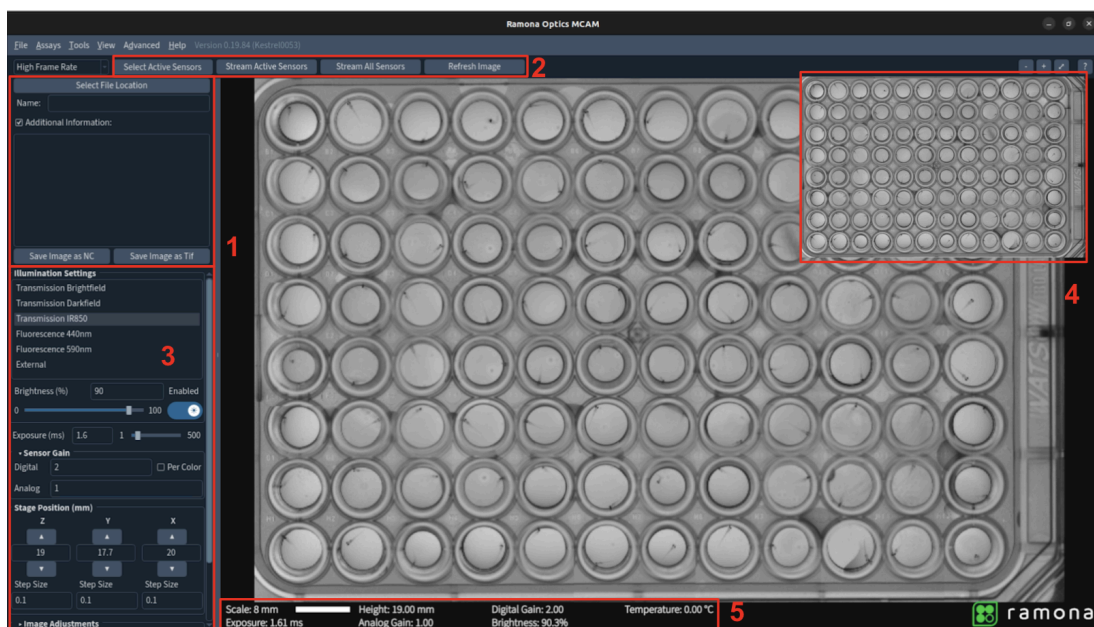


Fig 7: GUI layout with 1) Selected file location, 2) Image Acquisition Mode Buttons, 3) Image Acquisition Settings Panel, 4) Picture-in-picture and 5) Key Image Specifications highlighted.

At the top left of the window (Fig 7, #2) the Image Acquisition Mode Buttons allow the user to select which area under the microscope they want to image. The following are the functionalities provided:

- Stream All Sensors: allows users to see everything happening under the microscope.



- **Select Active Sensors:** allows users to select certain camera sensors to record from a specific area of interest. You can either select just one sensor by simply moving the cursor around, the display will be bright around the area covered by that one sensor and have shadows on the remaining area, or you can select several sensors. For selecting several sensors, the first click of the user cursor will define the beginning of the area that will be recorded, a selection box will pop up which can be expanded/reduced by moving the cursor and finally the second click will define the end of the area (Fig 8).

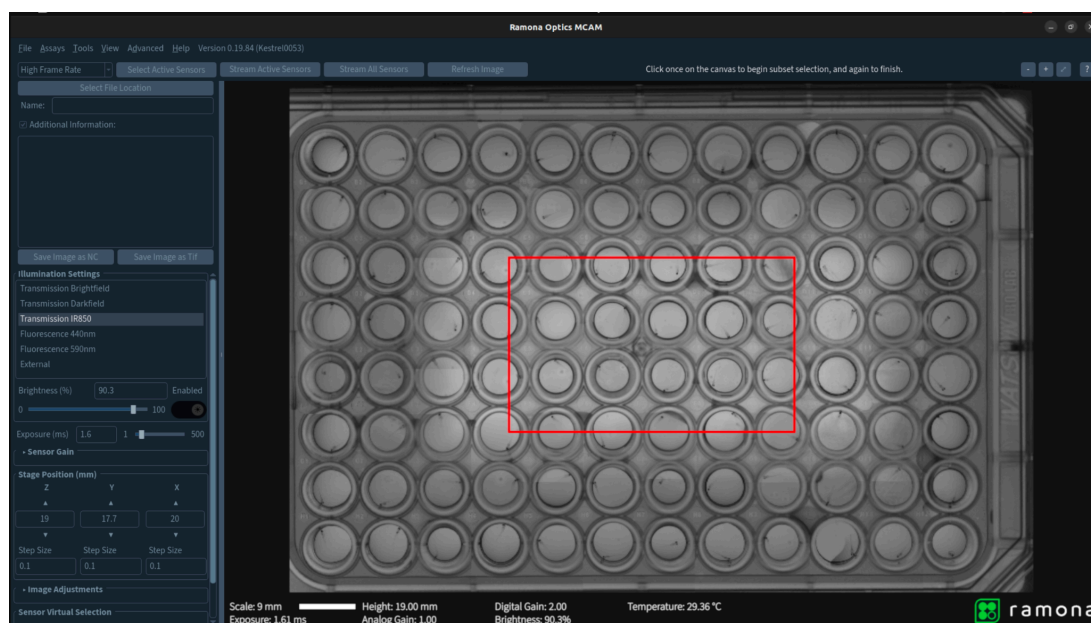


Fig 8: Streaming from certain selected sensors.

- **Stream Active Sensors:** allows users to see everything happening under the area they selected using the Select Active Sensors function.

The image acquisition settings panel is on the left side of the viewing window (Fig 7,, #3).

While panning, the picture-in-picture (Fig 7, #4) in the right-hand corner of the screen shows what portion of the full image is currently displayed.



Fig 9: Key Image Specifications



Some key image parameters are displayed at the bottom of the viewing window (Fig 7, #5 and Fig 9) including a scale bar providing an approximate scale. Note that the scale bar assumes a constant width for each pixel, and may not be accurate across the entire sample if 3D features are present. Additionally on the right side, RGB color values of the LEDs are displayed which have been preselected using calibration data.



MCAM Settings

Most of the MCAM Settings can be adjusted using the Settings Panel on the left side of the GUI.

The panel can be hidden to make more room on the screen by clicking on the white line border between the settings panel and the image, and sliding to the left. The GUI allows the user to set the system parameters within a range of values acceptable for most common experimental setups. More customized settings can be set in the Advanced submenu.

Most settings values can be adjusted either by using the sliders or by directly typing a value into the text box. When using the text box, the accepted values will still be limited to the maximum and minimum values shown on the respective slider. In addition, the setting displayed in the text box may adjust to a value slightly higher or lower than the value entered. This is because for certain settings (particularly digital and analog gain), the MCAM can only accept distinct values. The GUI will automatically round the entered setting to the nearest acceptable value, and send that parameter to the MCAM.

LOADING/SAVING SETTINGS

On first use, the GUI will load a set of predetermined default settings which were chosen to produce generally high-quality images. Afterwards, the GUI will automatically load the settings from its last use. Because each unique imaging environment and sample may have different optimized settings, the settings for individual experiments can be saved and loaded using the "File" menu in the toolbar, selecting "Save Settings" within it. The default settings can also be re-loaded using the "Restore Default Settings" button in the same menu.

IMAGE ACQUISITION SETTINGS

The sensor exposure, gains, and LED parameters can all be set from the setting panel on the left side of the GUI, as well as contrast levels of the displayed image. These settings are further explained below.

EXPOSURE

This setting controls the amount of time that the camera sensor acquires light for each frame. A longer exposure will yield a brighter image with more motion artifacts while shortening the exposure will darken the image with fewer motion artifacts. Additionally this setting will affect the frame rate when using streaming acquisition modes with longer exposures decreasing potential frame rate. In order to maximize frame rate, consider keeping the exposure as low as possible and balance lighting with the brightness setting.

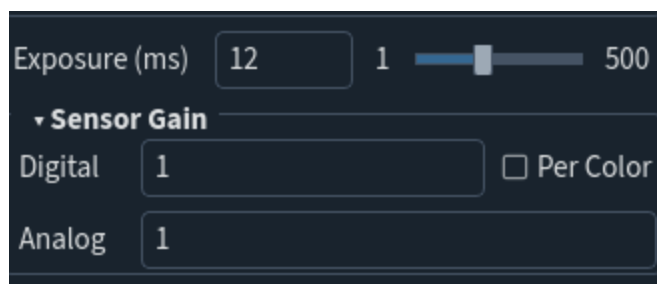


Fig 10: Camera sensor settings.

DIGITAL AND ANALOG GAIN

Gain settings will magnify the signal produced by the image sensors. Analog gain increases the sensitivity of the sensors using hardware while digital gain increases the signal using software to multiply the signal once it has been converted to a digital signal. Increasing gain values will increase image intensity but will also increase noise proportionally. Keeping gain values as low as possible will reduce inherent noise (Fig 10).

BRIGHTNESS

Located under "LED Settings", this will control the overall brightness of the LEDs used to illuminate the sample (Fig 11).

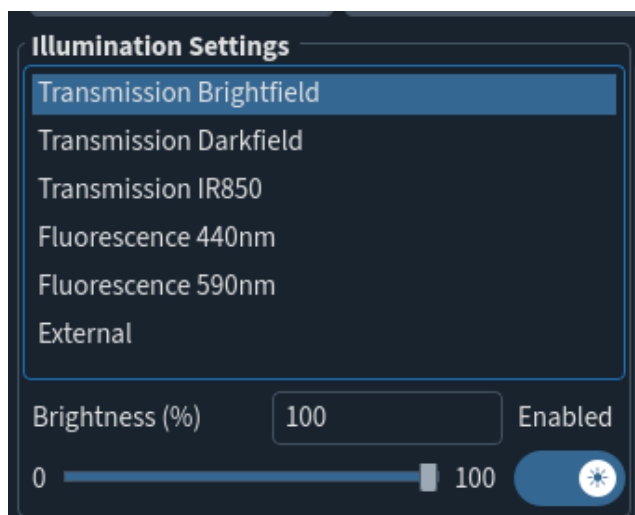


Fig 11: GUI LED board settings

Zebrafish Panel Settings

VIBRATION STIMULUS

Both the frequency and amplitude of the vibration stimulus can be controlled using the MCAM user interface. The desired stimulus frequency can be selected from the dropdown menu located under "Assays">"Zebrafish". As each stimulus frequency is paired with a specific stimulus amplitude (documented in the lookup table below), varying stimulus frequencies will also vary stimulus amplitude. Use the lookup table below to control the amplitude of the vibration stimulus.

VIBRATION STIMULUS FREQUENCY-AMPLITUDE LOOKUP TABLE FOR ZABER STAGE X-LSM200A	
FREQUENCY (Hz)	AMPLITUDE (μm)
250	1
300	0.25
300	0.5-6
300	0.75
300	1
350	1

ILLUMINATION MODE SELECTION

- The transmission LED board: the standard light source for use with the MCAM when imaging semi-transparent samples and is installed in the base of the unit. Selecting this light source, the stage will be lit from below and pass through the sample before reaching the camera sensors. The illumination options that are linked to this board are the transmission brightfield, darkfield and IR850 modes.
- The Kestrel Reflection Illumination Module is optional and can be mounted above the stage. Selecting this option will light the stage from above and light will reflect off the sample before reaching the camera sensors. The illumination options displayed will change depending on the module purchased with the system.

STAGE POSITION SETTINGS

The location of the Z-stage can be changed using either the up and down arrows or by manually entering a location (given in millimeters from the highest point where the stage can be positioned) (Fig 12) on the Z scale within the "Stage Position Panel". Step size controls the increment by which the arrows will adjust the height. Adjusting the Z-stage location will move the stage and sample in and out of focus. X and Y position can also be adjusted in the same manner.



Fig 12: Z-stage settings.

GAMMA

Adjusting this value will control the encoded image's sensitivity to bright and dark tones and is an effective method of controlling the overall image contrast as perceived by humans (Fig 13). Reducing Gamma will decrease shadows and darker tones yielding a brighter image.



Fig 13: GUI contrast settings.

MAX AND MIN LEVELS

The “Max level” and “Min Level” sliders are used to set the contrast of the image as displayed in the GUI (Fig 13). All pixels in the raw image data with values equal or greater to the selected “Max Level” will be displayed using the highest pixel value, while any data pixels with values below the “Min Level” will show up as black. The remaining middle range of pixels will be displayed using the full range of display pixel values. These settings are only for immediate display, and do not impact the data being gathered or saved. Fig 14 shows the impact on the displayed image of changing the Max and Min levels.

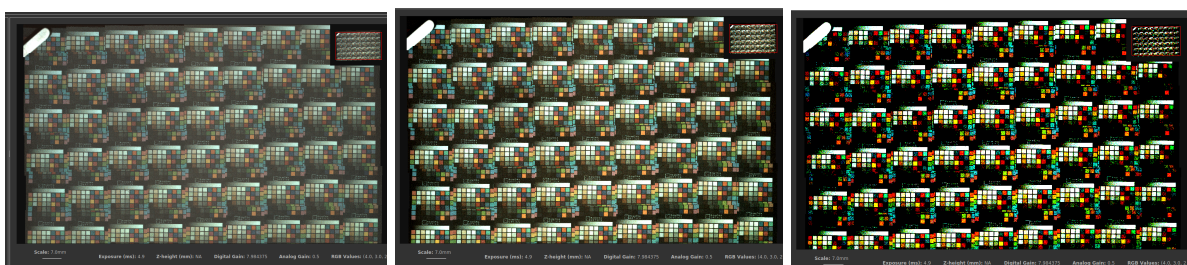


Fig 14: Images with different Max and Min levels: 0 and 1 (left), 0.7 and 0.2 (center), 0.5 and 0.5 (right).

Save and Export Images

MCAM data can be exported or saved from the GUI in several ways. First, high resolution images can be exported using the “Save Image as NC” or the “Save Image as Tif” buttons (Fig 15). “Save Image as NC” will save the full resolution MCAM data in a single “.nc” file which can be opened for analysis in the Ramona Optics custom MCAM Image Viewer or in MathWorks software. “Select File Location” will open the directory in which the most recently saved images were placed.

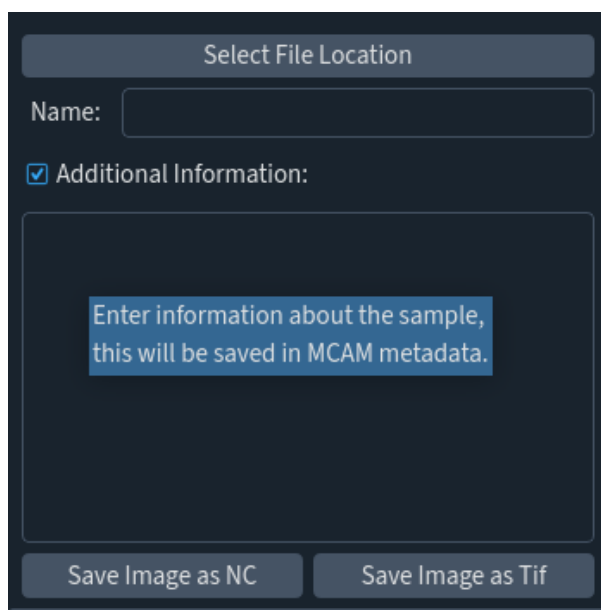


Fig 15: Save settings.

“.nc” files use the netcdf4 data format. More information on the implementation and usage of this file structure can be found here: <https://unidata.github.io/netcdf4-python/>

Saved images are given a filename according to the following rules:

1. The user defines the prefix of this filename by entering in the “Image Name” field (Fig 15).
2. The suffix of this filename is generated as a timestamp with the following format

YYYYMMDD_HHMMSS_mmm where:

- YYYY is the year when the image is saved, e.g. 2020.
- MM is the month when the image is saved, e.g. 08.
- DD is the day when the image is saved, e.g. 06.
- HH is the hour (in 24 hour format) when the image is saved, e.g. 15.
- MM contains the minute when the image is saved, e.g. 43.
- SS contains the seconds when the image is saved, e.g. 52.
- mmm contains the milliseconds when the image is saved, e.g. 549.
- The final timestamp will appear as: 20200806_154352_549

Notes can be entered in the “Additional Information” field which are saved with the image file and displayed in the MCAM Viewer with the image.

Low resolution screenshots of the currently displayed images can be saved using “Capture Screen-shot” in the “File” menu of the toolbar. This screenshot will include the main canvas, as well as the picture-in-picture and the main settings (see Fig 16).

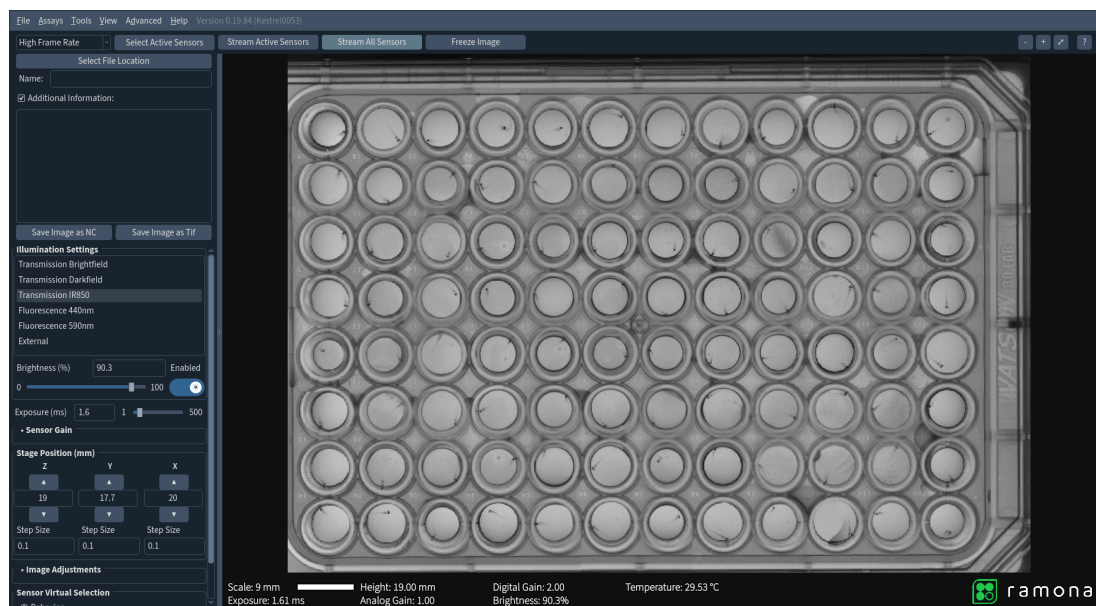


Fig 16: Screenshot of GUI Images

Note: If a filename is entered in the "Image name" field and the workspace is changed, the filename will be deleted and the user will need to enter a new filename.



Save Video or Z-Stack

To take and save a video or Z-stack, open the Assays window by selecting the “Assays” menu. Select either “Acquire Video” or “Acquire Z-stack” within it. A panel on the right side of the screen will pop up (Fig 17, Fig 18) and the user can start by entering a filename and any pertinent notes in the “Additional Information”.

When acquiring videos, the number of frames corresponds to the selected exposure time in the main window. For example, if the exposure time is set to 200 ms, a 10 frame video will be 2 s long (Fig 17). When acquiring Z-stacks the step size as well as bounds can be manipulated as shown in Fig 18.

Fig 17: Advanced Functions Window with Video settings.



Acquire Z Stack

Assay Protocol

LoadSave

Well Alignment

Select Alignment

Z Stage

Start height (mm)1.5Current height

Mid height (mm)2.05

End height (mm)2.6Current height

Step size (mm)0.1

Number of steps11

Options

ProjectionNone

Save ModeExport Images as Tif

Start Delay (seconds):0

☒ Include Timestamp

Select File Location

Name: _stack

Additional Information

Acquire Z Stack

Fig 18: Advanced Functions Window with Z-stack settings.



Annotation Tools

Within the MCAM GUI and Viewer, image annotation tools are provided to assist with image analysis. These tools can be accessed from the "Tools" menu in the toolbar (Fig 19).

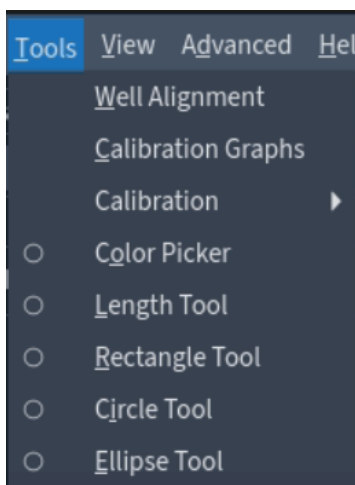


Fig 19: Tools menu

Color Picker

When the Color Picker tool is selected, clicking on the screen will display the pixel color at this location. Values are displayed as (Red, Green, Blue) integers corresponding to the three color channels of the image (Fig 20).

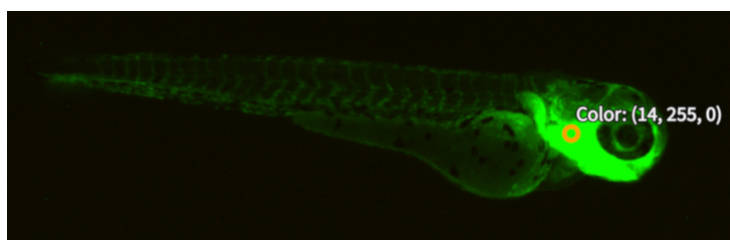


Fig 20: Example of color picker function on zebrafish image.



It is important to note that pixel values displayed on the screen are subject to color corrections and the viewing color space. Under the “View” menu in the toolbar, the color space can be adjusted which will alter the displayed image. It is important to note that pixel values displayed on the screen are subject to color corrections and the viewing values (Fig 20). In the below example, the color space has been changed from RGB to Grayscale and the RGB values all display the same color channel (Fig 21).

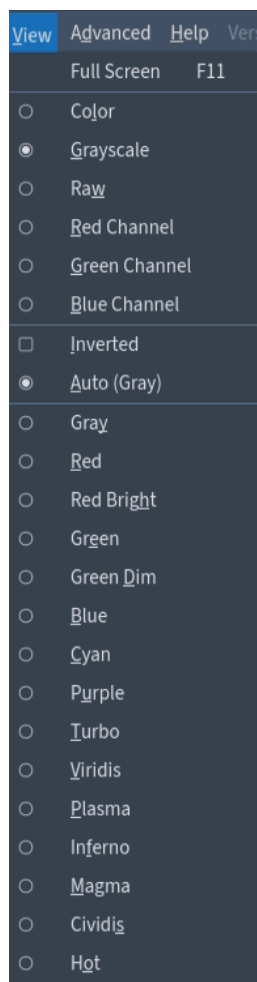


Fig 21: View menu

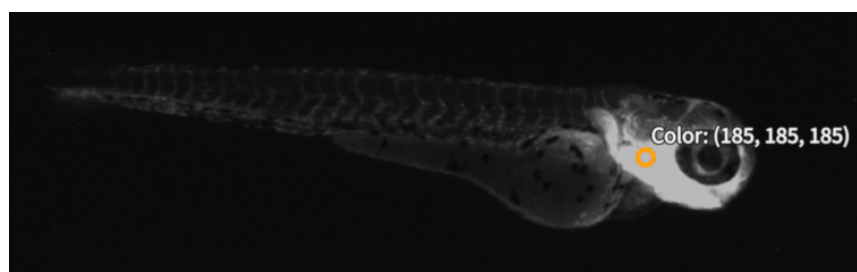


Fig 22: Example of View menu application to image referenced in Fig. 21



Length Tool

When the Length Tool is selected, straight lines can be measured on the screen (Fig 23). Clicking once with the left mouse button and once with the right mouse button places two points on the screen and a straight line is drawn between them. The length of the line is displayed on the screen. This computation requires an accurate pixel width calibration to convert from units of pixels to millimeters.

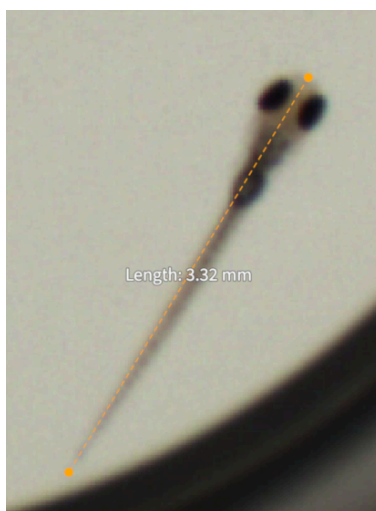


Fig 23: Length tool



Advanced Settings

Additional settings can be found in the “Advanced” menu and are described below (Fig 24).

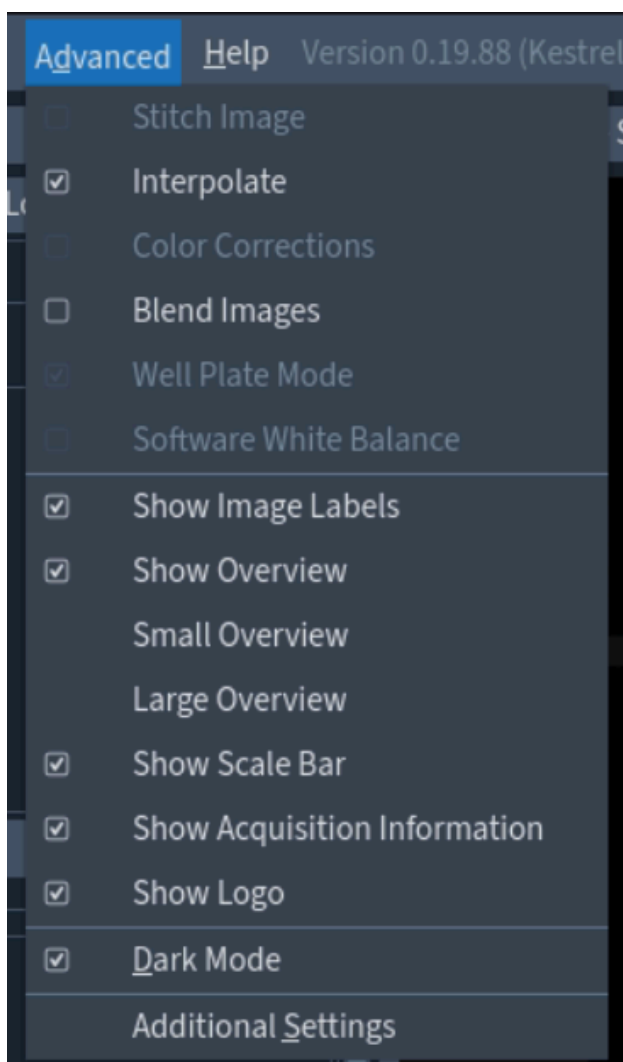


Fig 24: Advanced Menu

STITCH IMAGE

The individual images can be displayed in the GUI as stitched or unstitched (Fig 25). The full image is stitched using a calibration file specific to the individual MCAM setup (see [System Calibration](#)), which the GUI automatically loads and uses to place the images in the GUI canvas. If the image is un-stitched, the central square



from each imaging sensor will be displayed with a small border between them. Whether the image is stitched or unstitched makes no impact on the GUI performance.

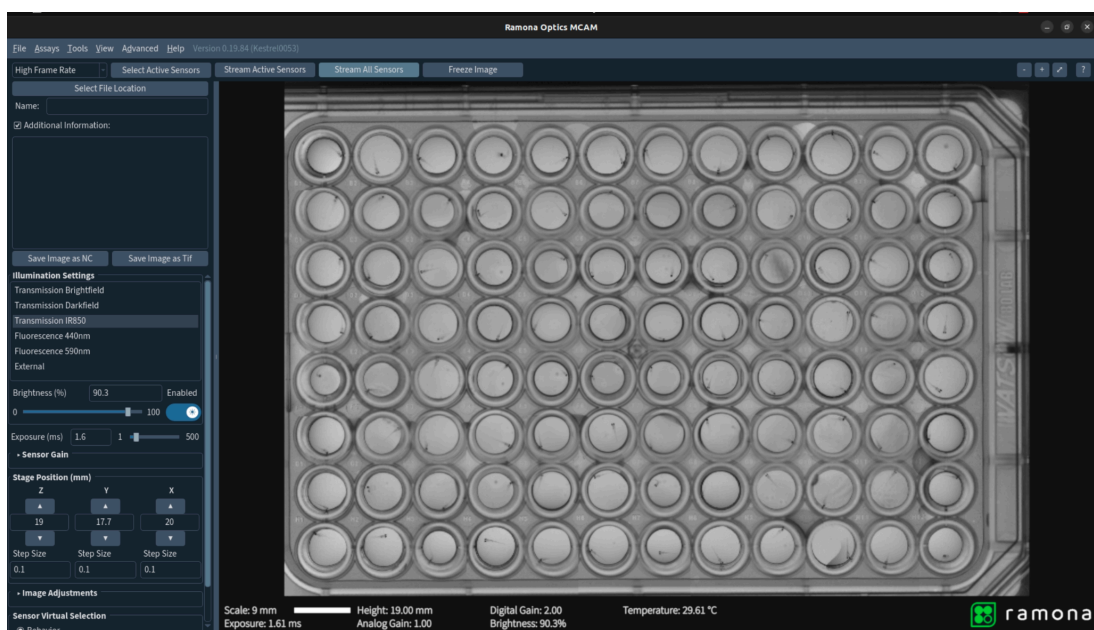


Fig 25a: GUI with stitched image.

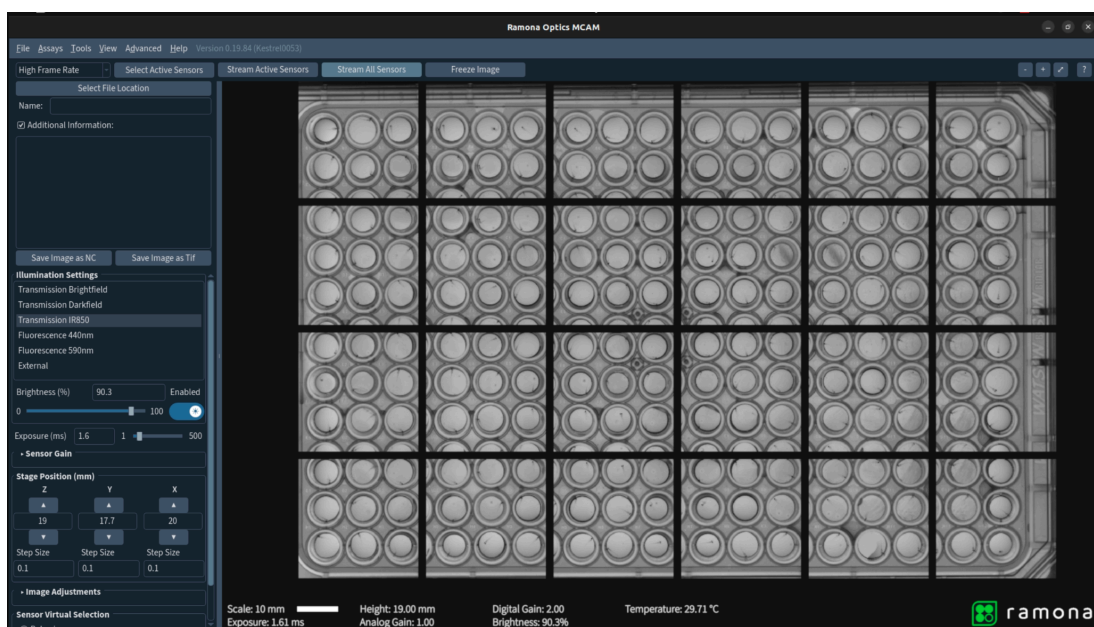


Fig 25b: GUI with unstitched image.



INTERPOLATE

Checking this allows the user to see a smooth image when zooming in to levels that would otherwise be shown as pixelated.

COLOR CORRECTIONS

Checking this box applies sensor corrections calculated during system calibration. The default state for this setting is active.

BLEND IMAGES

Checking this reduces the visibility of seam between stitched images, but may cause shadowing if features from overlapping images do not line up perfectly.

SOFTWARE WHITE BALANCE

Checking this compensates for the sensor's uneven responsiveness to red, green and blue when imaging in brightfield (white) light.

SHOW IMAGE LABELS

Checking this provides an overlay to display the well in each camera, if a well extracted dataset, or the label of the sensor that acquired the image.

SHOW OVERVIEW

Checking this shows the overview image in the top right of the screen to help keep track of the current view of the image scene. Predefined sizes of the overview window can be applied by clicking Small Overview or Large Overview, or the overview window can be adjusted by clicking on the edge of the overview window.

SHOW SCALE BAR

Checking this will make a scale bar appear in the picture-in-picture panel.

SHOW ACQUISITION INFORMATION

Checking this will make the key image specifications (height, exposure, analog gain, digital gain, brightness and temperature) appear in the picture-in-picture panel.

SHOW LOGO

Checking this will make the Ramona logo appear in the picture-in-picture panel.

DARK MODE

Checking this allows the user to see the dark mode of the interface, meaning that elements will be presented in darker colors and the background of the picture-in-picture panel will be black instead of white.

ADDITIONAL SETTINGS

The additional settings window allows the user to set the minimum and maximum values that the GUI will accept for the exposure and gain settings (Figure 26).

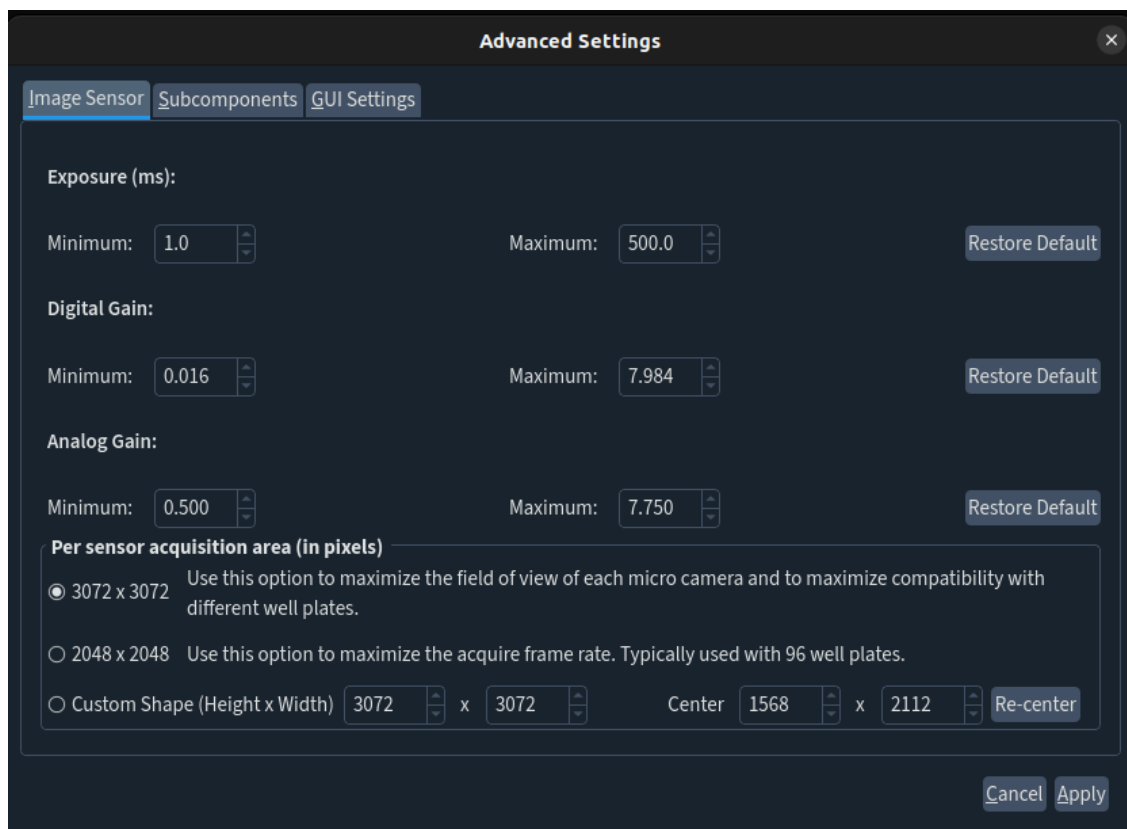


Fig 26: Additional Settings Window.



MCAM Viewer

The MCAM Viewer software allows for viewing images that have been previously acquired and saved using the MCAM GUI.

The MCAM Viewer software is provided as a distributable executable (see [MCAM Viewer Installer](#)) and can be run using Linux systems. Compatible file types include ".nc" files which use the netcdf4 data format as well as ".png", ".tif" and ".bmp" images.

In the MCAM Viewer, image contrast can be adjusted using the Gamma, Minimum Level and Maximum Level settings in the bottom left corner. Metadata that has been saved with image files is displayed on the left side of the viewer window. Additionally, functionality similar to the GUI is available (see [Graphical User Interface \(GUI\)](#) for more information) (Fig 27).

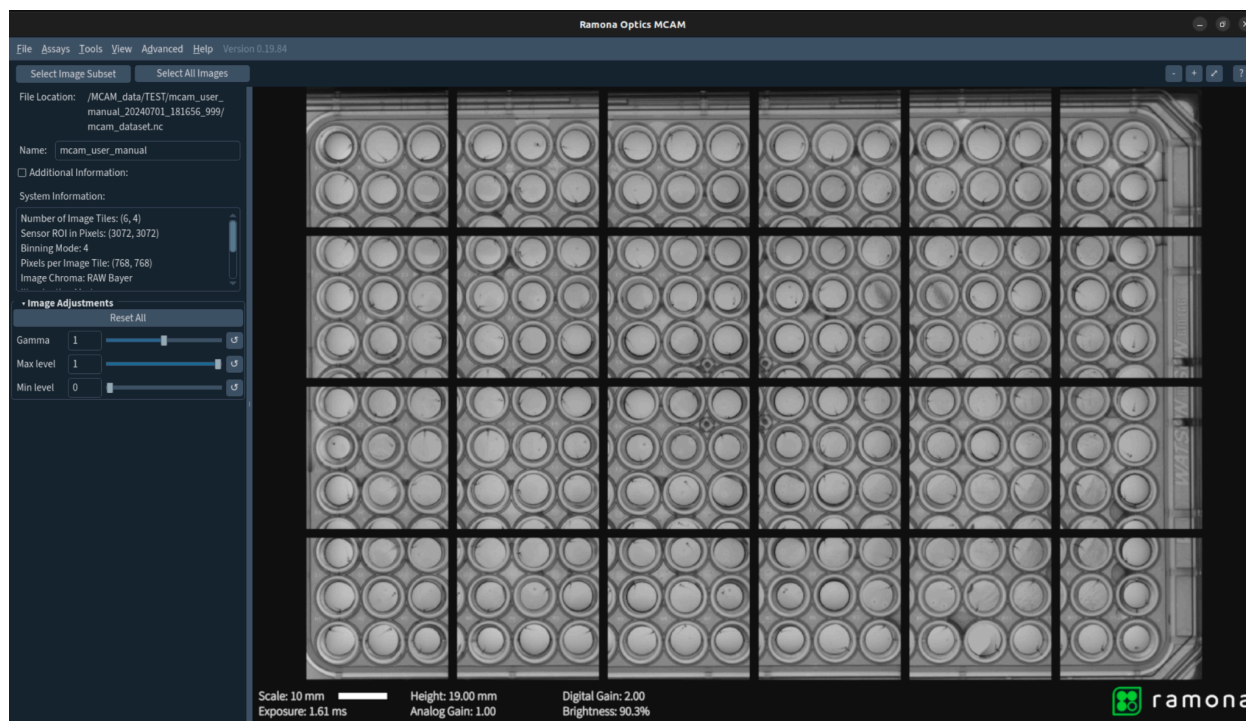


Fig 27: MCAM Viewer Window.



Video Playback

The MCAM Viewer allows for video playback with multiple settings affecting playback speed.

- Frame Skip allows the user to specify an interval of frames that will skip between frames.
- Viewer FPS allows the user to request a playback FPS. This is dependent on the amount of data in each frame.
- Viewer FPS will be limited to 60 therefore if the recording is done at a greater FPS Frame Skips need to be added to be able to visualize the video at the speed at which it was recorded.

Combining these options allows for a user to reach a higher effective FPS than otherwise possible (Fig 28).

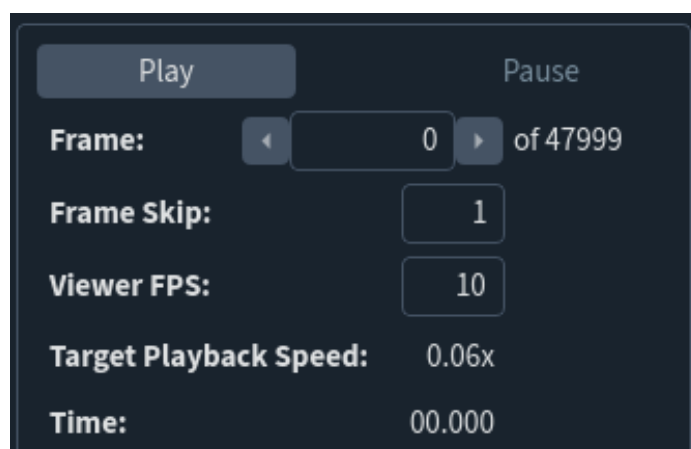


Fig 28: Video playback settings



MCAM Data Analysis

Documentation of our Python API with examples is available at docs.ramonaoptics.com.

Acquisition Considerations

Recommended Video Acquisition Parameters

For behavior analysis and motion tracking we recommend the following imaging parameters in “High Frame Rate” mode:

- Transmission Illumination: Transmission IR850
- Exposure: 2 milliseconds
- Brightness: 60%
- Digital Gain: 2.0
- Analog Gain: 1.0

A settings file to load the recommended imaging parameters for behavioral analysis is located [HERE](#) for download.

For eye analysis, data must be captured in either “Standard (bin x2)” or “High Resolution” mode in order to properly resolve the shape of the eyes. We recommend the following imaging parameters in “Standard (bin x2)” mode:

- Transmission Illumination: Transmission IR850
- Exposure: 2 milliseconds
- Brightness: 60%
- Digital Gain: 2.0
- Analog Gain: 1.0

For morphology analysis we recommend the following imaging parameters in “High Resolution” mode:

- Transmission Illumination: Transmission Brightfield
- Exposure: 10 milliseconds
- Brightness: 30%
- Digital Gain: 1.0
- Analog Gain: 1.0

Z-height must be set initially by the user to focus and properly resolve the image. This parameter is unit specific.

Infrared illumination is generally used in behavioral imaging experiments because zebrafish do not see wavelengths in the infrared spectrum and thus the lighting does not change their behavior. For some behavioral experiments such as prey capture and hunting behavior experiments, visible RGB spectrum lighting may be necessary so that zebrafish can see their prey.

Frame rate is an important parameter to consider and is likely specific to each experiment. For general use we recommend setting the frame rate to 120 frames per second (fps) however in some cases 160 fps may be achievable while for longer (>5 minute) durations, recordings may be limited to 30 fps. Please contact us for assistance in tuning this value for your experiment.

The captured dataset, containing both the video, and metadata relating to the experiment can be opened in the MCAM Viewer by drag-and-dropping the file into the Viewer window or by selecting “File > Load Settings” from the menu bar.

Imaging Models and Pixel Binning

The MCAM acquires video data in three different imaging modes: “High Resolution”, “Standard (bin x2)”, and “High Frame Rate”. The mode can be selected in the drop down menu in the top left corner of the GUI (Fig 29).

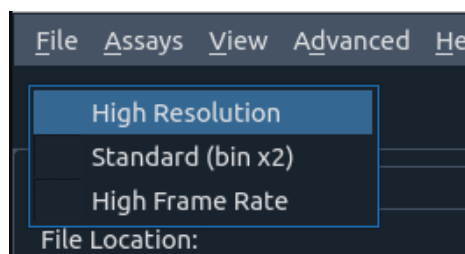


Fig 29: Imaging modes



In High Resolution mode all pixels in the camera sensor are activated and streamed. In Standard resolution mode, pixels are binned by two, meaning every other pixel is skipped and thus half of the pixels are streamed over the full sensor, halving the resolution. In High Frame Rate Mode, one in every four pixels is streamed so the resolution is reduced by a factor of four. The benefit of these imaging modes is that by streaming less data at one time, frame rate can be increased almost linearly.

Movement Blur: Frame-Rate vs. Exposure

Both frame-rate and exposure are important parameters to optimize. Reducing exposure means decreasing the amount of time the camera's sensor is exposed to light resulting in less light entering the camera and hitting the sensor, which can reduce motion blur. Reducing the exposure time also means that less light is entering the camera, which can result in an underexposed image. To compensate for this, the brightness of the MCAM is increased. Increasing frame rate gives you more chances of capturing each body position within a movement however increasing frame-rate also increases overall data size and should be monitored closely.

Sensor Shape for Frame-Rate Optimization

Video frame rate can be optimized by changing the camera sensor size. This is done by cropping which pixels are streaming from each camera sensor as it acquires frames. Each camera has 3136 x 4224 pixels total. Most images with a behavioral MCAM can be best acquired with a 3072 x 3072 or 2048 x 2048 pixel field of view. Specifically 24- and 96-well plates can be imaged with the smaller image shape to increase frame-rate to 160 fps. 48-well plates require a field of view of 3072 x 3072 pixels to capture all wells on the well plate which allows a maximum of 120 fps acquisition. To change the sensor shape:

STEP

ACTION

- 1
- Select "Additional Settings" Advanced menu on the toolbar, pictured below (Fig 24, Fig 30).

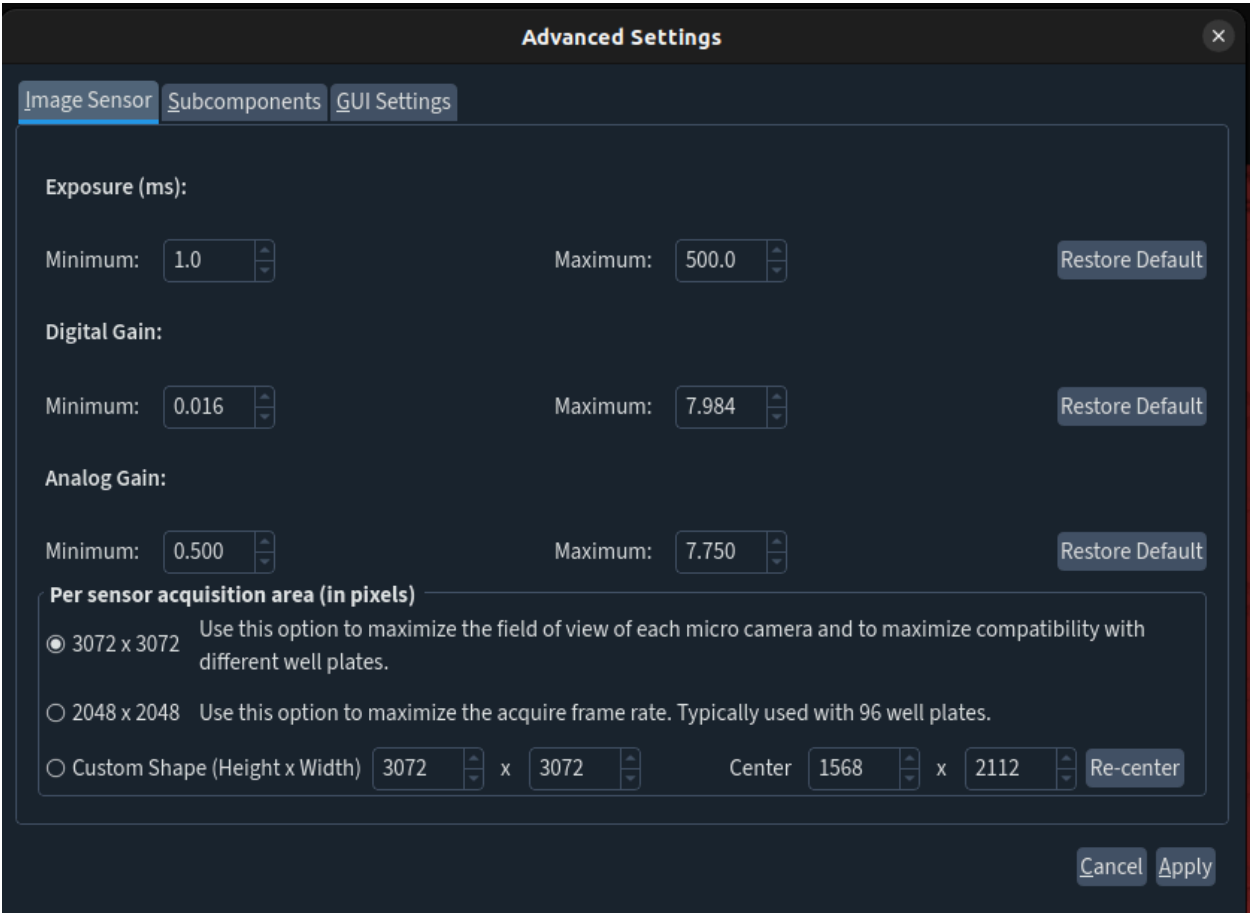


Fig 30: Advanced menu, additional settings panel

- 2
- In the bottom section "Per sensor acquisition area", select the desired field of view.

Video Compression

The first step of all analysis pipelines is to compress the raw “.nc” file dataset to an mp4 video and accompanying metadata. To accomplish this task, a Well Alignment Keypoint file must be created, specific to each MCAM unit and well plate being used. This file simply describes the location of the center and radius of each well within each image captured. This file only needs to be created once assuming there are no changes to the physical setup of an experiment. If the stage moves from side to side or the well plate in use is modified, a new keypoint file may be necessary.

Creating a Keypoint File

STEP	ACTION
1	To create a new Keypoint file, open a .nc file in the MCAM Viewer and open the “Well Alignment” panel from the “Tools” menu on the menu bar (Fig 19).
2	Input the number of columns and rows corresponding to the well plate in use. 96-well plates have 12 columns and 8 rows, 48-well plates have 8 columns and 6 rows, and 24-well plates have 6 columns and 4 rows as shown below. For machine learning tracking the diameter value is specific for each tracking model and for proper tracking and analysis functionality it is required to enter 1024 pixels for 96-well, 1408 pixels for 48-well, and 2048 pixels for 24-well. The exact diameter in millimeters will update automatically and may vary slightly between systems depending on the calibrated pixel width (Fig 31).

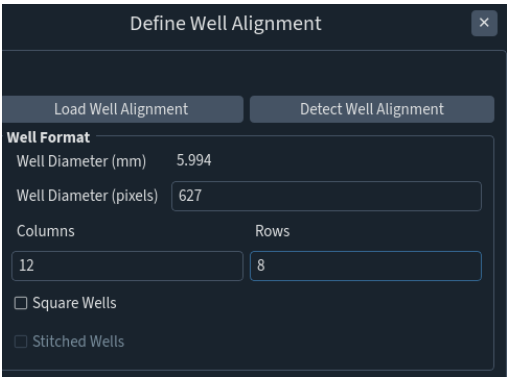


Fig 31: Well Alignment panel

3	Once the correct parameters have been entered into the required fields, click “Detect Well Alignment”.
---	--

- 4 When initially detecting well keypoints, it is likely this step will fail as shown below with circle indicators not lining up with wells. This is expected (Fig 32).

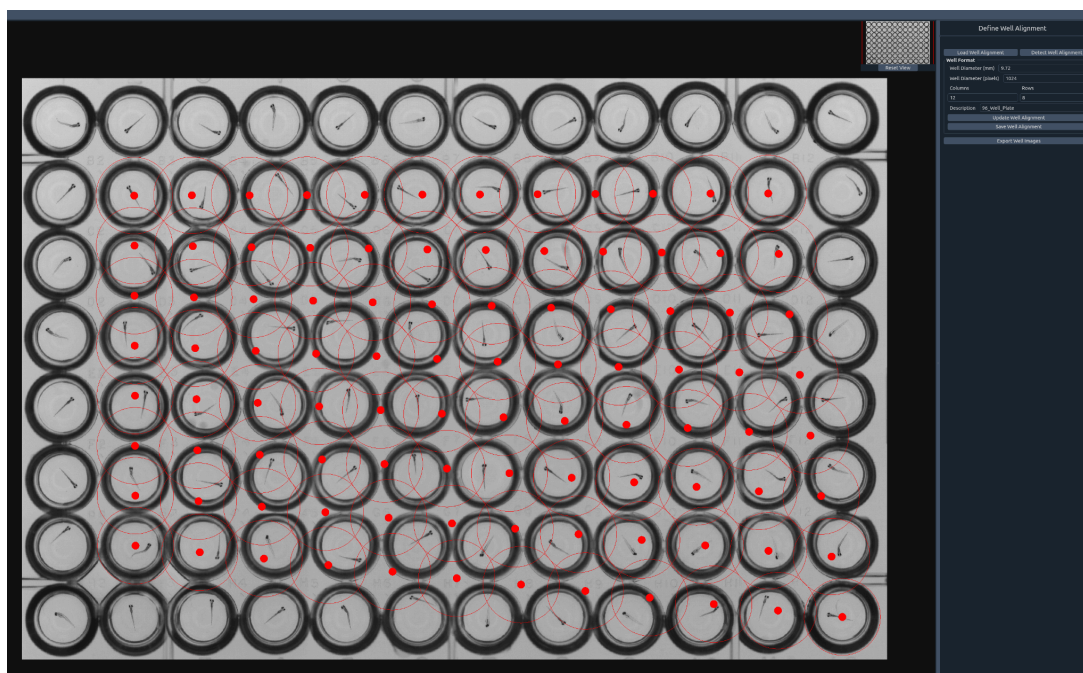


Fig 32: Well Alignment initial guess.

- 5 To correct this initial guess, only the four corners must be repositioned. Double click on the center point of a corner circle and drag the circle over a corner well until it lines up with the outer edge of the well. This is shown in the top left corner of the image below (Fig 33).

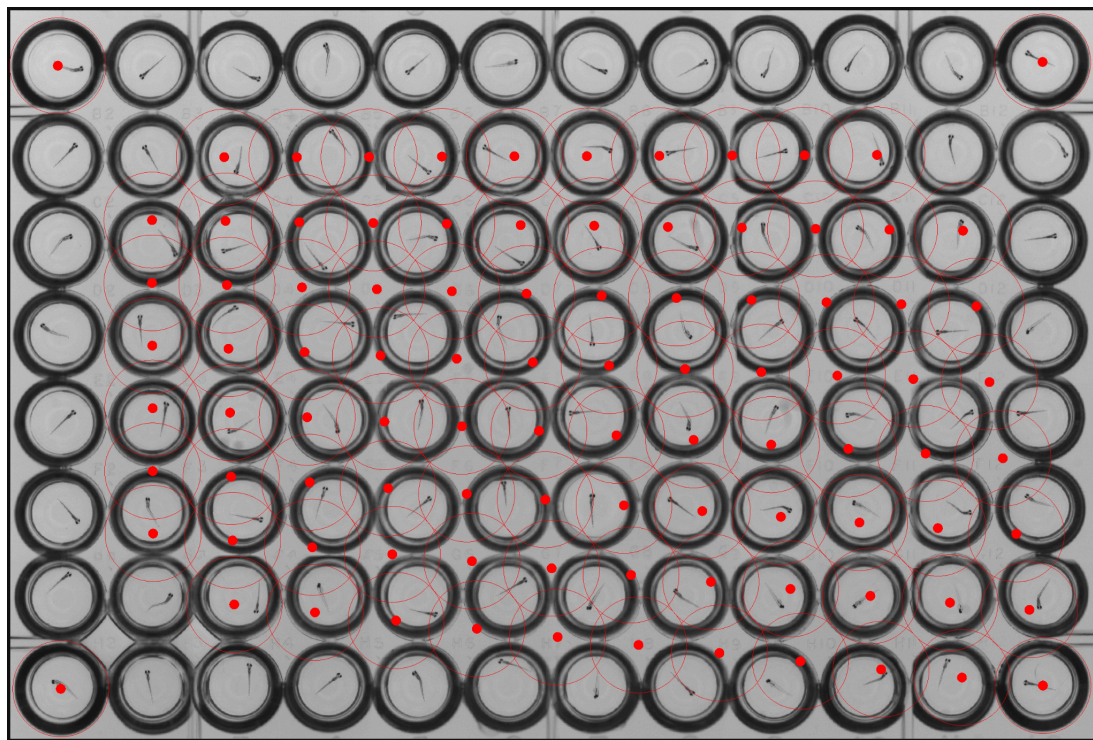


Fig 33: Well Alignment corrected initial guess.

- 6 Once the four corner well indicators are aligned with the corner wells, click “Update Well Alignment” from the right panel to reposition the remainder of the indicator circles (Fig 34).

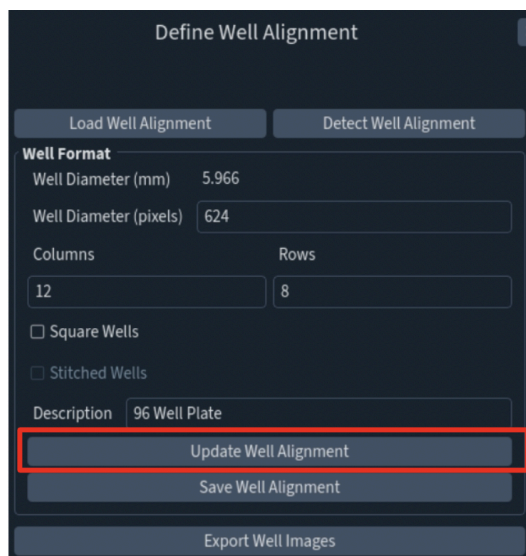


Fig 34: Well Alignment panel with “Update Well Alignment” option.



- 7 Ensure that well indicator circles and center points are correctly aligned with each well (Fig 35). Individual adjustments can be made if necessary by again double clicking on the center point of each red circle, and dragging to the correct location.

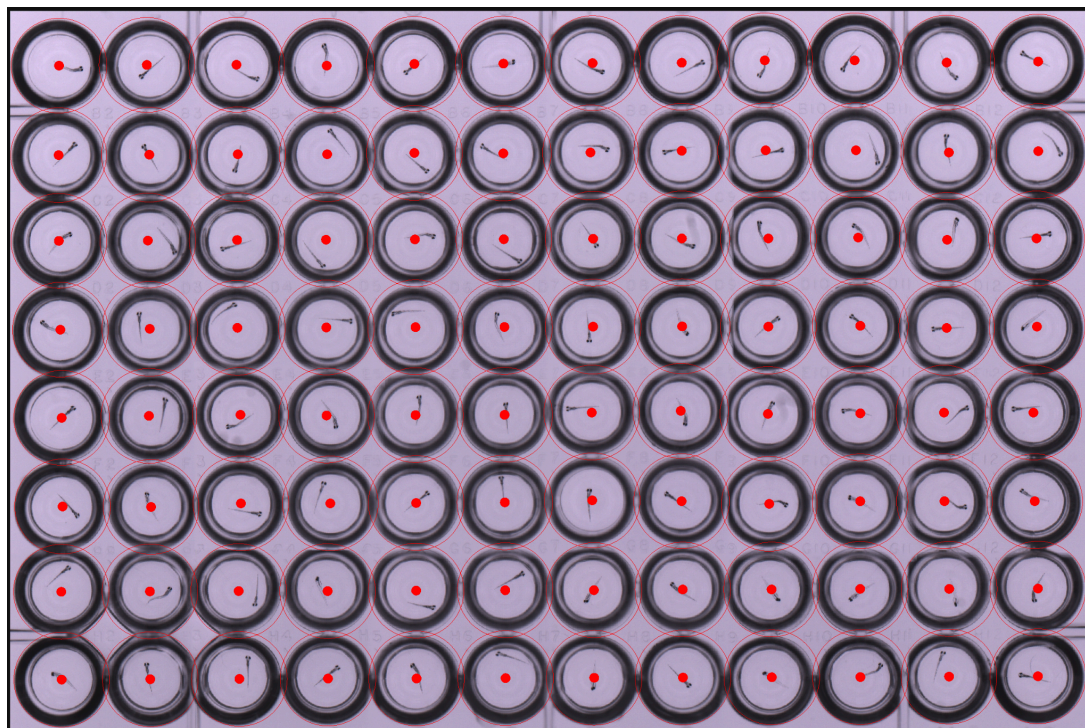


Fig 35: Fully aligned Well Alignment.

- 8 Select "Save Well Alignment" from the right panel.
- 9 Select a save location and file name for this ".json" file. We recommend file names with a date and/or experiment description as well as the well plate type (Fig 36).

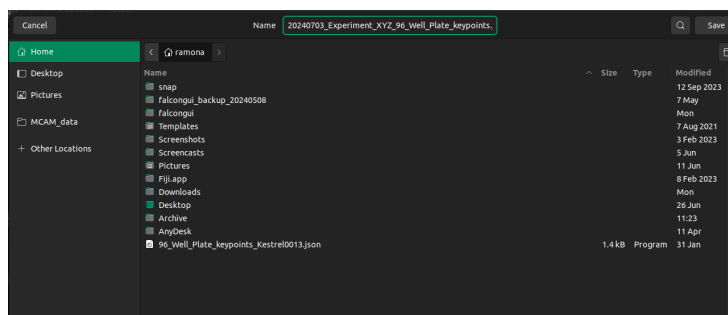


Fig 36: Format for saving well alignment.

Square Well Plate Keypoints

When using a 96 square well plate, the above procedure should be followed similarly, starting with selecting the Square Well option which will result in square well indicators (Fig 37).

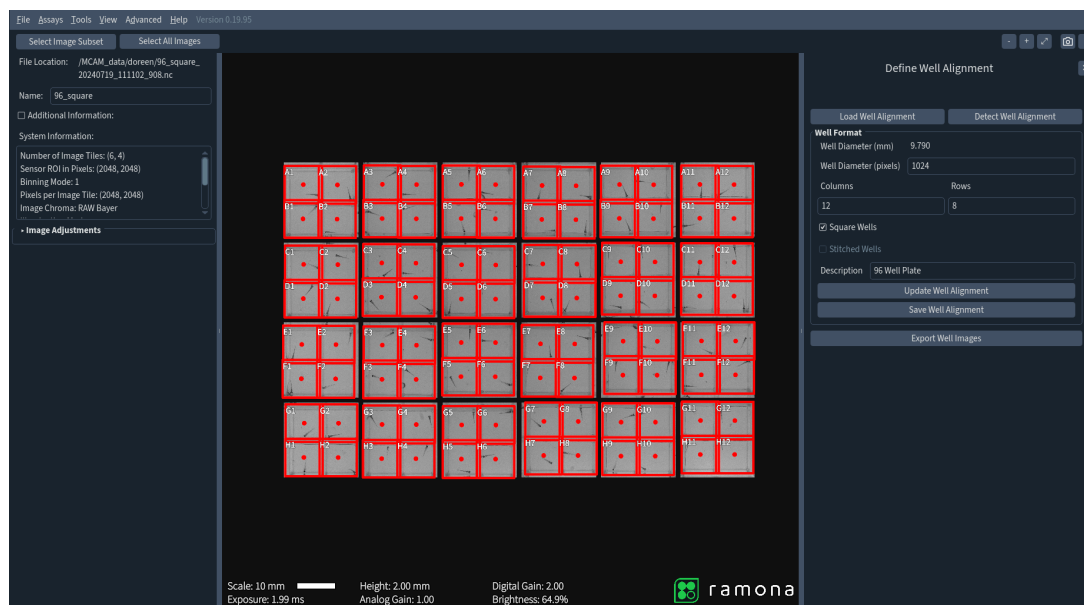


Fig 37: Well Alignment for square well plates.

Compress the Video

Prior to analysis, the raw (.nc) data format acquired by the MCAM must be compressed to a mp4 file format. Video compression allows for reduced data sizes while still maintaining data fidelity.

STEP	ACTION
1	Open the “Compress Video” panel within the “File” menu (Fig 38).

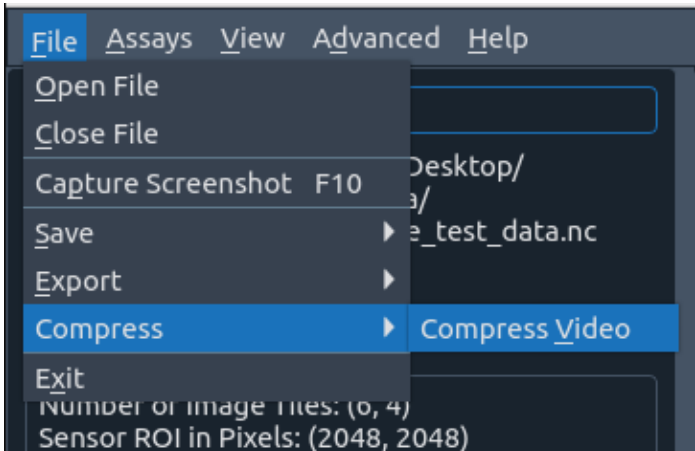


Fig 38: File menu.



- 2 Click "Select Alignment File" and select the well keypoints file corresponding to the well plate in use that was saved previously (Fig 39a). If the file has already been aligned then this will not be available and the compression can be run without this step (Fig 39b).

The screenshot shows a dark-themed window titled "Compress Dataset" with a close button (X) in the top right corner. Below the title, there are two rows of controls. The first row is labeled "Well Alignment" and contains a button labeled "Select Alignment". The second row is labeled "Export Suffix" and contains a text input field with the value "_export". Below these, there are two sections, each preceded by an unchecked checkbox. The first section is titled "Compression Options" and contains a checkbox labeled "Generate One MP4 per Video Tile" and a text label "Encoder: hevc_nvenc". The second section is titled "Batch Analysis" and contains a single line of text. At the bottom of the panel is a large, light-colored button labeled "Compress".

Fig 39a: Panel for dataset compression with alignment option.

This screenshot is identical to the one in Fig 39a, showing the "Compress Dataset" panel. However, the "Select Alignment" button is now disabled, appearing as a light gray button. The rest of the interface, including the "Export Suffix" field, the "Compression Options" and "Batch Analysis" sections, and the "Compress" button, remains the same.

Fig 39b: Panel for dataset compression without alignment option.

Note the blue circles that appear, marking where the wells will be extracted and compressed. Well alignment cannot be changed from this panel (Fig 40).

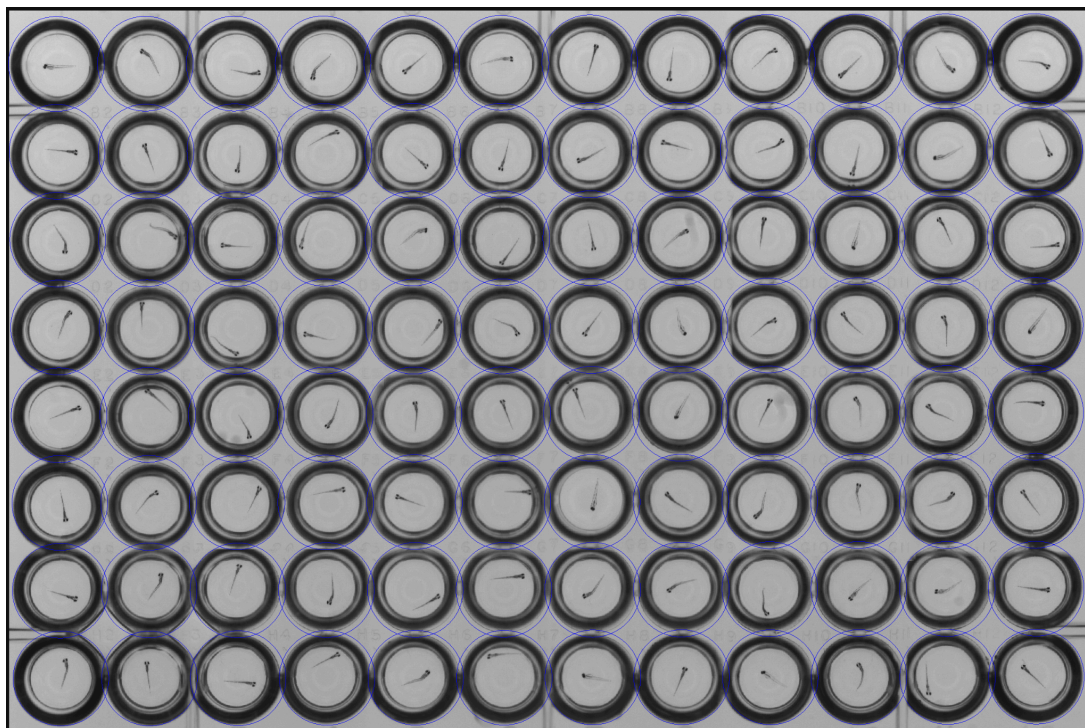


Fig 40: Well alignment overlay shown before final video compression.

- 3 Click compress and the compressed mp4 dataset with chosen export suffix will be saved in a new folder in the same directory as the original .nc file (Fig 41).

Name	Size	Modified	
96_well_plate_test_data_export	2 items	17:29	☆
96_well_plate_test_data.nc	1.6 GB	25 Sep 2022	☆
96_well_plate_test_data_keypoints.json	16.1 kB	10 Mar	☆

Fig 41: File format compression results will save as.



- 4 Multiple files can be compressed at once by clicking the “Batch Analysis” checkbox and adding more files to the queue. **Note that all files to be compressed in a batch must share the same well alignment file (Fig 42).**

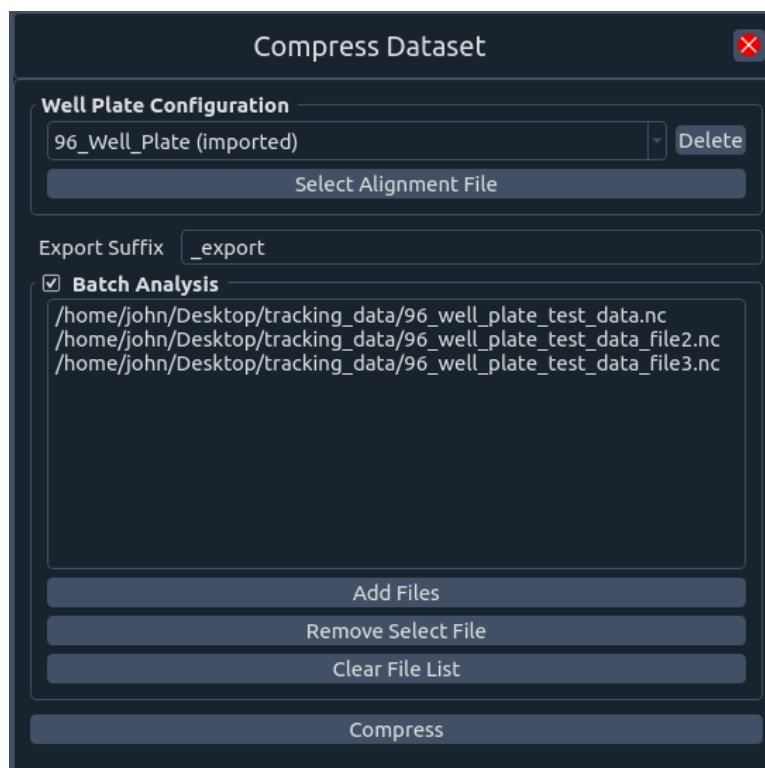


Fig 42: Batch analysis option within video compression.



Compression Options

For ease of data management, and repeatable analysis, we recommend compressing videos using H264 with the acquired video stored in a file with a mp4 extension (Fig 43). Ramona Optics has tuned the encoding parameters to prefer higher quality videos that are suitable for re-analysis with Ramona's MCAM software. Ramona will attempt, when possible, to store the entire dataset in a single mp4 with the metadata stored in a file called metadata.nc. To maximize the quality of the videos, compression ratio, and analysis speed, all datasets are compressed at a fixed frame rate of 30 frames per second. This is accomplished by encoding many individual video tiles into one composite video which can be encoded and stored as one object as well as retrieved and played back in the MCAM Viewer efficiently. The default behavior of the compression panel will generate a "tiledmp4" when possible and no option modifications are necessary within the MCAM Viewer compression panel.

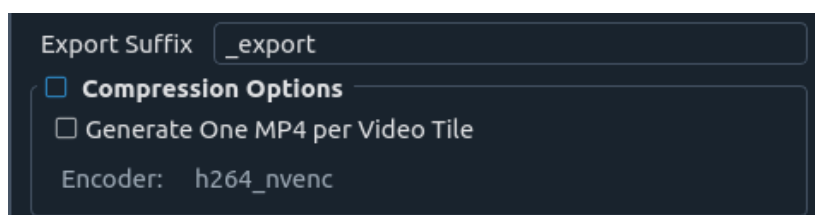


Fig 43: Default compression options.

At higher resolutions, combining many video streams together may not be possible due to software limitations of available video encoders. In this case one mp4 will be generated per well extracted and these will be stored as a "multi_mp4" dataset. For example, a dataset contains 24-well plate data acquired in High Resolution imaging mode with a 2048 x 2048 sensor size. If this data were encoded together into one tiled mp4 video, the resulting image size would be 12,288 x 8,192 pixels. Current video encoders reach a limit of 4096 x 4096 pixels although this limit is evolving quickly. In this case, an individual video will automatically be generated for each well resulting in 24- 2048 x 2048 pixel videos. This dataset can still be loaded and viewed together using the MCAM Viewer by asking the MCAM Viewer to open the associated metadata.nc file. The user can select "Generate One mp4 per Video Tile" to force the video compression on a tile-by-tile basis (whether this is a camera-by-camera basis, or well-by-well basis) as shown below (Fig 44).

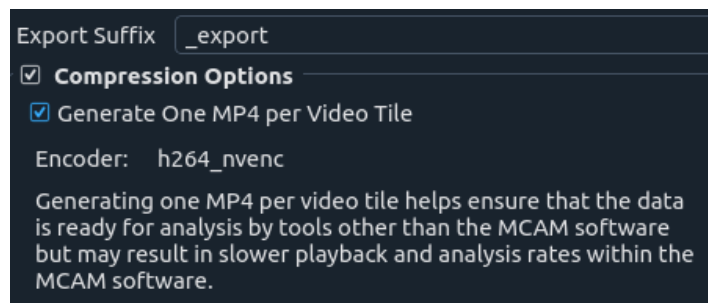


Fig 44: Compression panel option which enables users to select the option of "Generate One MP4 per Video Tile".

The default video encoder (h264_nvenc) will utilize the NVidia GPU provided with the MCAM workstation to improve performance of video encoding.

Compressed datasets (both "tiledmp4" and "multi_mp4") can be loaded for display into the MCAM Viewer by either double clicking the "metadata.nc" file in dataset folder or be drag-and-dropping the full dataset folder into the MCAM Viewer window.

Video Orientation

Prior to generating the final outputs, the MCAM video data is stored in one of two classes of formats:

- 1 In an .nc file where both the metadata and the video data are stored within the same file.
- 2 In a folder containing an metadata.nc file for the metadata and an .mp4 file containing the video data to be analyzed.

To avoid accumulated error before the final results are computed, minimal lighting and orientation corrections are applied to the mp4 file. Therefore, the video as played back by a standard video player may not appear exactly as it does in the MCAM Viewer. For datasets stored in mp4 formats, an EXIF orientation value of 8, as is common for Kestrels in the upright configuration, will result in an apparent rotation of 90 degrees in most video players. This rotation does not apply to the results stored in tabular format.

For more information on how to use EXIF the orientation metadata in your own analysis pipelines, please refer to the [Loading Image Files and EXIF Orientation](#) section.



MCAM User Interface Activity Metric Analysis

MCAM Workflows Manual

For details on specific imaging and analysis workflows available within MCAM Software for the Kestrel and the Vireo, please refer to the Workflows Manual in Ramona's online documentation: <https://docs.ramonaoptics.com/>.

Zebrafish Embryonic Photomotor Response Assay

The Zebrafish Embryonic Photomotor Response (EPMR) assay (Fig 45) consists of 3 recording periods, each separated by a light flash. The first period is the Background period, followed by the first flash. The second period is the Excitatory period followed by the second flash. The last period is the Refractory period, ending the assay. The assay records at 16 fps and defaults to sensor and illumination settings that are tested to work well with the analysis. The analysis is performed in parallel with the acquisition and computes the number of changed pixels between frames via the following formula:

$$\sum_{x=1}^n \sum_{y=1}^m \left[\frac{|P_i(x,y) - P_{i+1}(x,y)|}{(P_i(x,y) + P_{i+1}(x,y)) \div 2} > T \right] \times [|P_i(x,y) - P_{i+1}(x,y)| > D]$$

Where:

- $P_i(x,y)$ is the pixel value ranging from 0 to 255 of frame i
- $P_{i+1}(x,y)$ is the pixel value ranging from 0 to 255 of frame $i + 1$
- T is a relative threshold set to .1
- D is an absolute threshold set to 20
- m is the frame height
- n is the frame width

This analysis is exported as a csv with a column per each image and a row for each time point. A plot for each column is also exported for quick review, along with the full video dataset. These are all placed in the provided directory.



Acquire Zebrafish EPMR

Assay Protocol

LoadSave

Well Alignment

Select Alignment

Save Format

Save to RAM

Assay Settings

Background Period (sec)

30

Excitatory Period (sec)

9

Refractory Period (sec)

9

Duration (sec)

Lux

First Light Flash

1

6000

Second Light Flash

1

6000

Flash Illumination Mode

White Light

☐ Advance Settings

Pixel Threshold

.1

Difference Threshold

20

Frame Rate

16

Start delay (seconds)

0

Select File Location

Name:

_epmr_video

Additional Information

Prepare Video

Fig 45: Zebrafish Embryonic Photomotor Response Assay panel.



MCAM Viewer Activity Metric Analysis

Activity metric analysis provides a rapid method for computing movement in a video. This is accomplished by subtracting each pair of adjacent frames in a video to quantify pixel change. All pixels are treated equally so while total movement is measured over time, we cannot determine “how” something within a frame moves.

In the MCAM Viewer, Activity Analysis can be accessed from the “Assays” file menu (Fig 46, Fig 47).

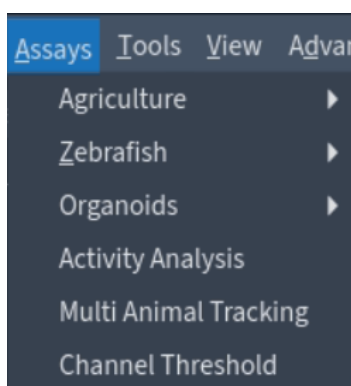


Fig 46: Assays menu.

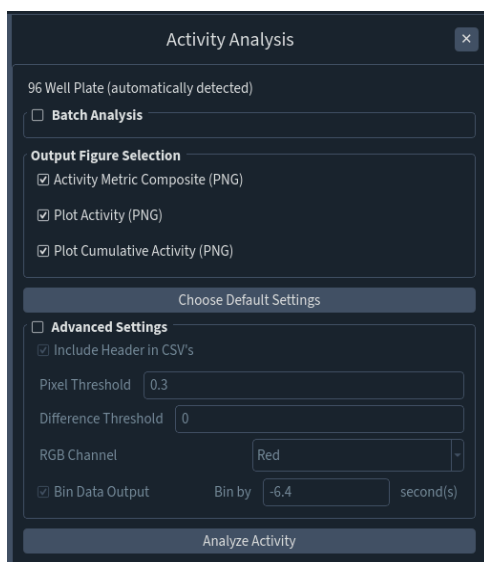


Fig 47: Activity Analysis panel.

MCAM Viewer Activity Metric Outputs

ACTIVITY METRIC COMPOSITE

The activity metric composite shows an overview of total activity within each well or video tile. Values are displayed on a log10 scale (Fig 48).

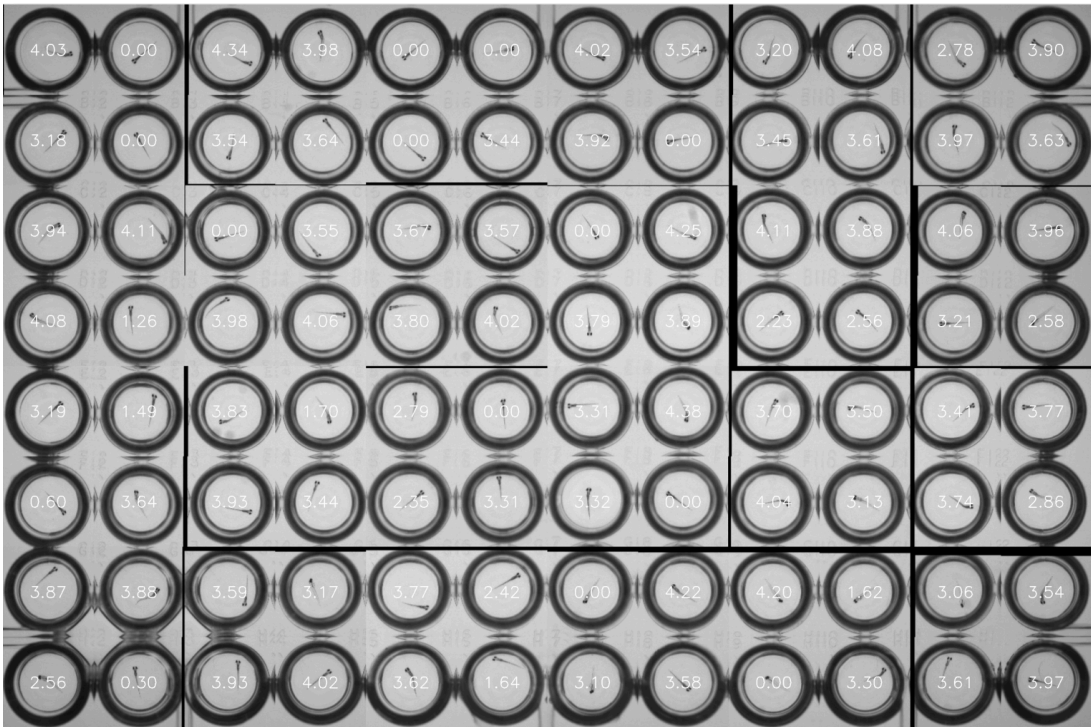


Fig 48: Example of activity metric composite.



ACTIVITY PLOTS

A plot of activity over time (Fig 49a) is generated as well as a plot of cumulative activity over time (Fig 49b). Axes are normalized to the maximum value of the overall dataset to facilitate comparison between wells or video tiles.

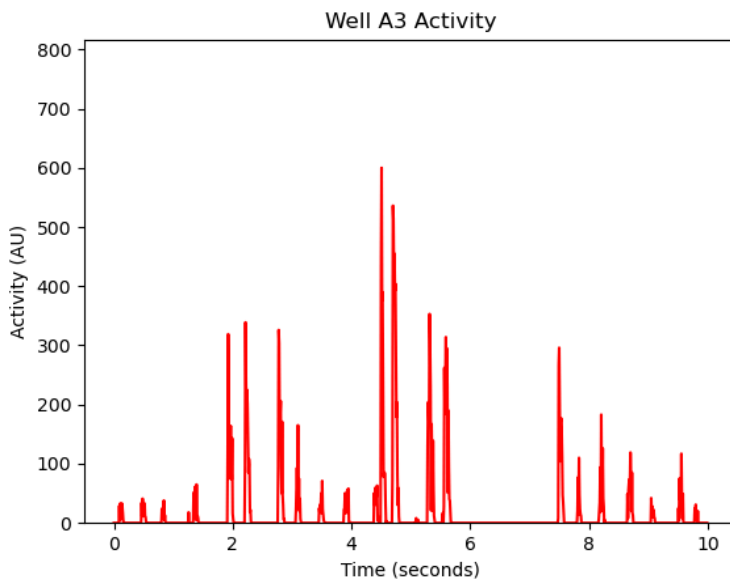


Fig 49a: Example plot of well activity over time.

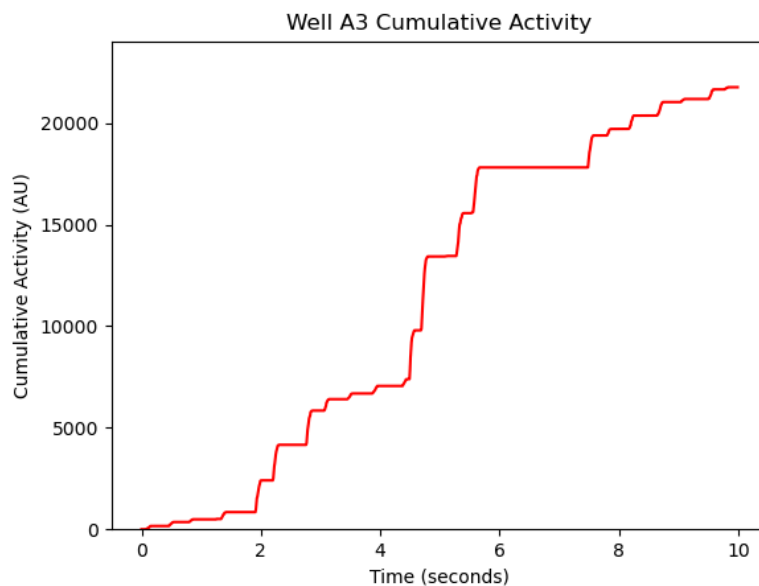


Fig 49b: Example plot of cumulative well activity over time.

ACTIVITY DATA (.CSV)

The tabular CSV output “activity_data.csv” displays the sum of pixel differences for each frame on a per well basis (Fig 50).

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y
1	row	A	A	A	A	A	A	A	A	A	A	A	A	B	B	B	B	B	B	B	B	B	B	B	B
2	column	1	2	3	4	5	6	7	8	9	10	11	12	1	2	3	4	5	6	7	8	9	10	11	12
3	time																								
4		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5		0.00833266	7	0	0	0	0	0	0	543	499	541	412	0	0	0	0	0	0	0	0	51	667	570	566
6		0.016665321	8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7		0.024997982	11	0	0	0	0	0	0	0	9	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8		0.033330643	1	0	0	0	0	0	0	0	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9		0.041663304	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10		0.049995965	0	0	0	16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0
11		0.058328626	0	0	0	7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12		0.066661286	0	0	0	28	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
13		0.074993947	0	0	0	62	0	0	0	0	21	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Fig 50: Example csv of activity data.

MCAM Zebrafish Analysis

Zebrafish analysis software includes 8-point tracking and morphology analysis as well as the activity metric analysis discussed previously.

Machine Learning Tracking

Zebrafish 8-point tracking with the MCAM is fully automated end-to-end across an entire well plate. Currently 24-, 48- and 96-well plates are supported in Visible and Infrared transmission illumination. Tracking relies on custom-trained machine learning algorithms that label key features of interest. Our Zebrafish models consider eight key-points throughout the fish. These include: the Snout, Left Eye, Right Eye, Center-point (center of swim bladder), Caudal Fin (tail-tip), Mid-Tail (between center and caudal fin bisecting the previous two points), Between-Center-and-Mid, and Between-Mid-and-Caudal. Labeled points are shown in the figure below (Fig 51). For each key-point an x and y coordinate within the image are recorded with a confidence value attached. These values are output to the user as raw tracking data for each well in a plate as well as further analysis, visualizations and derived metrics. Machine learning tracking models improve over time by adding more diverse training data.

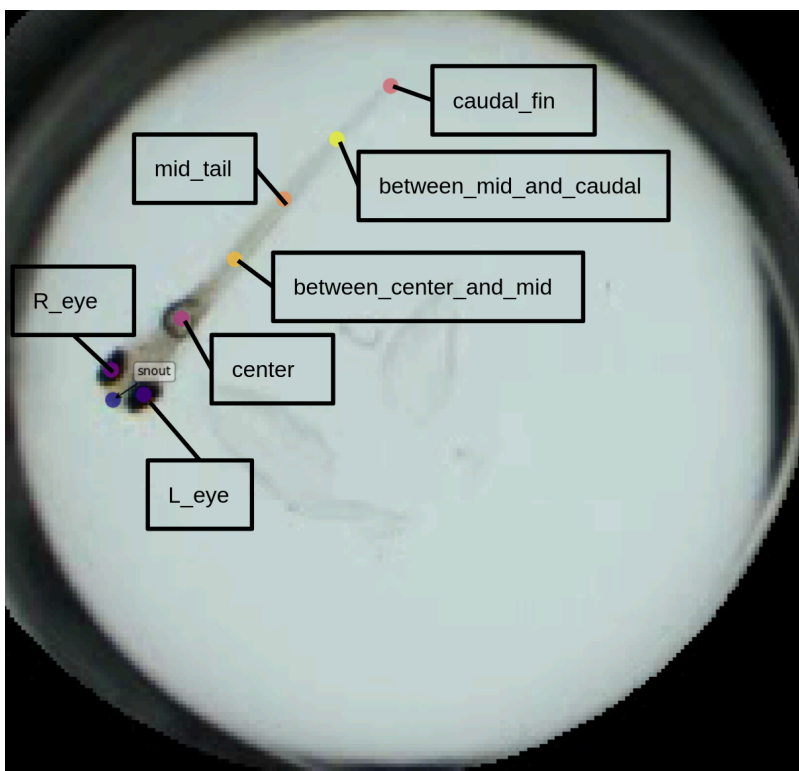


Fig 51: 8 points used for machine learning Zebrafish tracking.

Once a video .nc file has been compressed, load the compressed dataset by dragging and dropping the entire directory into the MCAM Viewer window. Alternatively the "metadata.nc" file within this directory can be double clicked to open the compressed dataset. The "Zebrafish Assay" panel (Fig 52) can be opened within the MCAM Viewer from the Assays > Zebrafish > Zebrafish Assay menu in the menu bar (Fig 46).

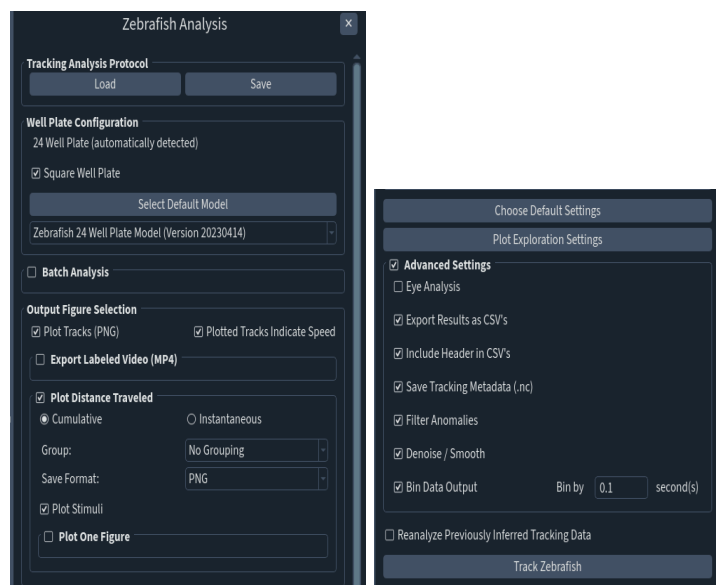


Fig 52: Animal Tracking Assay panel.

STEP	ACTION
1	The well plate configuration (24-, 48-, or 96-well) will be automatically detected by the MCAM Viewer software.
2	If square well plates are in use, check the box indicating “Square Well Plate”.
3	Select the output files of interest. We recommend always generating the “Raw 8-keypoint Model Output” which includes the raw tracking data for each well which is used in further analysis computation. Output files are described below.
4	Click “Track Zebrafish” to begin the automated tracking analysis.
5	Multiple files can be tracked and analyzed at once by clicking the “Batch Analysis” checkbox and adding more files to the queue. All files must be of the same well plate configuration (24-, 48-, or 96-well) and use one tracking model to be analyzed together. Note: to load multiple compressed datasets into the viewer for batch analysis, the entire dataset folder should be selected, not the “metadata.nc” file within the dataset folder.
6	If “Reanalyze Previously Inferred Tracking Data” is checked, the software will find a pre-existing “tracking_metadata.nc” file and re-run all analyses using this data.

Tracking Analysis Outputs

OUTPUT FILES

Output files are saved in a "Results" folder located in the same parent directory as the compressed/exported dataset (Fig 53a & b). Additionally "tracking_metadata.nc" contains the complete results in Xarray netcdf format for further pythonic analysis.

DATASET

Name	Size	Modified	
 results	7 items	17:56	☆
 dataset.mp4	109.3 MB	17:29	☆
 metadata.nc	986.3 kB	17:29	☆
 tracking_metadata.nc	7.1 MB	17:56	☆

Fig 53a: File structure of tracking analysis output files.

RESULTS

Name	Size	Modified	
 composite_tracked_video.mp4	1.8 MB	17:56	☆
 distance_traveled_key_metrics.csv	5.2 kB	17:56	☆
 distance_traveled_metrics.csv	1.1 MB	17:56	☆
 plotted_tracks.png	8.1 MB	17:56	☆
 skeleton_data.csv	6.5 MB	17:56	☆
 tail_angles.csv	3.6 MB	17:56	☆
 tracking_data.csv	7.3 MB	17:56	☆

Fig 53b: File structure of tracking analysis output files.



- Raw Tracking Data - "tracking_data.csv" - CSV file containing 8 key-points per fish, x, y coordinate and confidence for each keypoint. Note: This file is likely too large to be opened in Excel or other spreadsheet softwares due to the large number of columns in the data array. This file can be opened and manipulated using Python, please contact us for example scripts to open this file and to learn more.
- Plotted Tracks - "plotted_tracks.png" - Visualization of movement over time from blue (earliest, cold) to red (most recent, hot). If "plotted tracks indicate speed" is selected during analysis, the color gradient represents is from blue (slow) to red (fast) and green is any speed over 50 mm/second.
- Distance Traveled Data - "distance_traveled_metrics.csv" - Distance traveled and speed calculated for each frame.
- Aggregate Metrics - "distance_traveled_key_metrics.csv" - Summarized peak speed, total distance traveled, and percent of frames found to be not confidently tracked for each well.
- Video Composite - "composite_tracked_video.mp4" - mp4 video containing key-point and skeleton labeled fish for all wells of a well plate.
- Skeleton Data - "skeleton_data.csv" - The length, orientation, and tracking confidence of each of the eight segments defined by two tracked keypoints for each well. Angles in this output are defined with reference to the image coordinate frame and have rotations with counter-clockwise orientation.
- Tail Angles - "tail_angles.csv" - For each well, four angles, and tracking confidence for each of these angles, are given each describing the angle formed by the "center-to-snout" segment and the center to each of the four tail keypoint segments. More information on this analysis can be found below in the "Tail Bend Analysis" section.
- Fish Lengths - "fish_lengths.csv" - Fish length, head length, and tail length are given on a per well basis. More information on this analysis can be found below in the "Fish Length" section.
- Eye Analysis - "eye_analysis.csv" - Eye position, distance between eyes, eye area, fish orientation, and eye angle are given on a per well basis. More information can be found below in the "Eye Analysis" section.

The CSV header includes MCAM information and may or may not be useful to the user. Users can specify whether to include a header by selecting the corresponding checkbox in the Zebrafish Analysis panel.



DATA SIZES

In the tables below, "Raw Outputs" refers to the sum of all individual well files generated by the tracking process, "Minimal Outputs" refers to the sum of individual well mp4 files with all other individual well files removed and "Results" refers to the sum of all final files located in the results folder.

To use this table in estimating how much disk space you need for your experiment, you should assume that you need the sum of the "Raw Outputs" and "Results" combined for initial processing. Once this process is complete, individual well files can be removed (except for individual mp4's that you will want to save in case you decide to re-track your data) to reduce the data footprint. The current GUI default will remove extra files automatically.

96 WELL PLATE

DURATION	FPS	RAW OUTPUTS (GB)	MINIMAL OUTPUTS (GB)	RESULTS (GB)
30	30	0.3	0.12	0.1
	160	1.33	0.71	0.5
60	30	0.54	0.26	0.19
	160	2.6	1.43	0.1
300	30	2.44	1.34	0.93
	160	12.78	7.2	4.91

24 WELL PLATE

DURATION	FPS	RAW OUTPUTS (GB)	MINIMAL OUTPUTS (GB)	RESULTS (GB)
30	30	0.09	0.05	0.03
	160	0.57	0.29	0.14
60	30	0.2	0.1	0.05
	160	1.15	0.6	0.27
300	30	1.08	0.55	0.25
	160	5.8	2.98	1.34

EXAMPLE TRACKING WORKFLOW

An example workflow has been prepared that you can use to understand this process. It can be run either from a python script or from the MCAM GUI. If you would like to run it, please download the following files:

- [96-well plate .nc file - 120 fps, 10 second duration, IR illumination](#)
- [Well keypoint file](#)

FROM THE MCAM GUI

STEP	ACTION
1	Open the .nc file you have downloaded by double clicking on the file in the MCAM Viewer software.
2	Compress the file following the directions above in the section titled "Video Compression".
3	Load the compressed dataset and use the directions above in the section titled "MCAM Zebrafish Analysis" to track the compressed dataset.
4	An output folder will be created in the same location as the compressed dataset.

FROM A PYTHON SCRIPT

The example script to run the workflow is titled "Zebrafish Tracking Workflow" and can be found here:
<https://docs.ramonaoptics.com/tracking.html>

We recommend that the "dataset.mp4" and all metadata files should be retained as tracking can be reproduced from these files alone in the future if re-tracking is necessary.

To view the accuracy of zebrafish tracking, we recommend the "plotted_tracks.png" and "composite_tracked_video.mp4" for visualization.

The result is located [HERE](#) for download and viewing.



Visualizations

PLOTTED TRACKS

Plotted tracks display the path of each fish over the duration of the video analyzed. Color gradients can display time from blue (earliest, cold) to red (most recent, hot). If “plotted tracks indicate speed” is selected during analysis, the color gradient represents is from blue (slow) to red (faster, 50 mm/sec) and green is any speed over 50 mm/second (Fig 54).

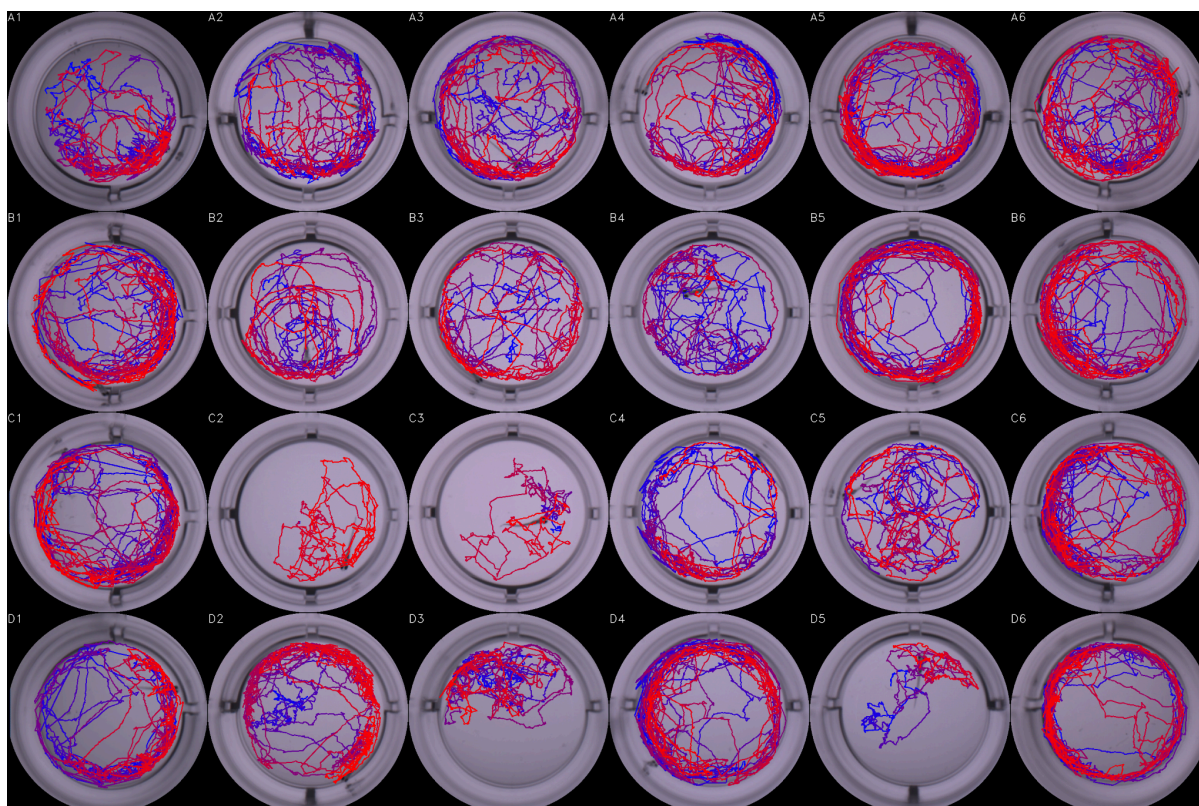


Fig 54: Example of plotted tracks result.



COMPOSITE VIDEO

The composite video displays each well in the array with the skeleton overlaid on each fish in green (Fig 55).



Fig 55: Example of composite video.



Assays

THIGMOTAXIS

According to one publication (Schnorr S., et. al, 2011), the authors validate a Thigmotaxis assay using a 24 well plate (16.2 mm diameter) with both light/dark conditions and stimulant/depressant chemical experimental conditions. The authors comments (Materials and Methods Section 2.3, pg. 368) that well plate selection is determined by assuring the "swimming arena must be sufficiently large to allow distinction between inner and outer zones" and to do this both inner and outer zones must be "at least equivalent or larger than the body length of the larvae (approx 4 mm for larvae aged 5 dpf)". It is also noted that a 6 well or 12 well format could work for this assay, however using a 24 well plate the area of inner and outer zones are equal "thus ruling out biases in the analysis of zone preference related to differences in zone size." Finally they comment that both 96- and 48-well plate formats are likely too small to fit these requirements.

Reference: Measuring thigmotaxis in larval zebrafish Schnorr et al. 2011

10.1016/j.bbr.2011.12.016

<https://www.sciencedirect.com/science/article/abs/pii/S0166432811008758?via%3Dihub>

This assay has been formulated with the 24-well plate in mind. A schematic is given below showing the concentric zones of each well of a 24-well plate. 4 mm is considered the default outer zone width however this can be defined by the user (Fig 56). An example script for this assay based on MCAM tracking data can be found at: <https://docs.ramonaoptics.com/tracking.hl#zebrafish-thismotaxis-assay>

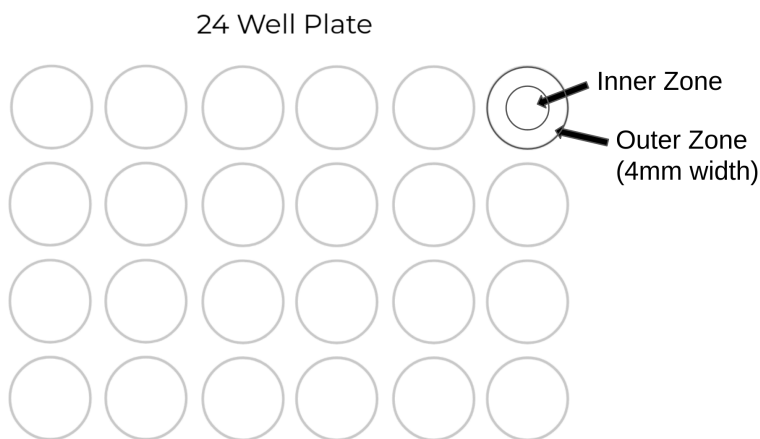


Fig 56: 24-well plate schematic.

Assay Output:

Fraction of frames confident per well.

Fraction time spent in the outer zone per well.



FISH LENGTH

A function for computing fish length is included in the MCAM Zebrafish Analysis package. Five skeleton segments are defined using the eight key point tracking model and the lengths of these segments are computed using euclidean distance for each frame. The median value of each segment across all frames is selected as the computed length for each segment in order to filter out extreme values that may exist from one frame of faulty tracking. The five (5) segments of interest here (Fig 57) are the:

- snout - to - center
- center - to - between_center_and_mid
- between_center_and_mid - to - mid_tail
- mid_tail - to - between_mid_and_caudal
- between_mid_and_caudal - to - caudal_fin

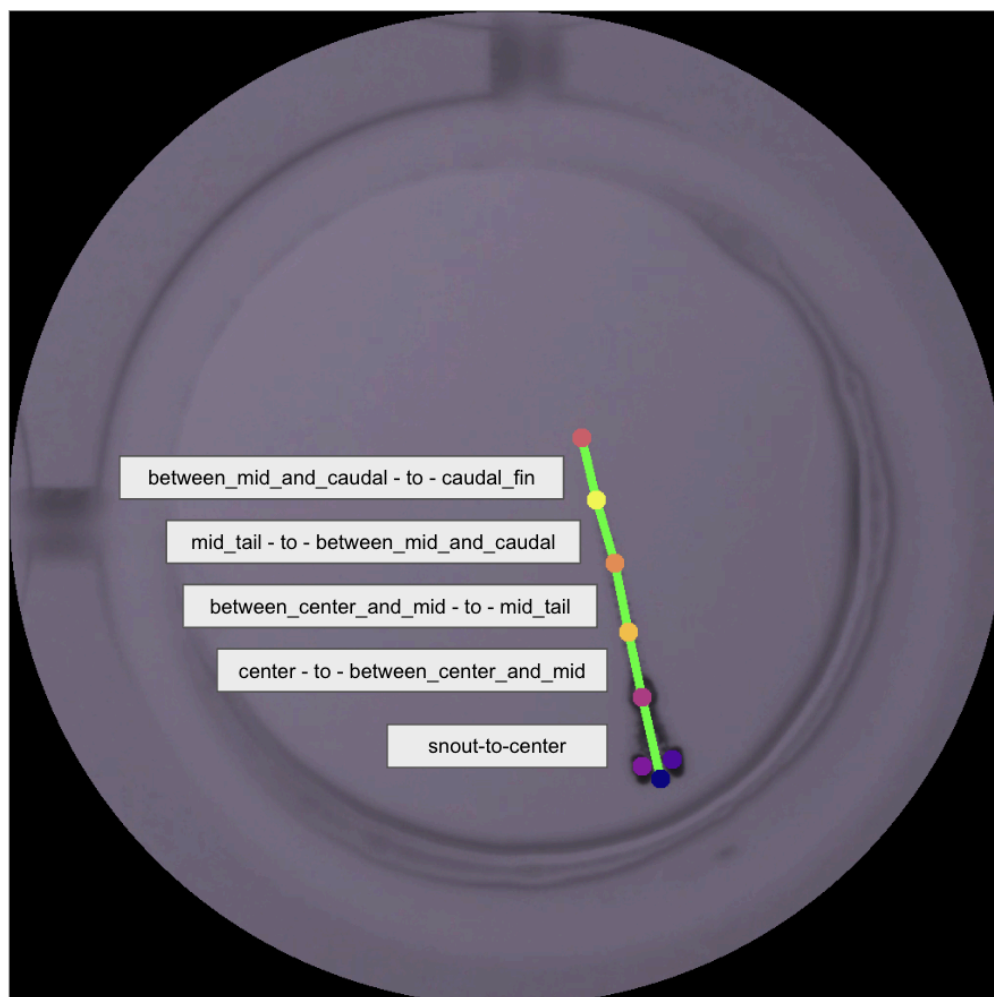


Fig 57: Fish length segments diagrammed on a specimen.

The head length is defined as the first of these segments (snout-to-center).

The tail length is defined as the sum of the second through fifth segments spanning from the center point to the caudal fin.

Outputs for this computation on a per well basis include:

- Fish Length
- Head Length
- Tail Length

Example output fish_lengths.csv (Fig 58):

	A	B	C	D	E
1	row	A	A	A	A
2	column	1	2	3	4
3	fish_lengths				
4	fish_length	0.00319271914969561	0.00345900251733039	0.00302840482747702	0.00333807841558264
5	head_length	0.000706766798948251	0.0008493974507506	0.00067947291088219	0.000800122312161942
6	tail_length	0.00248595235074736	0.00260960521138763	0.00234893198899875	0.00253795603101677
7					
8					

Fig 58: Example csv of fish lengths.

SKELETON ANALYSIS

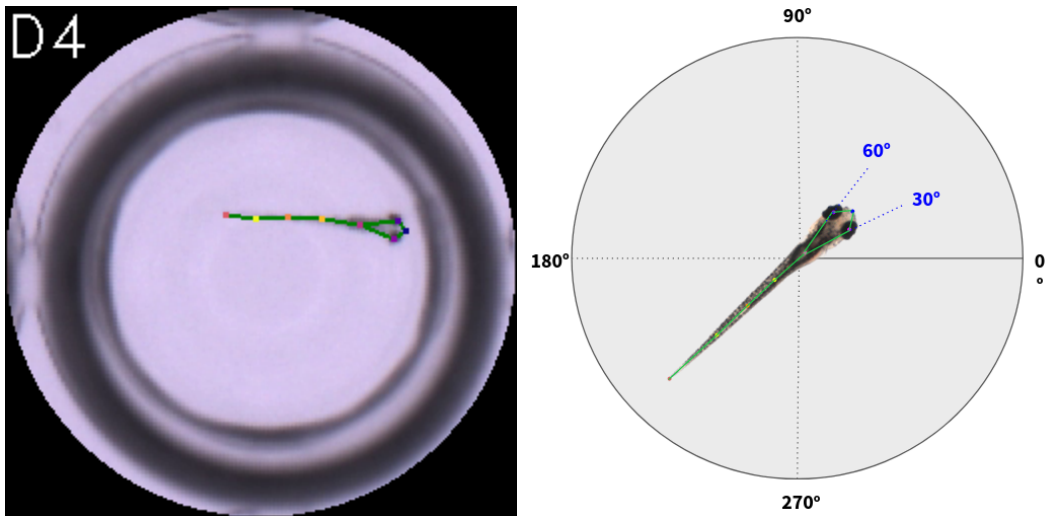


Fig 59: Skeleton analysis overlay example and orientation diagram.



The zebrafish “skeleton” refers to the line segments connecting each of the key points of the fish. Potentially interesting values include the length of each of these segments and the segment angle within the image frame (Fig 59). The skeleton is automatically overlaid on each fish in the “composite_video.mp4” results file and can be used to qualitatively validate tracking accuracy visually in addition to potential analysis use.

row	A		A		A		A		A		A	
column	1		1		1		1		1		1	
skeleton_segments	snout_L_eye		snout_L_eye	snout_L_eye	snout_R_eye		snout_R_eye	snout_R_eye	L_eye_center		L_eye_center	
skeleton_parameters	length		orientation	likelihood	length		orientation	likelihood	length		orientation	
time												
0	0.000167822		306.51505	0.99992716	0.00019055085		210.77898	0.9997435	0.00055632746		249.89345	
0.00833266	0.00018655944		305.3795	0.9999535	0.0001682418		225.25974	0.9997274	0.0005448785		251.39401	
0.016665321	0.00018775568		304.42078	0.9998976	0.00016042321		227.93529	0.9993975	0.00054826593		251.6475	
0.024997982	0.00020886432		308.1435	0.9999385	0.00017919637		225.98944	0.9997347	0.0006118647		250.72958	
0.033330643	0.00020114404		306.72565	0.9999441	0.00016762139		230.32314	0.9997192	0.0005735198		251.6355	
0.041663304	0.000205851		306.53952	0.9999647	0.0001701626		226.84749	0.9998816	0.00058060355		250.19803	
0.049995965	0.00019881657		309.12616	0.9999484	0.00016406094		229.48671	0.9998529	0.0005866663		252.0717	
0.058328626	0.00019412277		311.927	0.99997497	0.000159417		230.15408	0.9999347	0.0005693245		252.65105	
0.066661286	0.00019719578		310.30228	0.99995875	0.0001644481		231.84258	0.9998739	0.0005736228		253.73679	
0.074993947	0.00019903985		309.38858	0.9999808	0.00016689155		231.45796	0.9999418	0.00057154987		252.46735	

Fig 60: Example skeleton data csv.

The “skeleton_data.csv” results file contains segment measurements of each skeleton segment with lengths defined in meters and angles (orientation) defined in degrees with counter-clockwise rotation from zero (3 PM on a clock) such as the unit circle (Fig 60).

IMPORTANT NOTES:

As of August 22, 2023 and owl software version 0.18.316, skeleton segment angles are defined with counter-clockwise rotation from zero similar to the unit circle. Datasets acquired with previous owl versions have angles defined with clockwise rotation. Owl software version is reported in the header of all output files by default.

There is a 90 degree rotation between the acquisition and display of images on most Kestrel systems. All tracking metrics are computed with the acquisition coordinate frame and thus there is likely a 90 degree difference between zebrafish images and angles reported in results files. Please see “[Imaging Orientation](#)” for more information.

If you have any further questions regarding these points, please contact us at help@ramonaoptics.com.



TAIL BEND ANALYSIS

Tail bend angles to each of the four tail key points are calculated and reported for each well along with a tracking confidence. Tail angles are defined as the deflection angle from the fish body axis pointing towards the tail (Fig 61). The two vectors defining each angle are formed from the snout-to-center point vector and the center-to-tail keypoint vector for each of the four key points. Deflections to the fish's left are positive and to the right are negative (Fig 62, Fig 64).

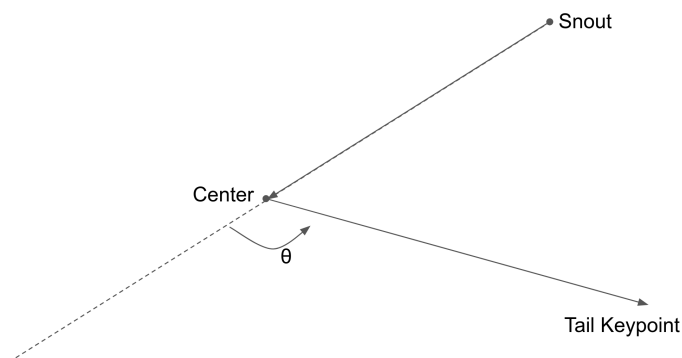


Fig 61: Tail bend analysis orientation diagram.

The output for tail bend analysis is given in the "tail_angle.csv" file. For many experiments plotting just the "caudal fin" angle may be sufficient for tail angle visualization such as below (Fig 63).

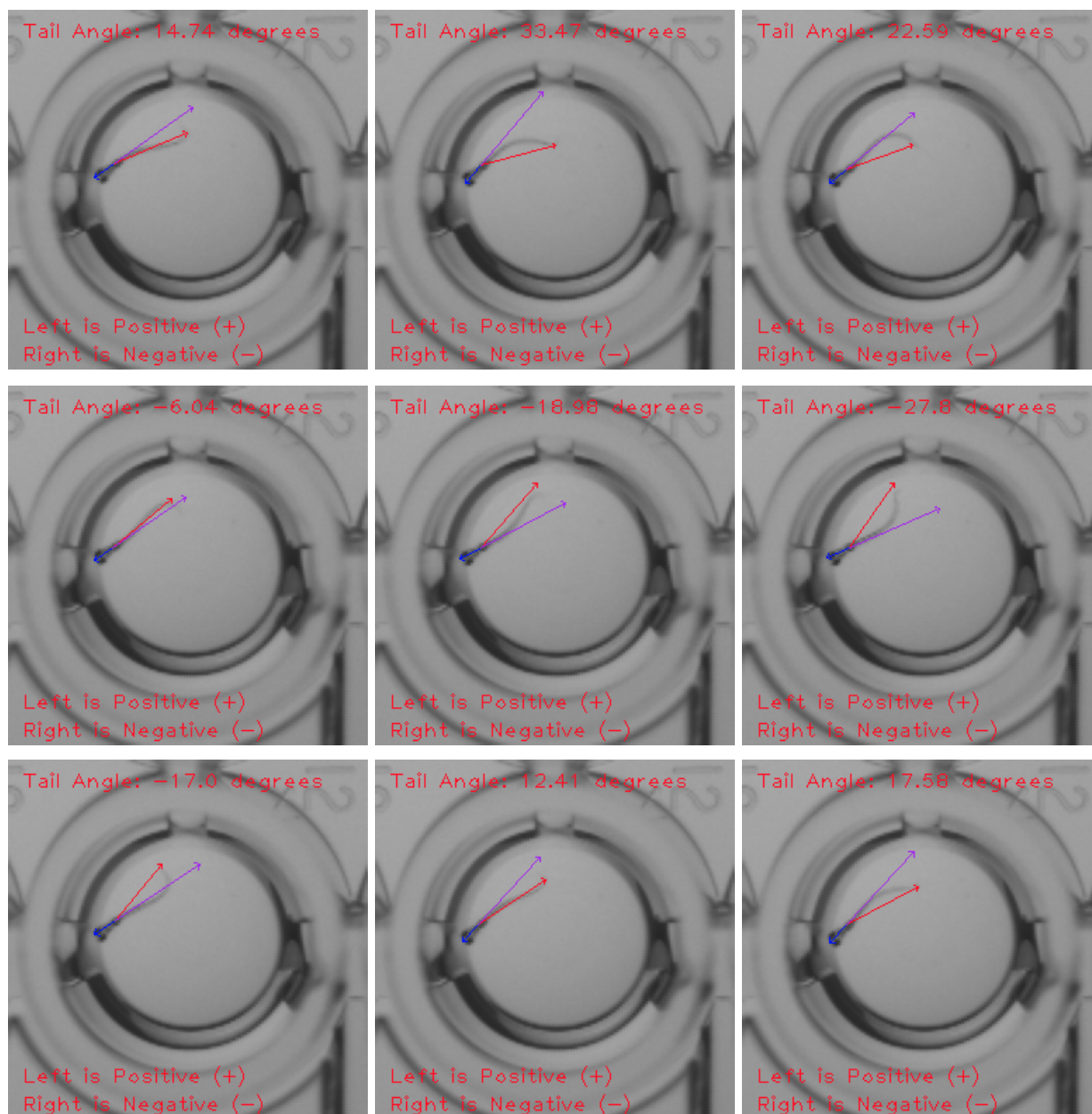


Fig 62: 160 fps acquisition, high frame rate (Bin 4) imaging mode, 2ms exposure. Blue is the center-to-snout vector, purple is the body axis pointing to the opposite direction, and red is the center-to-caudal fin vector.

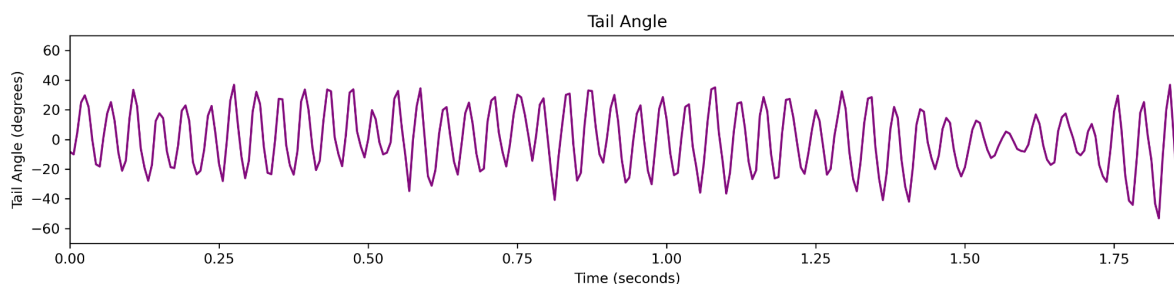


Fig 63: Example fin angle plot.

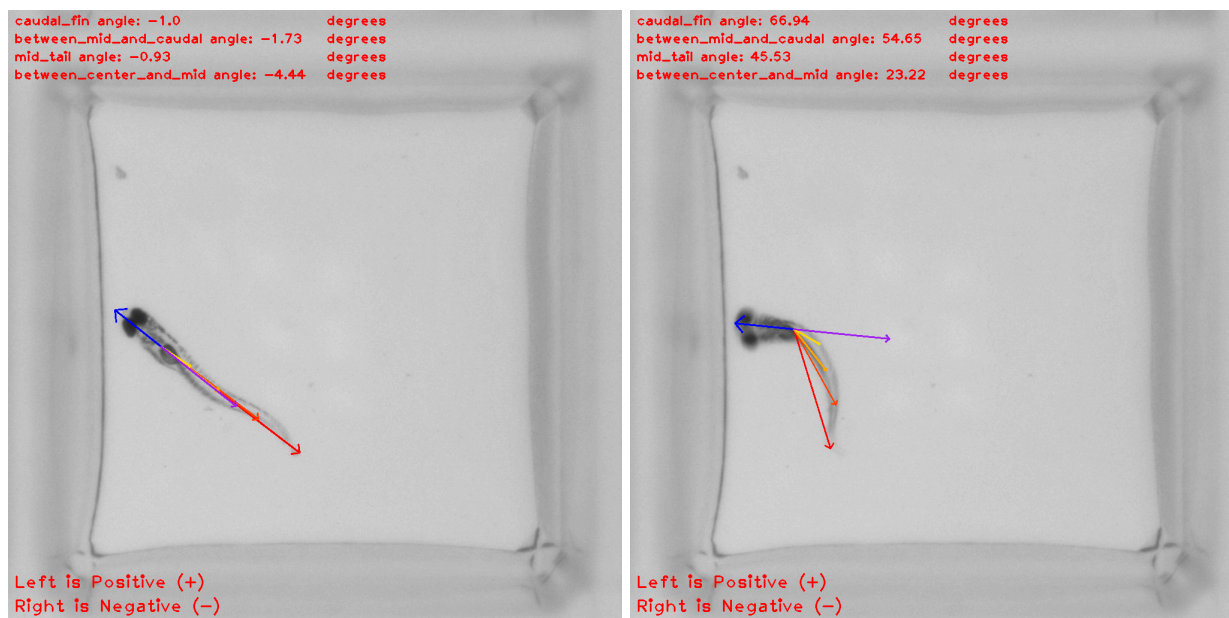


Fig 64: 30 fps acquisition, high resolution (Bin 1) imaging mode. Diagram of four key points used for tail analysis.

The four key points used for tail analysis are the "caudal_fin", "mid_tail", "between_center_and_mid", and "between_mid_and_caudal" which exist in the tracking dataset in that order (Fig 65). Tail information can be accessed in python by opening the "tracking_metadata.nc" dataset and accessing "tracking_dataset.tail_information". An example "Tracking Dataset - Data Analysis" noting how to access this data for analysis is given [HERE](#).



	A	B	C	D	E	F	G
1 row	A	A	A	A	A	A	A
2 column	1	1	1	1	1	1	1
3 tail keypoints	caudal_fin	caudal_fin	mid_tail	mid_tail	between_center_and_mid	between_center_and_mid	
4 tail_parameters	angle	likelihood	angle	likelihood	angle	likelihood	
5 time							
6	0	27.862060546875	0.999464333057404	11.4173889160156	0.99946355819702	6.749755859375	0.99946355819702
7	0.00833266	16.1387329101563	0.999912977218628	14.928466796875	0.999964356422424	7.45391845703125	0.999964356422424
8	0.016665321	21.91845703125	0.999897599220276	14.2645568847656	0.999897599220276	5.9500732421875	0.999897599220276
9	0.024997982	27.1479187011719	0.999639511108398	12.3472900390625	0.999953627586365	5.82742309570313	0.999953627586365
10	0.033330643	17.7627563476563	0.999951601028442	14.4105834960938	0.999951601028442	9.187255859375	0.999951601028442
11	0.041663304	19.4642639160156	0.99960070848465	17.7007141113281	0.999974966049194	9.74789428710938	0.999974966049194
12	0.049995965	21.74072265625	0.999964356422424	14.2489318847656	0.999964356422424	7.65878295898438	0.999964356422424

Fig 65: Example csv of tail angles output.

EYE ANALYSIS

Automated eye analysis for zebrafish using MCAM provides detailed morphometric information about zebrafish eye structures. The analysis uses key point tracking data and yields the location of each eye within the image frame, fish heading, distance between eyes, area of each eye, and the angle of each eye with respect to the fish body axis. For the complete analysis it is important to process data acquired at high (Bin 1) or medium (Bin 2) resolutions to ensure accurate analysis of elliptical eyes, as high frame-rate (Bin 4) data only provides the distance between the eyes. This is because at high frame-rate eyes appear more circular than elliptical and thus major and minor axes of the ellipse cannot be accurately determined consistently.

The analysis algorithm consists of several steps:

STEP	ACTION
1	Image Cropping: Using the location of the tracked center key point an area is cropped from each raw image around each fish to reduce computation of the next steps and exclude unnecessary search space.
2	Image Pre-processing: The cropped images are preprocessed using an adaptive thresholding algorithm to enhance contrast and remove noise, facilitating more accurate segmentation of eye structures.
3	Eye Segmentation: A contour-detection method is used to delineate the eye boundaries and find the centroid of each eye within the cropped image frame.
4	Eye Spacing: This distance between eyes is computed in units of pixels.
5	Fish Heading: The fish heading is computed as the angle of a line with respect to the overall image frame drawn perpendicular to a line connecting the two eye centroids the points away from the center point of the fish.
6	Eye Feature Extraction: The algorithm calculates the area of each eye in units of pixels, and the angle of each eye with respect to the fish heading.
7	Coordinate Mapping: The eye centroid coordinates in the cropped frame are mapped back to the coordinate frame of the original image.



-
- | | |
|---|--|
| 8 | Distance and area values are converted from units of pixels to units of meters and square meters using the calibrated pixel width of the MCAM. |
|---|--|
-
- | | |
|---|--|
| 9 | Results are output in a CSV file titled "eye_analysis.csv" in the results folder of the dataset being processed. |
|---|--|
-

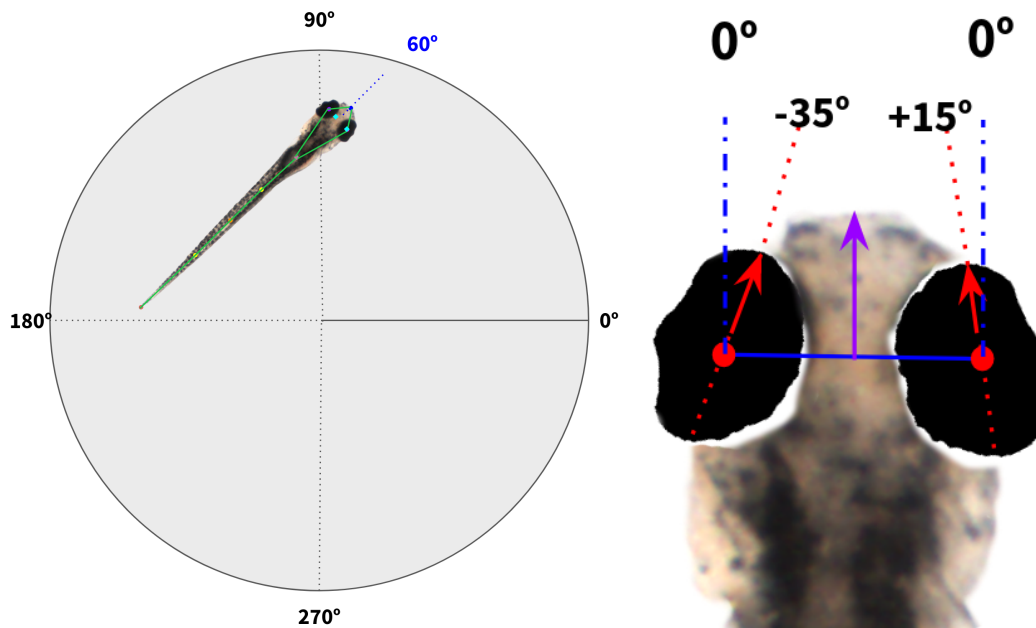


Fig 66: Description of head angles and eye angles.

LEFT

Heading angle / Orientation within a well. Zero degrees (0°) is at 3PM on a clock, similar to the unit circle, with counter-clockwise rotation increasing the heading angle. The heading angle of the fish is at sixty degrees (60°). Three hundred and sixty degrees does not exist and the angle range is (0° , 360°).

RIGHT

A solid blue line connects the eye centroids of the left and right eyes. The purple arrowed line represents the heading angle of the fish. Two blue dotted lines are parallel to the fish heading angle pointing in the same direction denoting the direction of the fish's body axis. Two red arrows display one half of the major axis of the eye region ellipses. Deflection of an eye to the left of the body axis is positive (+) rotation while rotation to the right of the body axis is negative (-).

7	row	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
8	column	1	1	1	1	1	1	1	1	1	1	1	1	1	1	2
9	eye_parameters	L_eye_centroid_y	L_eye_centroid_x	R_eye_centroid_y	R_eye_centroid_x	distance_between_eyes	L_eye_area	R_eye_area	fish_orientation	L_eye_angle	R_eye_angle	L_eye_centroid_y	L			
10	time															
11	0	-0.008107949	-0.011070198	-0.008248714	-0.011740469	0.00068489194	1.80E-07	1.89E-07	281.86047	-24.425888	9.842819	0.0101702055				
12	0.012498365	-0.008106091	-0.011070973	-0.008248749	-0.011737753	0.0006818681	1.81E-07	1.84E-07	282.07645	-23.811401	10.312225	0.0101702055				
13	0.02499673	-0.008103624	-0.011069432	-0.008246429	-0.011738672	0.00068430684	1.81E-07	1.86E-07	282.04535	-24.350647	10.306763	0.0101702055				
14	0.037495095	-0.008101419	-0.011070198	-0.008248749	-0.011737753	0.00068361766	1.80E-07	1.84E-07	282.4457	-25.894852	9.942978	0.010169508				
15	0.04999346	-0.008101374	-0.011071606	-0.008245106	-0.01174018	0.0006838493	1.78E-07	1.84E-07	282.13306	-25.404144	9.496933	0.010169461				
16	0.062491826	-0.008101374	-0.011071606	-0.008246429	-0.011738672	0.0006826553	1.78E-07	1.86E-07	282.2681	-25.539185	10.084015	0.010168764				
17	0.074990191	-0.008101374	-0.011071606	-0.008247536	-0.011740787	0.000684957	1.78E-07	1.84E-07	282.32092	-25.592026	9.639954	0.0101702055				
18	0.087488556	-0.008101374	-0.011071606	-0.008246429	-0.011738672	0.0006826553	1.78E-07	1.86E-07	282.2681	-25.539185	10.084015	0.010169508				
19	0.099986921	-0.008101374	-0.011071606	-0.0082465345	-0.011740175	0.0006841465	1.78E-07	1.87E-07	282.25	-25.521088	10.411102	0.010169508				
20	0.112485286	-0.0082465345	-0.011740175	-0.008101374	-0.011071606	0.0006841465	1.87E-07	1.78E-07	282.25	10.411102	-25.521088	0.010169508				
21	0.124983652	-0.008101374	-0.011071606	-0.008242949	-0.011742267	0.00068544084	1.78E-07	1.87E-07	281.92004	-25.191147	9.683777	0.010169508				
22	0.137482017	-0.008101374	-0.011071606	-0.008245225	-0.011741683	0.000685344	1.78E-07	1.86E-07	282.1162	-25.387299	9.807175	0.010169508				
23	0.149980382	-0.008101374	-0.011071606	-0.008242949	-0.011742267	0.00068544084	1.78E-07	1.87E-07	281.92004	-25.191147	9.683777	0.010168042				
24	0.162478747	-0.008101419	-0.011070198	-0.008240706	-0.011743188	0.0006872523	1.80E-07	1.86E-07	281.69333	-25.142487	9.874344	0.01016764				
25	0.174977113	-0.008103624	-0.011069432	-0.008245225	-0.011741683	0.0006870025	1.81E-07	1.86E-07	281.89468	-24.199997	10.028687	0.0101698255				

Fig 67: Example csv of eye angles output.

Fig 67 is an example of “eye_analysis.csv” with data on a per well basis. Each row displays one time point, given in seconds. Locations and distances are displayed in units of meters with reference to the well plate and the reference frame origin at the center of the well. Area is displayed in units of square meters. Angles are measured in degrees. Orientation is measured with zero at 3 PM on a clock similarly to the unit circle. There is likely a 90 degree rotation between how images are acquired and displayed so this orientation may not immediately be apparent. Please see the section on EXIF rotations for more information. Eye angles are measured with respect to the fish heading with a positive sign representing a deflection to the left of the body axis and a negative sign representing a deflection to the right.

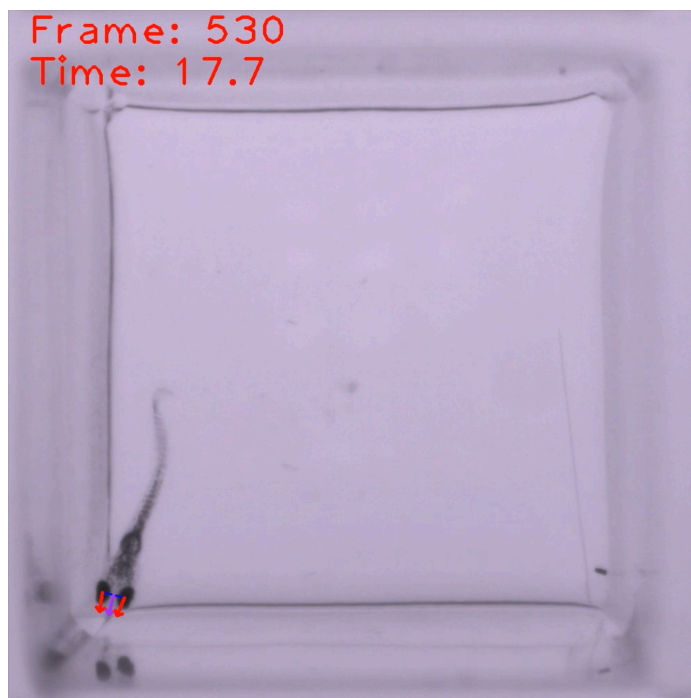


Fig 68: Visualization of High Resolution (Bin 1) data acquired from a 96-well plate at 30 frames per second: The blue line drawn represents the distance between eye centroids, the purple arrowed line is the fish heading, and the two red lines are the direction of each eye respectively. The frame number and time in seconds are displayed for context within the overall experiment.



EYE ANALYSIS IMAGING PARAMETERS

Eye analysis has been optimized for specific imaging parameters using brightfield visible spectrum illumination:

- Standard resolution (Bin 2)
- Illumination: Visible spectrum transmission
- Exposure: 10 ms
- Brightness: 25%
- Digital gain: 1.0
- Analog gain: 1.0

If your eye analysis requires other imaging parameters, please contact us at help@ramonaoptics.com for assistance.

Tracking Data Filtering

Tracking data filtering is divided into two steps: **Anomaly Detection**, which finds errors in raw tracking data, and **Denoising**, which removes pixel jitter between series of frames. Anomaly detection is applied prior to denoising.

The “Filter Anomalies” and “Denoise / Smooth” checkboxes under **Advanced Settings** in the **Animal tracking panel** are enabled by default. To filter and reanalyze previously tracked data, check the corresponding box shown in the screenshot below (Fig 69). Note: Reanalyzing data will overwrite the previous analysis files (however, raw tracking data is always stored in raw, unfiltered form).

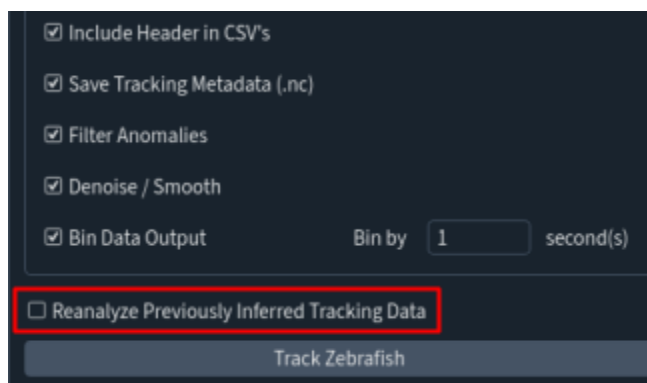


Fig 69: Animal tracking panel data reanalyzing option.



ANOMALY DETECTION

Anomaly detection finds and removes large errors in tracking using criteria to identify situations that are not physically possible. Data points found to be erroneous are removed and these regions are filled by linear interpolation based on the surrounding regions. Four filters are used in combination: boundary filtering, speed thresholding, center-of-mass filtering, and outlier window filtering.

Boundary filtering sets a boundary within each image based on the frame centroid and well radius. Coordinates outside of this boundary are replaced with null values and these empty regions are filled by linear interpolation after all anomaly detection filters have been applied (Fig 70).

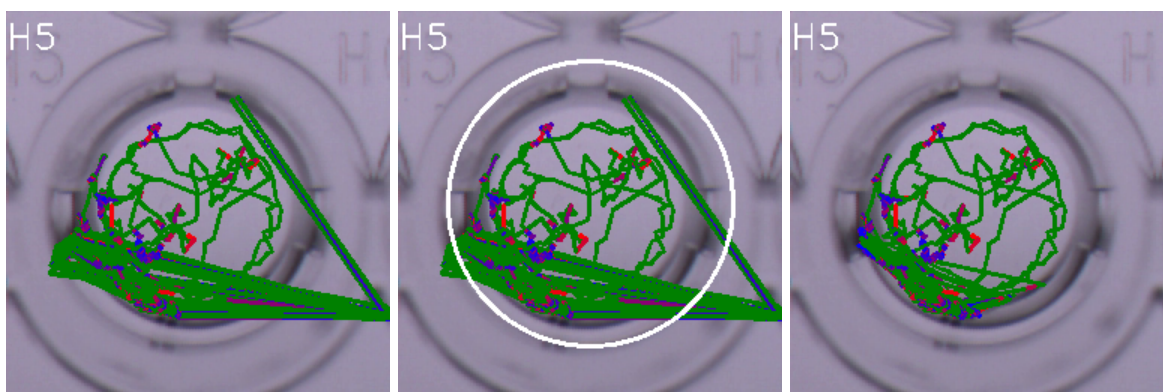


Fig 70: Example of boundary filtering of a 96 well plate. (Left) Erroneous tracking occasionally jumps outside of a well. (Center) The predefined boundary is visualized on the well in white. (Right) Filtered data removes tracking data that jumps outside of the well.

Speed thresholding defines a maximum speed that is possible for zebrafish and values above this threshold are removed. The speed threshold is set at 120 mm/sec above which we believe is not possible for zebrafish larvae. Data points resulting in a speed greater than this threshold are replaced with null values and these empty regions are filled by linear interpolation after all anomaly detection filters have been applied. Following interpolation movement metrics are recomputed. This can result in speed values above the threshold value, however when we quantify the number of these values, we find ~135x decrease in abnormal speed values.

Center-of-Mass (CoM) filtering computes the average location of key points in each frame and if any key point is greater than 0.7 times the length of the fish away from the CoM, the point is removed. In the GUI workflow fish length is computed prior to filtering so that the length measurement can be used in CoM filtering, and then the fish length is computed again after filtering is complete. (Fig 71) Fish length must be computed prior to CoM filtering. Eye keypoints are not considered in this computation because if included, the CoM is biased towards the head of the fish.

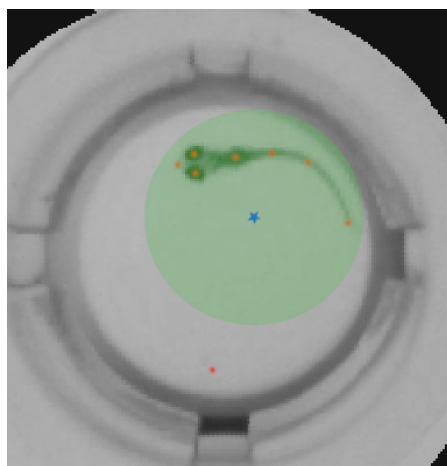


Fig 71: Artistic rendition of Center-of-Mass (CoM) computation displaying zebrafish key points in orange and the computed CoM as the blue star. The green circle indicates the region in which key points could reasonably reside. Note: Eye key points are not included in the CoM computation.

Outlier window filtering is applied after all other anomaly detection filters and removes data that the deletions of previous filtering has indicated to be likely to be erroneous. Outlier window filtering finds regions of good data bounded by null values (null values are only included in a dataset when anomaly filtering deletes data) and removes each of these regions if each single region is shorter than four (4) frames.

It is important to note that raw tracking data is always stored in the "tracking_metadata.nc" file in original form to ensure that it is unaltered for future re-analysis. When **Filter Anomalies** are checked in the **Animal tracking panel**, computations stored in the NC file, such as skeleton, tail, length, and eye information, are based on the data post anomaly detection. Anomaly detection must be applied each time the dataset is loaded for data manipulation because the data is stored in raw form (**the filtered dataset is never stored**).

DENOISING

Denoising attempts to remove jitter from keypoints placed on sequential frames of data. At low frame rates and short acquisition durations this jitter is less important, however at high frame rates and long acquisition times the accumulated distance computed between these pixel movements will greatly outweigh the signal of interest.

Ramona's wavelet denoising algorithm estimates the variance of each keypoint and then applies wavelet denoising using the "sym4" wavelet. Examples of raw and smoothed distance traveled data for both instantaneous and cumulative distance traveled are given below (Fig 72).

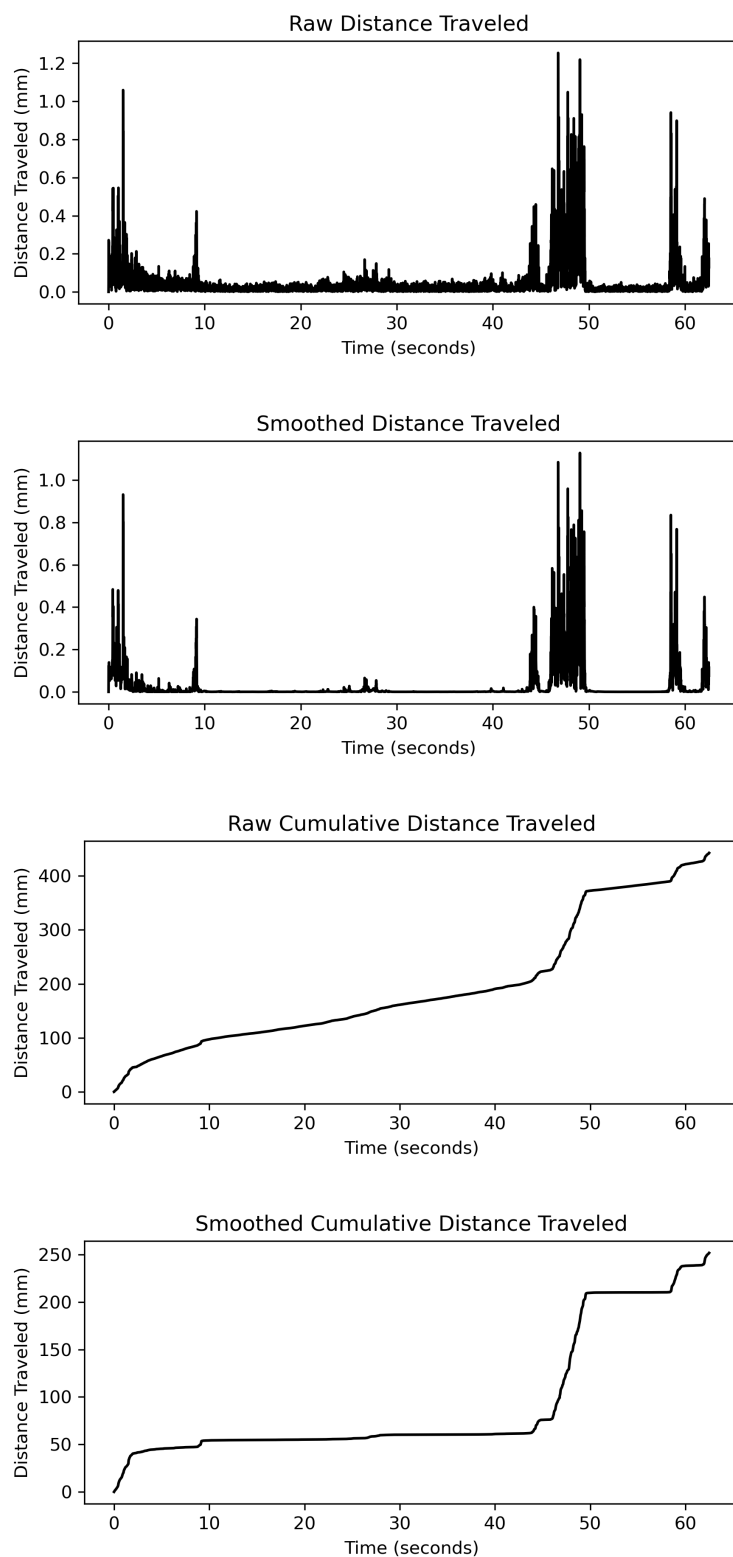


Fig 72: Examples plots before and after denoising application.



Filtering has been optimized for the zebrafish center key point and should only be used for key points of the head which move at similar frequencies. The tail key points for example move at higher frequencies and should not be filtered with this strategy. Smoothing key points that move at higher frequencies has the potential to significantly change collected data and introduce artifacts, as can be seen in the example below. This example shows smoothed tail key points (Top) compared to raw (Bottom) (Fig 73). Notice that smoothing in this case changes the data significantly throughout as well as introduces at least one rather extreme artifact highlighted by the red arrow. We therefore choose to not smooth tail angle data. This is acceptable because jitter error does not accumulate over time in the movement of these key points. Smoothing of all key points is accessible through our API. Please contact us for further assistance as this workflow will be workflow-specific.

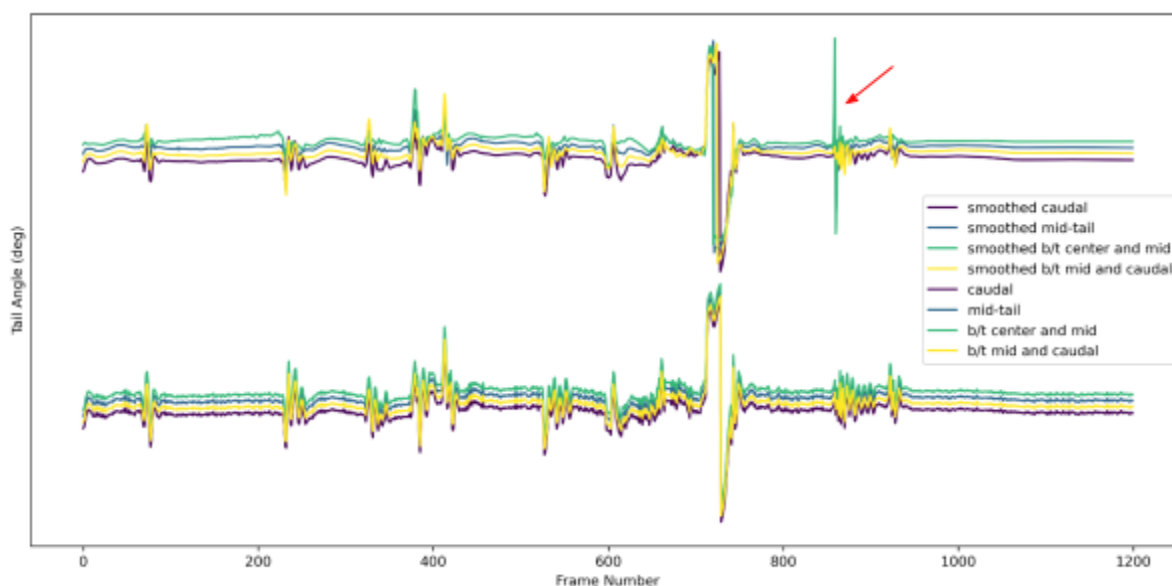


Fig 73: Example of how denoising tail angle data can introduce artifacts.

Denoising is implemented by default and can be disabled by unchecking "Denoise / Smooth" in the animal tracking panel. **It is important to note that the algorithm is only applied during the distance traveled computation so distance traveled and speed are the only output metrics affected.** All tracking data is stored in raw form both in output CSVs and in the "tracking_metadata.nc" file so that we can ensure it is unaltered for future re-analysis and thus denoising must be applied each time the dataset is loaded for data manipulation.

TIME BINNING

Tracking data output in CSV format is often massive in scale, making it unusable. Traditional CSV softwares including Microsoft Office Excel and LibreOffice Sheets have limits on the number of rows and columns that can be displayed, and high frame rate, long duration recordings will often surpass these limits in which case the CSV's cannot be opened in their entirety.



Tracking data is thus binned by default in one second increments. Unchecking **Bin Data Output** will disable this feature and yield raw outputs. The time bin increment can be changed in the field to the right of the **Bin Data Output checkbox**. All output data is binned by this time increment. Distance traveled data is summed within each time period while all other data is averaged. Time bin increments are limited to one decimal place so 0.1 second is the smallest bin allowed. If the duration is not evenly divisible by the user requested time bin, the last bin will not be full and it will be removed automatically from the output data to ensure that all bins are of equal importance. We recommend choosing time bins that divide the total duration evenly.

It is important to note that the validity of binned data depends on interpretation of each specific experiment's requirements. It is likely that binned tail angle and eye angle data which is averaged over one second has less validity than the native resolution and thus we recommend not binning data where high time resolution is required such as with these high frequency measurements.

For other questions, please contact us at help@ramonaoptics.com.

Zebrafish Segmentation

Image segmentation based workflows have been integrated into the MCAM Viewer software for use with zebrafish. The basis of this technique locates a zebrafish within each well and defines a region of interest highlighting the fish within the image. Once the fish region is defined, analyses within this region can be performed such as cell counting, fluorescence quantification or area measurements. For more information on the underlying mechanics of this workflow, please see our publication in PLoS One (2023) titled [Automated, high-throughput quantification of EGFP-expressing neutrophils in zebrafish by machine learning and a highly-parallelized microscope.](#)

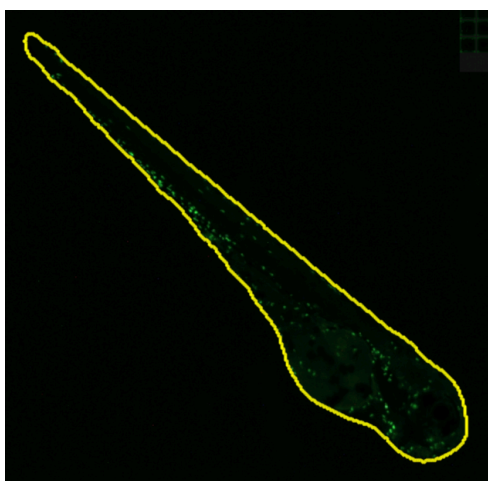


Fig 74: Zebrafish expressing Lyz:EGFP segmented from the background. The defined region of interest is displayed with a yellow line.



This type of analysis can be customized for many use cases that involve analysis within a defined region. For example, rather than segmenting the entire fish, it is possible to segment just the eyes or yolk sac in order to measure morphological areas. For more information on extended use cases and customization, please contact us at help@ramonaoptics.com.

SEGMENTATION GUI PANEL

The segmentation panel within the MCAM Viewer can be opened from the Assays > Zebrafish menu (Fig 75).

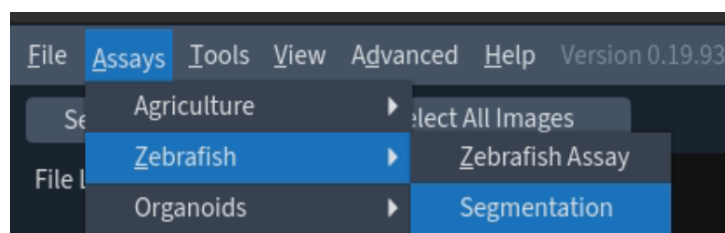


Fig 75: MCAM Viewer Panel for Segmentation

Segmentation analysis is initiated by selecting a segmentation model from the "Model Selection" menu and then clicking "Segment Instances". The base segmentation analysis simply determines a region of interest for each well. Additional analyses within the region of interest can be added by selecting the respective check box in the "Analysis Options" section.

Input datasets to the segmentation workflow can be either a single frame of a well plate or well plate subset or a Z-stack acquisition which has multiple frames per well with varying levels of focus. If a stack dataset is found, automatic focusing is applied before the final segmentation which selects the most in-focus frame for each well. A new dataset is output to the results folder containing only the best-focus frames.

The image shows a 'Segmentation' panel with the following sections:

- Segmentation Analysis Protocol:** Contains 'Load' and 'Save' buttons.
- Model Selection:** Includes a dropdown menu and a 'Load Custom Model' button.
- Batch Analysis:** A checkbox that is currently unchecked.
- Choose Default Settings:** A button.
- Segmentation Options:** Contains a checked checkbox for 'Filter Regions'.
- Analysis Options:**
 - 'Compute Region Areas' is unchecked.
 - 'Units' is set to 'Millimeters'.
 - 'Pixel Intensity Threshold' is set to '55'.
 - 'Color Channel' is set to 'Green'.
 - 'Compute Fluorescence Intensity' is unchecked.
 - 'Count Blobs' is checked, with sub-options:
 - 'Minimum Blob Radius (μm)' set to '13.1'.
 - 'Maximum Blob Radius (μm)' set to '65.5'.
- Export Settings:**
 - 'Combine CSV Results' is checked.
 - 'Export Masked Images' is unchecked.
 - 'Export Regions' is unchecked.
 - 'Export Annotations' is unchecked.

At the bottom of the panel is a 'Segment Instances' button.

Fig 76: Detailed segmentation panel.

SEGMENTATION OPTIONS

FILTER REGIONS

This option, selected by default, filters all segmented regions and returns only the largest region identified.

ANALYSIS OPTIONS

COMPUTE REGION AREAS

When selected, an area value is computed for each segmented region. This value is reported in square SI units defined in the panel above. Accurate pixel width calibration of the MCAM is required as this value is used to compute areas.



COMPUTE FLUORESCENCE INTENSITY

When selected, a bulk fluorescence value is computed for each region following a pixel intensity threshold. The threshold is applied setting any pixel values within the region of interest below the pixel intensity threshold (user-defined above) to zero while pixel values equal to or greater than the threshold value remain unchanged. Only one color channel is considered in this analysis as defined by the selected color channel just above in the segmentation panel. Pixel values within the region are summed and reported with arbitrary units (AU).

COUNT BLOBS

When selected, blobs are counted within each region of interest following a pixel intensity threshold. A blob is a round-like group of pixels which can be extended to fluorescent cells, chromatophores or other such entities. The pixel intensity threshold is applied, setting any pixel values within the region of interest below the pixel intensity threshold (user defined above) to zero while pixel values equal to or greater than the threshold value remain unchanged. Only one color channel is considered in this analysis as defined by the selected color channel just above in the segmentation panel. Minimum and maximum blob size thresholds are defined to limit computation when searching for blobs.

EXPORT SETTINGS

COMBINE CSV RESULTS

When selected, all output metrics are displayed in one CSV file with a column for each individual metric and row for each well. This setting is selected by default, making it easier to visualize multiple endpoint metrics together. If this setting is deselected, each metric is displayed in a separate CSV file in an arrayed format similar to a well plate layout.

EXPORT MASKED IMAGES

When selected, an image is exported for each well with the segmented region overlaid on top of the image. Overlays are represented by a yellow line defining the outer perimeter of the region of interest.

EXPORT REGIONS

When selected, segmented regions are exported as numpy arrays which can be used in further analysis. Each array consists of one channel with the same height and width as the segmented region. Most values of the array are zeros, with ones in region locations. This is considered an advanced feature.



EXPORT ANNOTATIONS

When selected, cartesian coordinates for each segmentation mask are exported along with individual images. Annotations are exported in vgg16 data format and can be used for training and fine-tuning additional segmentation models. This is considered an advanced feature.

Analysis Computation Explained

Explanations of how analysis metrics of interest are calculated. We are constantly striving to improve our methodologies and depend on feedback from our collaborators and customers to continue this optimization.

For questions and feedback, please contact: help@ramonaoptics.com

UNITS

Unless otherwise specified, exported units implicitly:

- Distance or length: meters
 - Time: seconds
 - Velocity or speed: meters/second
 - Orientation/Angle: degrees
-

DISTANCE TRAVELED

1 Location values (x, y) below a confidence threshold are removed and the resulting gaps are filled by linear interpolation between the previous and following confident points. By default, this threshold is set to 0.1 as of owl version 0.18.100.

2 The percentage of the total frames deemed non-confident are recorded to be provided to the user in the file.

3 The positions, x and y are then filtered with a moving average of length

$$length = \max(\lfloor \frac{frame\ rate}{60} \rfloor, 1)$$

$$amplitude = \frac{1}{length}$$

a) Where $\lfloor \cdot \rfloor$ denote the integer flooring operation.

4

The euclidean distance between each key-point in two adjacent frames is calculated with units of meters (Fig 77).

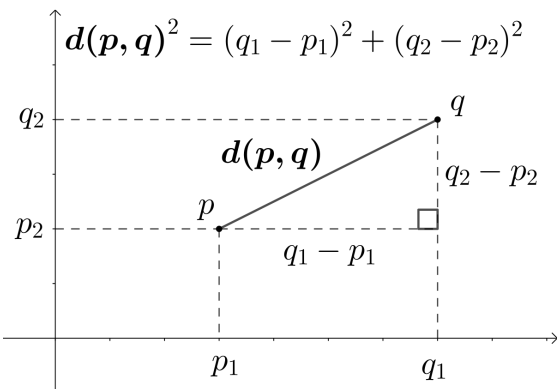


Fig 77: Diagram of euclidean distance.

- 5
- The first frame is assigned a distance traveled of zero because movement cannot be calculated for this frame.
- 6
- Total distance traveled is the sum of the distances traveled between each frame.

SPEED

- 1
- The sampling period for each frame of an acquisition is defined to be the inverse of the frame rate.
- 2
- Speed is computed to be the distance traveled between two adjacent frames (see above) divided by the sampling period. Speed is computed for every frame in a video.
- 3
- Peak speed is the maximum speed computed over all frames using the above methodology.
- 4
- The first frame is assigned a speed of zero because movement and thus speed cannot be calculated for this frame.

Stitching Images Through MCAM Viewer



Fig 78: stitching panel.

- 1 With your desired single frame dataset loaded into the MCAM Viewer navigate to File > Export Image (as stitched).
- 2 The stitching panel will pop up on the right side of the viewer (Fig 78).
- 3 Select the output resolution of your stitched image.
- 4 Once resolution is selected, frame the viewer to the FOV you would like your stitched image to cover. You can navigate this viewer the same as in the MCAM Viewer and MCAM.



-
- 5 To quickly fit the FOV to the full composite press the following button (Fig 79):

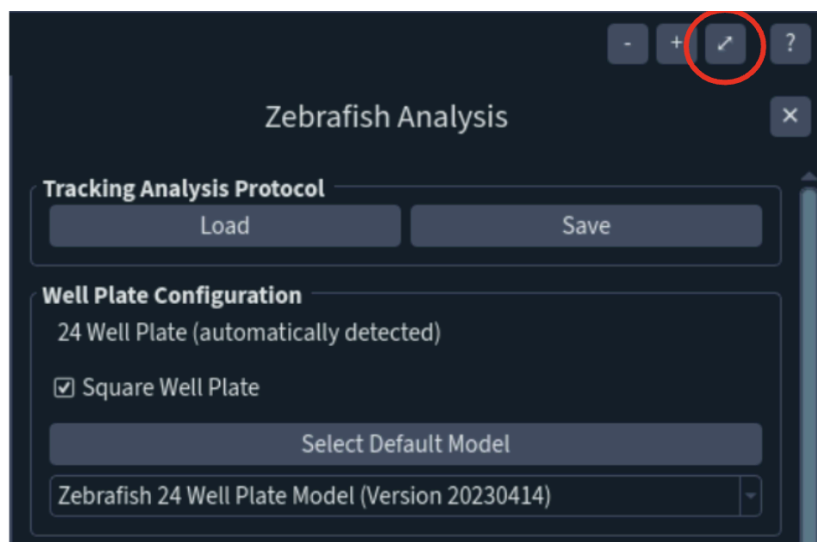


Fig 79: Full composite button.

-
- 6 Deselecting "Quick Stitch" will improve blending across the seams of the image, but will take a much longer time to perform the stitch.
-
- 7 If working with a stack (video, z stack, exposure stack, etc.) then selecting "Stitch Stack" will allow you to stitch multiple sequential frames between the given "Start Frame" and "End Frame".
-
- 8 Once the preview window is set, press the "Stitch" button. This will bring up a File Navigator that will allow you to select the filename of the stitched image.
-
- 9 A progress bar will appear to monitor the stitching and display the location of the stitched image.
-



Estimated File Size and Recording Duration for Video Acquisition

We provide the table below to help guide users through the different options available for recording video. For maximum data bandwidth, the data captured from the MCAM must be stored in RAM. The workstations provided with the MCAM contain between 128 GiB and 1024 GiB of RAM depending on the configuration that can be used for video acquisition. For data rates below 2 GiB/s, it is possible to use the provided SSDs instead of RAM to store the raw video files. For data rates below approximately 610 MiB/s, it is possible to compress the video during the recording to extend the recording duration given the same file save on the storage media.

WELL PLATE		HIGH RESOLUTION BIN x1	STANDARD RESOLUTION BIN x2	HIGH-SPEED BIN x4
96	MAXIMUM FPS	1024 × 1024 / well @ 30 fps	512 × 512 / well @ 60 fps	256 × 256 / well @ 160 fps
	96 GiB RAM Recording	2.8 GiB/s (30 sec)	1.4 GiB/s (65 sec)	0.93 GiB/s (100 sec)
	300 GiB Disk Recording	N/A	200 sec (3 mins, 20 sec)	320 sec (5 min, 20 sec)
	30 GiB mp4 Recording	N/A	30 fps, 400 sec (6 mins, 40 sec)	80 fps (10 mins, 40 sec)
24	MAXIMUM FPS	2048 × 2048 / well @ 30 fps	1024 × 1024 / well @ 60 fps	512 × 512 / well @ 160 fps
	96 GiB RAM Recording	2.8 GiB/s (30 sec)	1.4 GiB/s (65 sec)	0.93 GiB/s (100 sec)
	300 GiB Disk Recording	N/A	200 sec (3 mins, 20 sec)	320 sec (5 min, 20 sec)
	30 GiB MP4 Recording	N/A	30 fps, 400 sec (6 mins, 40 sec)	80 fps (10 mins, 40 sec)
48 ⁺	MAXIMUM FPS	1408 × 1408 / well @ 20 fps	704 × 704 / well @ 30 fps	352 × 352 / well @ 30 fps



	96 GiB RAM Recording	4.2 GiB/s (20 sec)	1.58 GiB/s (60 sec)	0.39 GiB/s (240 sec)
	300 GiB Disk Recording	N/A	440 sec (7 min, 20 sec)	30 min
	30 GiB mp4 Recording	N/A	N/A	30 min
8	MAXIMUM FPS	2304 × 2304 / well @ 30 fps	1152 × 1152 / well @ 60 fps	576 × 576 / well @ 120 fps
	96 GiB RAM Recording	1.18 GiB/s (80 sec)	1.58 GiB/s (60 sec)	0.39 GiB/s (240 sec)
	300 GiB Disk Recording	240 sec (4 min)	500 sec (8 mins, 20 sec)	1000 sec (16 mins, 40 sec)
	30 GiB mp4 Recording	N/A	500 sec (8 mins, 20 sec)	1000 sec (16 mins, 40 sec)

*RAM recordings for 48 well plates are required to capture the full 3072 x 3072 region of interest of each of the 24 micro cameras in the Kestrel. As such, RAM recordings are slightly less efficient than their counterparts that use the well plate alignment to reduce the data size when saving raw data to an .nc file or compressed data to an mp4.



System Calibration

All system calibrations are configured during initial setup. For further information on system calibration, please contact us at help@ramonaoptics.com.

Programming Interface and API

Higher-level functionality for the MCAM is provided through a Python® 3 module. This module is installable through Ramona Optics' Anaconda channel. An authentication token will be provided to you at the time of sale.

Once installed, the owl module can be imported using

```
import owl
```

Python API details are provided in the API manual: <https://docs.ramonaoptics.com/>



Accessories

Stage Inserts

A Multiwell Microplate Insert from ASI Imaging is included with the MCAM. Other inserts can be purchased from ASI Imaging at www.asiimaging.com and utilized according to your workflow. The MCAM stage accepts inserts with dimensions 160 x 110 mm.

Multiwell Microplate Insert - I-3020

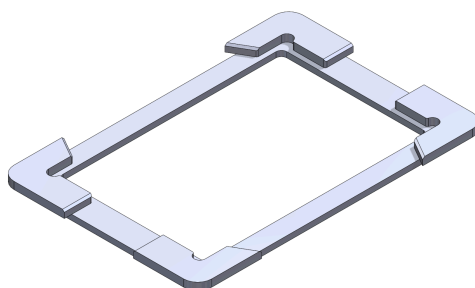


Fig 80: Multiwell Microplate Insert CAD render.

Insert Exterior Dimensions: 160 x 110 mm

Insert Interior Dimensions: 128 x 86.1 mm

Accepted Microplate Dimensions: 127.5 x 85 mm

Other Recommended Inserts

- 98mm Petri Dish Insert: I-3098
- 88mm Petri Dish Insert: I-3088
- 85mm Petri Dish Insert: I-3085

Vireo

Graphical User Interface (GUI)

Opening and Closing the MCAM

After installing the acquisition software and turning on the MCAM, the Graphical User Interface (GUI) can be started by clicking on the Ramona Optics application launcher on the sidebar. By default, the MCAM will attempt to connect to each of three components:

- The reflection illumination board,
- The transmission illumination board,
- The stage motor.

If any of these components are unavailable, the user can choose to not connect these components by deselecting their boxes in "Select Components" under the "Advanced" menu on the toolbar. This will automatically disable the ability to change the settings for these components.

The MCAM can be closed using the "Exit" button from the File menu, or by exiting the GUI with the "X" button in the upper right corner. Either method will stop all ongoing operations and safely close the MCAM.

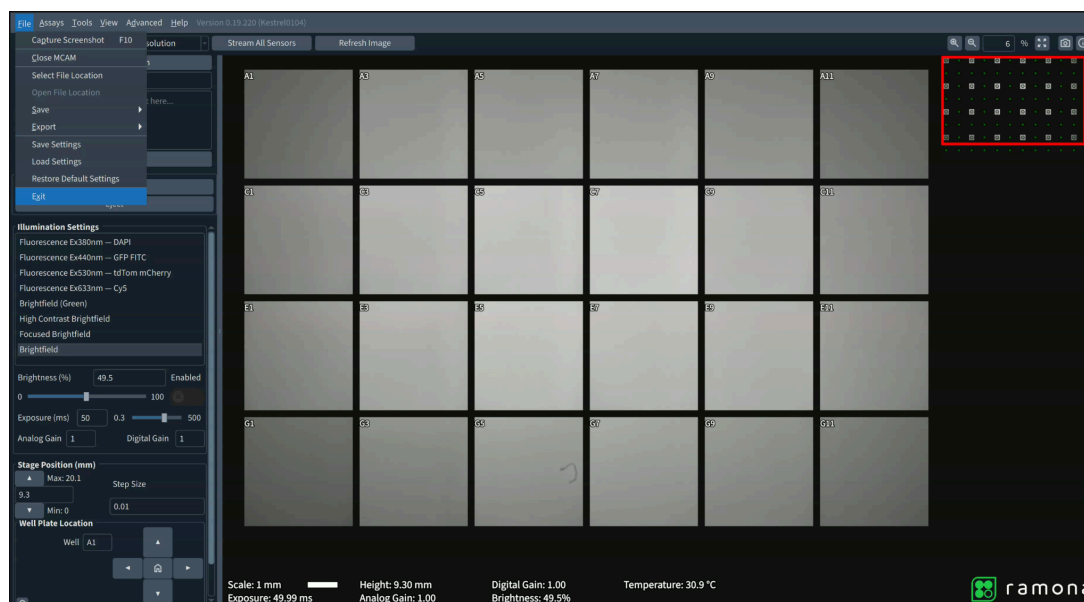


Fig 81: GUI start screen, with the "File" menu open and "Exit" highlighted.

GUI Navigation

The user can zoom and pan in the GUI to view any section of the streamed or acquired images. To zoom, either scroll up and down on the mouse wheel, or move the mouse while holding the right click mouse button. To pan, move the mouse in any direction while holding down the left click button.

At the top left of the window (Fig 82, #1) the current file location is displayed. This is the location where all files will be saved.

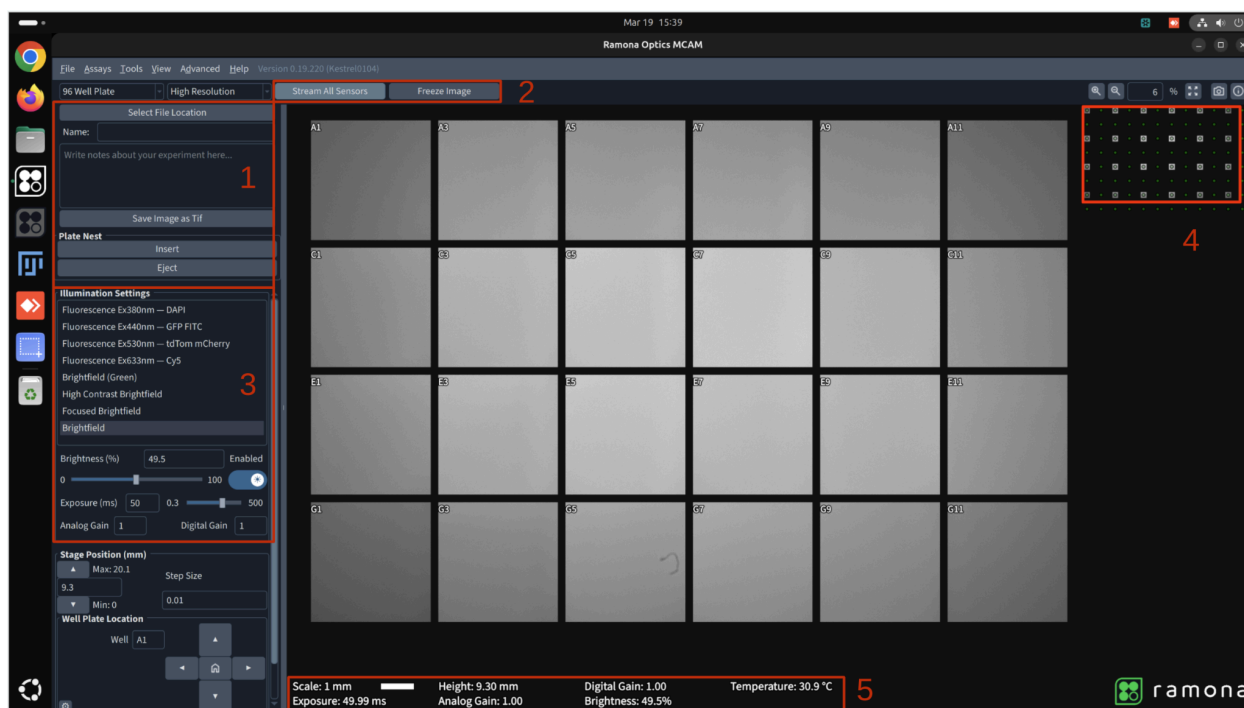


Fig 82: GUI layout with 1) Selected file location, 2) Image Acquisition Mode Buttons, 3) Image Acquisition Settings Panel, 4) Picture-in-picture and 5) Key Image Specifications highlighted.

At the top left of the window (Fig 82, #2), the Image Acquisition Mode Buttons allow the user to livestream the microscope or view at one single moment. The following are the functionalities provided:

- Stream All Sensors: livestream everything under the microscope at the wells selected by the users.
- Refresh Image: allows users to see everything under the microscope at one single moment at the wells selected by the users.

The image acquisition settings panel is on the left side of the viewing window (Fig 82, #3).



While panning, the picture-in-picture (Fig 82, #4) in the right-hand corner of the screen shows the portion of the wells currently in display.

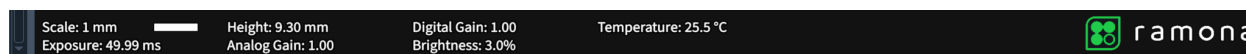


Fig 83: Key Image Specifications

Some key image parameters are displayed at the bottom of the viewing window (Fig 82, #5 and Fig 83) including a scale bar providing an approximate scale. Temperature indicated is the ambient temperature if the unit is without environmental control. Temperature indicated is the temperature in the environmental control chamber if the unit comes with environmental control.

MCAM Settings

Most of the MCAM Settings can be adjusted using the Settings Panel on the left side of the GUI.

The panel can be hidden to make more room on the screen by clicking on the white line border between the settings panel and the image, and sliding to the left. The GUI allows the user to set the system parameters within a range of values acceptable for most common experimental setups. More customized settings can be set in the Advanced submenu.

Most settings values can be adjusted either by using the sliders or by directly typing a value into the text box. When using the text box, the accepted values will still be limited to the maximum and minimum values shown on the respective slider. In addition, the setting displayed in the text box may adjust to a value slightly higher or lower than the value entered. This is because for certain settings (particularly digital and analog gain), the MCAM can only accept distinct values. The GUI will automatically round the entered setting to the nearest acceptable value, and send that parameter to the MCAM.

LOADING/SAVING SETTINGS

On first use, the GUI will load a set of predetermined default settings which were chosen to produce generally high-quality images. Afterwards, the GUI will automatically load the settings from its last use. Because each unique imaging environment and sample may have different optimized settings, the settings for individual



experiments can be saved and loaded using the “File” menu in the toolbar, selecting “Save Settings” within it. The default settings can also be re-loaded using the “Restore Default Settings” button in the same menu.

IMAGE ACQUISITION SETTINGS

The sensor exposure, gains, and LED parameters can all be set from the setting panel on the left side of the GUI, as well as contrast levels of the displayed image. These settings are further explained below.

EXPOSURE

This setting controls the amount of time that the camera sensor acquires light for each frame. A longer exposure will yield a brighter image with more motion artifacts while shortening the exposure will darken the image with fewer motion artifacts. Additionally this setting will affect the frame rate when using streaming acquisition modes with longer exposures, decreasing potential frame rate. In order to maximize frame rate, consider keeping the exposure as low as possible and balance lighting with the brightness setting.



Fig 84: Camera sensor settings.

DIGITAL AND ANALOG GAIN

Gain settings will magnify the signal produced by the image sensors. Analog gain increases the sensitivity of the sensors using hardware while digital gain increases the signal using software to multiply the signal once it has been converted to a digital signal. Increasing gain values will increase image intensity but will also increase noise proportionally. Keeping gain values as low as possible will reduce inherent noise (Fig 84).

BRIGHTNESS

Located under “Illumination Settings”, this will control the overall brightness of the LEDs used to illuminate the sample (Fig 85).

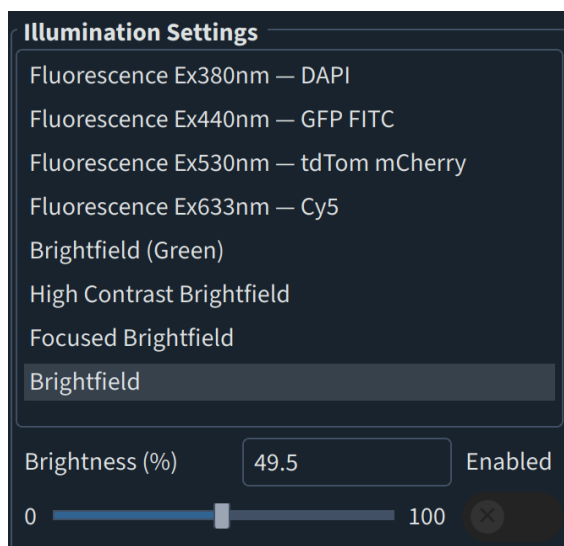


Fig 85: GUI LED board settings

STAGE POSITION SETTINGS

The location of the Z-stage can be changed using either the up and down arrows or by manually entering a location (given in millimeters from the highest point where the stage can be positioned) (Fig 86) on the Z scale within the “Stage Position Panel”. Step size controls the increment by which the arrows will adjust the height. Adjusting the Z-stage location will move the stage and sample in and out of focus.



Fig 86: Z-stage settings.

GAMMA

Adjusting this value will control the encoded image’s sensitivity to bright and dark tones and is an effective method of controlling the overall image contrast as perceived by humans (Fig 87). Reducing Gamma will decrease shadows and darker tones yielding a brighter image. Right-clicking on the Gamma bar allows users access to predetermined Gamma levels (Fig 88).

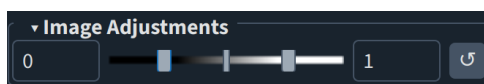


Fig 87: GUI contrast settings.

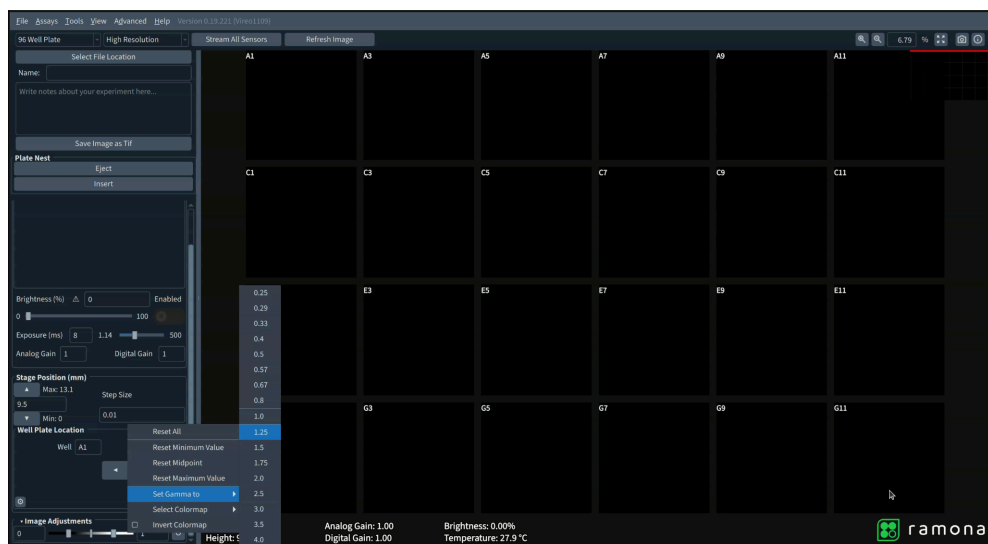


Fig 88: GUI contrast settings.

Save and Export Images

MCAM data can be exported or saved from the GUI in several ways. First, high resolution images can be exported using the "Save Image as Tif" button (Fig 89). "Select File Location" will open the directory in which the most recently saved images were placed.

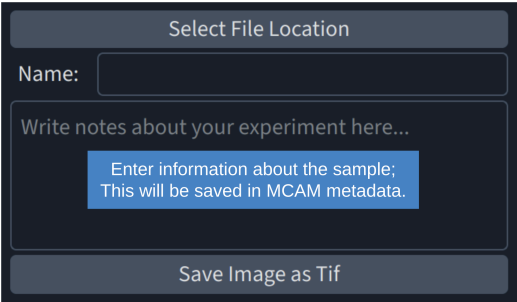


Fig 89: Save settings.

Saved images are given a filename according to the following rules:

STEP	ACTION
1	The user defines the prefix of this filename by entering in the "Name" field (Fig 89).
2	The suffix of this filename is generated as a timestamp with the following format • YYYYMMDD_HHMMSS_mmm where: • YYYY is the year when the image is saved, e.g. 2020. • MM is the month when the image is saved, e.g. 08. • DD is the day when the image is saved, e.g. 06. • HH is the hour (in 24 hour format) when the image is saved, e.g. 15. • MM contains the minute when the image is saved, e.g. 43. • SS contains the seconds when the image is saved, e.g. 52. • mmm contains the milliseconds when the image is saved, e.g. 549. • The final timestamp will appear as: 20200806_154352_549

Notes can be entered in the "Additional Information" field which are saved with the image file and displayed in the MCAM Viewer with the image.



Low resolution screenshots of the currently displayed images can be saved using “Capture Screenshot” in the “File” menu of the toolbar. This screenshot will include the main canvas, as well as the picture-in-picture and the main settings (see Fig 90).

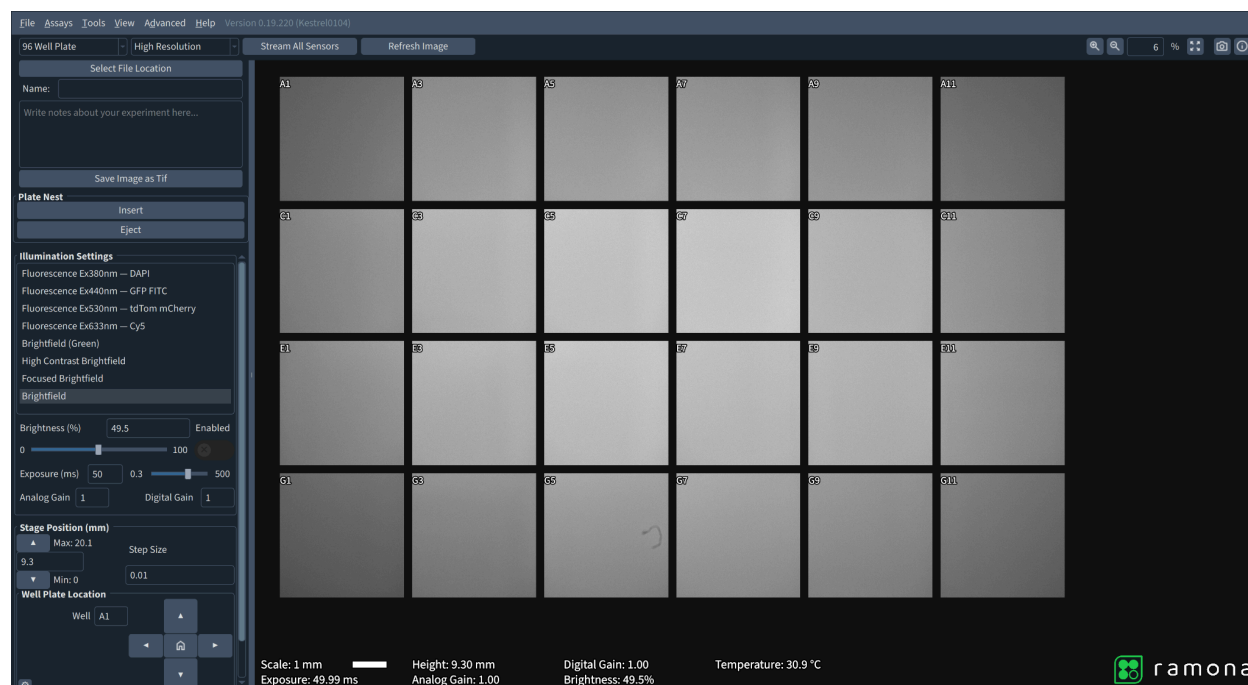


Fig 90: Screenshot of GUI Images

Note: If a filename is entered in the “Name” field and the workspace is changed, the filename will be deleted and the user will need to enter a new filename.



MCAM Data Analysis

Documentation of our Python API with examples is available at docs.ramonaoptics.com.

Background Subtraction

The Background Subtraction tool removes low-frequency background intensity variations from MCAM-acquired datasets. It is designed to correct for uneven illumination, optical haze, sensor bias, and large-scale autofluorescence that can interfere with downstream segmentation and quantitative analysis. This tool operates as a preprocessing step and is commonly used prior to Cell Counting, Thresholding, or Segmentation assays.

The analysis operates on a single image or a full XYZC stack. When used with a stack, background subtraction is applied after focus selection so that each image is processed using its best-focus plane. The analysis tool can be accessed in the MCAM Viewer by navigating to Advanced > Background Subtraction (Fig 91). When accessed directly from the MCAM Viewer Advanced menu, the resulting transformation only affects visualization. When selected in an analysis workflow (e.g. Cell Counting) background subtraction is applied to the dataset prior to subsequent computation.

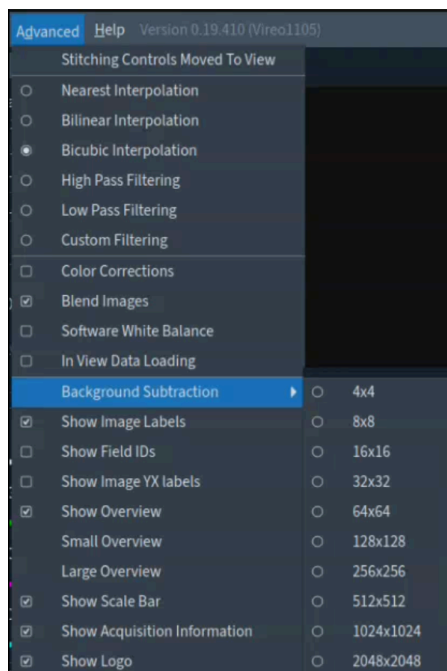


Fig 91: Advanced menu and chunk size options of the Background Subtraction tool.



Background subtraction in the MCAM software is implemented using a chunk-based spatial averaging model. Instead of estimating a single global background value, the algorithm computes a coarse background map by dividing each image into uniformly sized square regions ("chunks") and estimating the local mean intensity within each region. This approach preserves local contrast while suppressing large-scale intensity gradients.

The Background Subtraction visualization tool allows users to configure the following parameter:

1. **Chunk Size (pixels).** Defines the spatial scale over which background intensity is estimated.
2. Smaller values (e.g., 16–32 pixels) track finer illumination variations but may reduce contrast for large objects.
3. Larger values (e.g., 64–2048 pixels) provide smoother background estimates and are recommended for gradual illumination gradients or wide-field autofluorescence.
4. The optimal chunk size depends on magnification, feature size, and illumination uniformity. As a general rule, the chunk size should be significantly larger than the largest feature you wish to preserve, but smaller than the spatial scale of illumination variation.

The background removal process consists of the following steps:

1. **Chunking of Image Data.** Each image is divided into non-overlapping chunks of size $N \times N$ pixels, where N is the user-defined chunk size. The chunk size must be an integer divisor of both the image width (x) and height (y).
2. **Local Mean Background Estimation.** For each chunk, the algorithm computes the mean pixel intensity, producing a low-resolution background grid representing slow-varying illumination and sensor offsets.
3. **Background Surface Reconstruction.** The coarse background grid is resized to full image resolution using bilinear interpolation. This produces a smooth background surface that matches the spatial dimensions of the original image.
4. **Subtraction and Clipping.** The reconstructed background surface is subtracted pixel-wise from the original image. Resulting pixel values are clipped to the valid range $[0, 255]$ and cast back to the original image data type to ensure compatibility with subsequent analysis steps.

Only images marked as valid in the dataset metadata are processed; invalid or empty image tiles are skipped automatically.

1. Once the user clicks Apply Background Subtraction, the MCAM software performs the following operations:
2. **Data Validation.** Confirms that image dimensions are divisible by the selected chunk size and that image data are loaded into memory.
3. **Chunk Mean Computation.** The software reshapes the image into chunk blocks and computes mean intensities across spatial axes.
4. **Background Map Interpolation.** Chunk means are resized to full image resolution using linear interpolation.
5. **Image Correction.** The interpolated background is subtracted from the original image, and corrected images are clipped and visualized.

The operation is applied independently to each image and channel selected for analysis.

After background subtraction, the corrected image is displayed automatically in the MCAM Viewer. Visual inspection of multiple images is recommended to ensure that background gradients are removed without

suppressing true signal. Background subtraction affects image data only and does not alter detection thresholds, object sizes, or measurement parameters unless followed by additional analysis steps.

Background subtraction in the MCAM Viewer modifies the input dataset in memory for visualization only and does not generate standalone result files. When combined with other assays (e.g., Cell Counting), the corrected images are used directly by subsequent analysis stages.

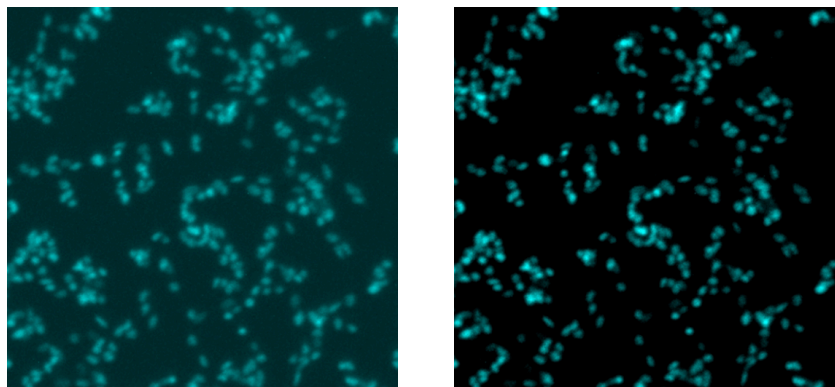


Fig 92: Nuclear stain in DAPI channel taken with 4x magnification (left), and with background subtraction applied (right).

Cell Counting

The Cell Counting module identifies discrete and solid fluorescent or brightfield features such as nuclei stain within each well of an MCAM-acquired dataset. The analysis operates on either a single image or a full XYZC stack (using automatic focus-slice selection). Results include bounding boxes for each detected cell and per-well cell counts. The Cell Counting Assay can be accessed in the MCAM Viewer by navigating to Assays > Cell Counting (Fig 93).

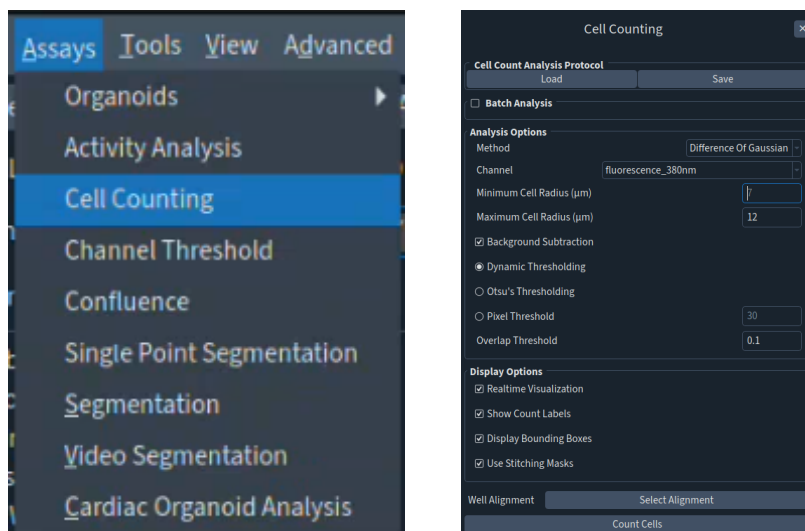


Fig 93: Assays menu (left); Cell Counting panel (right).



The Cell Counting panel allows users to define cell size range (μm) by minimum (1.0-499.0 μm) and maximum (2.0-500.0 μm) radii. These values define the allowable physical size of cells. Internally, the MCAM software converts radii from micrometers to pixel σ -values for the Difference-of-Gaussian (DoG) detector:

$$\sigma = \frac{r_{\text{meters}}}{\text{pixel width} \times \sqrt{2}}$$

Only cells with estimated radii within this range will be retained.

Background subtraction is selected by default to remove background signals. This improves performance when images contain haze, uneven illumination, or non-uniform autofluorescence. Any illumination channel present in the dataset may be selected. If a brightfield channel is used, images are automatically inverted because bright objects appear dark.

Cell counting relies on accurate foreground segmentation, with 3 methods to choose from:

1. Dynamic Thresholding (Default), where it automatically detects a cell-specific threshold from the image histogram by computing a smoothed histogram (Gaussian-filtered), calculating its derivative, then choosing the “first major gradient drop” as the threshold, on a per-FOV basis. This is recommended for heterogeneous illumination or low-signal images.
2. Otsu Thresholding, where it automatically selects a global threshold maximizing inter-class variance, on a per-FOV basis.
3. Manual Pixel Threshold, where users enter a fixed pixel value (0-255). The same value is then applied across all tiles.

Once the user clicks Detect Cells, the MCAM software goes through the following steps:

1. Channel Extraction and Focus Determination: the MCAM software extracts the selected channel from the dataset. If the dataset is a 3D stack, it computes a Laplacian slice-selection projection to pick the best-focus frame per tile to ensure that cell count operates on a uniformly focused mosaic.
2. Image Preprocessing: background signals are removed (if background subtraction is selected), and thresholds are defined using the method selected.
3. Cell count: the MCAM software uses the DoG method, with Gaussian filtering at multiple scales (σ values within user-defined min/max range); subtraction of adjacent scale images to enhance round structures; peak detection using `peak_local_max`, and conversion of detected peaks into cell coordinates and estimated radii.
4. Bounding Box Construction: the centroid of each detected cell is converted into a bounding box:

$$\begin{aligned} r &= \sigma \times \sqrt{2} \\ y_{\min} &= y - r \\ x_{\min} &= x - r \\ y_{\max} &= y + r \\ x_{\max} &= x + r \end{aligned}$$

The MCAM software ensures that bounds are clipped to valid image dimensions; cropping offsets are applied so bounding boxes are given in full-sensor coordinates within the stored analysis results, and each tile stores an array of all cell boxes.

5. Output Assembly: The MCAM software produces three datasets - Bounding boxes dataset, with all coordinates for all detected cells and per-tile cell counts; Cell count dataset, with per tile cell count;

Analysis input dataset, with threshold values, cell radius parameters, preprocessing flags that can be used for reproducibility and metadata export.

All outputs are saved in the cell_analysis_results folder. Two key files are generated: analysis_metadata.nc, where it contains bounding boxes, cell counts, threshold maps, preprocessing parameters, and illumination mode information; and cell_count.csv, where it summarizes cell counts on a per-FOV basis, and includes headers for minimum and maximum radius, thresholding mode(s) used, background subtraction flag, image inversion status, and assay timestamp and instrument metadata (Fig 94, left).

When analysis completes, bounding boxes appear as overlays around detected cells. "Cell Count" can be toggled in the Display Layers panel. Background thresholds and metadata are stored in the analysis results dataset. You may inspect each tile by clicking on wells in the MCAM Viewer, verifying segmentation accuracy, adjusting parameters, and re-running the analysis if required.

The Display Options group box controls how results are visualized in the MCAM Viewer. These settings affect display only; they do not change the underlying analysis results. By default, realtime visualization is enabled so that, as Cell Count runs, the MCAM Viewer shows a best-focus projection and overlays newly detected cells and counts per tile. When Count Labels is enabled, a text label such as "Count: 37" appears for each tile, anchored in the lower-left corner of the image. These labels update live during analysis and remain available afterward until you close the panel or disable labels. Finally, each detected cell is outlined by a red rectangle (bounding box) over the image in real time as cells are found, and post-analysis, when Bounding Box is enabled (Fig XXX, right).

Example Cell Detection Workflow

STEP	ACTION
1	Capture an XYZC stack at 4x or 10x magnification, optimally in the DAPI - 380nm fluorescence channel and Laplacian projection.
2	Open the acquired dataset in the MCAM Viewer software.
3	Go to Assays > Cell Counting. A panel will open on the right.
4	Input minimum and maximum cell radii in um.
5	Choose the threshold method.
6	Click "Count Cells" to run the analysis.



	A	B	C
1	# header		13
2	# owl_version	0.19.406	
3	# serial_number	Vireo1110	
4	# assay_timestamp_local	2025-08-18T08:40:06.230	
5	# notes		
6	# min_radius	0.000007	
7	# max_radius	0.000013	
8	# background_subtraction	True	
9	# invert_images	False	
10	# dynamic_thresholding	True	
11	# osu_thresholding	False	
12	# enable_nms	True	
13	# nms_iou_threshold	0.1	
14	Well Row Letter	Well Column Number	Cell Count
15	A	1	2430
16	A	2	3748
17	A	3	3310
18	A	4	3876
19	A	5	1435
20	A	6	1558
21	A	7	1771
22	A	8	1274
23	A	9	1175
24	A	10	509
25	A	11	1013
26	A	12	902
27	B	1	3162
28	B	2	3701
29	B	3	2756
30	B	4	3503
31	B	5	1587
32	B	6	1070
33	B	7	905
34	B	8	1420

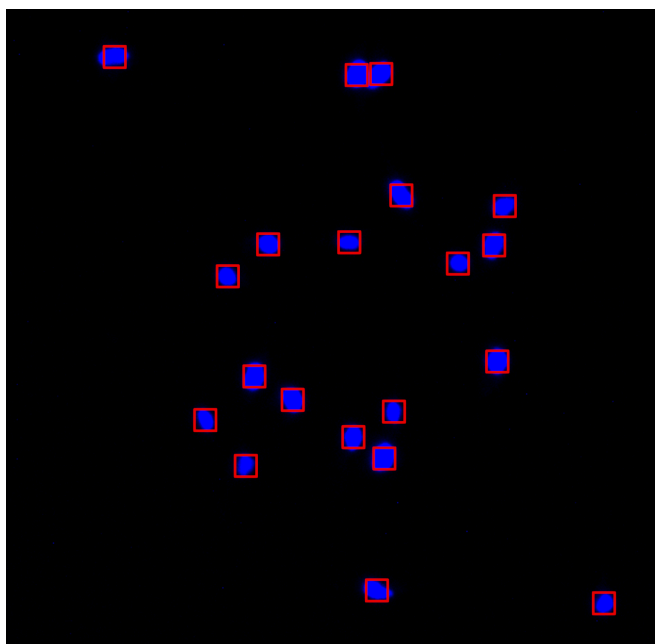


Fig 94: An example of a csv output (left), and nuclei detected in bounding boxes (right).

Channel Threshold

The Channel Threshold module quantifies signal-positive regions by counting the number of pixels within a user-defined intensity range for each illumination channel in an MCAM dataset. This analysis is designed for assays where a user wants to quantify the area of fluorescence.

The analysis operates on either a single image or a multi-frame XYZC stack (using optional projection methods). Results include binary threshold masks for each field of view (FOV) and per-image quantified areas, converted to SI unit. The Channel Threshold analysis can be accessed in the MCAM Viewer by navigating to Assays > Channel Threshold.

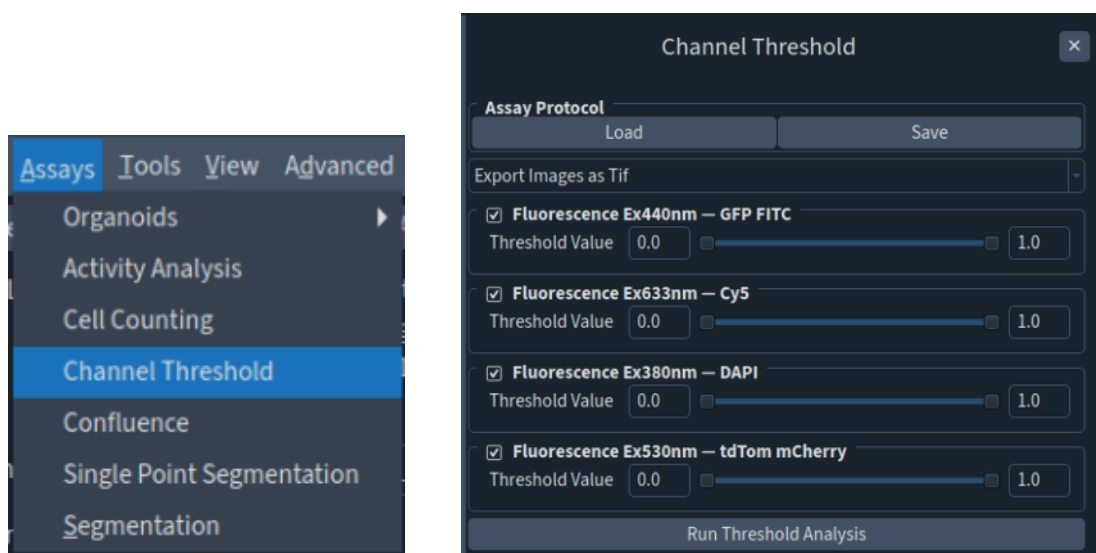


Fig 95: Assays menu (left); Channel Threshold panel (right).

Channel Threshold performs a pixel-wise inequality test on image intensities. For each selected illumination channel, pixels are retained if their values fall between a minimum and maximum normalized threshold, defined relative to the image data type's maximum possible intensity. Pixel counts are automatically converted to area measurements in SI units.

The Channel Threshold panel allows users to configure the following parameters:

- **Thresholds (Per Channel).** Thresholds are defined per illumination mode by users as a pair of normalized values of minimum and maximum threshold. Each threshold value is specified as a fraction of the maximum representable intensity for the image data type. Internally, the software computes:

$$\text{mask} = (\text{image} \geq \text{min_threshold} \times \text{max_value}) \text{ AND } (\text{image} \leq \text{max_threshold} \times \text{max_value})$$

Only pixels satisfying this are included in the threshold mask.

- **Projection Selection (Optional).** For XYZC stacks, users may specify a projection method per illumination mode (e.g., Laplacian-based selection). If projections are not specified, the middle frame is automatically selected. This ensures that thresholding is performed on a single, consistent 2D representation per image.
- **Background Subtraction (Optional).** An optional background subtraction step may be enabled using a user-defined chunk size to improve robustness in datasets with uneven illumination or background haze. When enabled, background is estimated and removed on a per-FOV basis. If this step is disabled, thresholding is applied directly to the raw image data.

Once the user initiates Channel Threshold analysis, the MCAM software performs the following steps:

1. **Dataset Validation.** The software verifies that the dataset is not a timelapse sequence, the image is monochrome (single chroma), and threshold and projection definitions are internally consistent. Invalid configurations result in descriptive runtime errors before processing begins.
2. **Channel Extraction and Projection.** For each channel and FOV, the corresponding illumination mode is extracted. If projections are specified, the selected projection method is applied. Otherwise, the center frame of the stack is used.

- Background Subtraction (if enabled). Illumination background is removed using tiled subtraction. Processing is performed independently for each FOV.
- Threshold Mask Generation. For each channel and FOV, image data is compared against the user-defined threshold range. A binary mask is generated where valid pixels are marked as 1.
- Area Calculation. The total number of threshold-positive pixels is computed per tile. If pixel width calibration is available, the pixel count is converted to physical area where $\text{Area} = \text{pixel count} \times (\text{pixel width})^2$.
Units are stored internally as m^2 and displayed as mm^2 . If calibration is unavailable, results are reported in px^2 .

All outputs are saved in the `threshold_analysis_results` folder. Two key files are generated: `metadata.nc`, which contains binary mask images indicating threshold-positive regions (one mask per channel per image) that automatically display as an overlay in the MCAM Viewer, and numeric area measurements per image, displayed as text overlays (e.g., "area: 1.23 mm^2 "); and `channel_threshold.csv`, which summarizes threshold areas for the selected channels on a per-image basis, and includes headers describing thresholds and output units, background subtraction flag, assay timestamp and instrument metadata for ease of reproducibility (Fig 96, left).

Prior to Channel Threshold analysis, threshold masks are shown as overlays as users change minimum and maximum threshold live to assist users in defining appropriate minimum and maximum threshold values. Binary masks appear automatically upon analysis completion, and area labels are rendered per image (Fig 96, right). Users may toggle mask visibility and label display in the Display Layers panel. Visualization settings affect display only and do not modify analysis results.

Example Channel Threshold Workflow

STEP	ACTION
1	Open the acquired dataset in the MCAM Viewer software.
2	Go to Assays > Channel Threshold. A panel will open on the right.
3	Deselect the channels that don't need to be analyzed by unchecking the box of the unrelated channels.
4	Define the upper and lower bounds of the threshold by moving the sliders or by entering values.
5	Click "Run Threshold Analysis" to run the analysis.

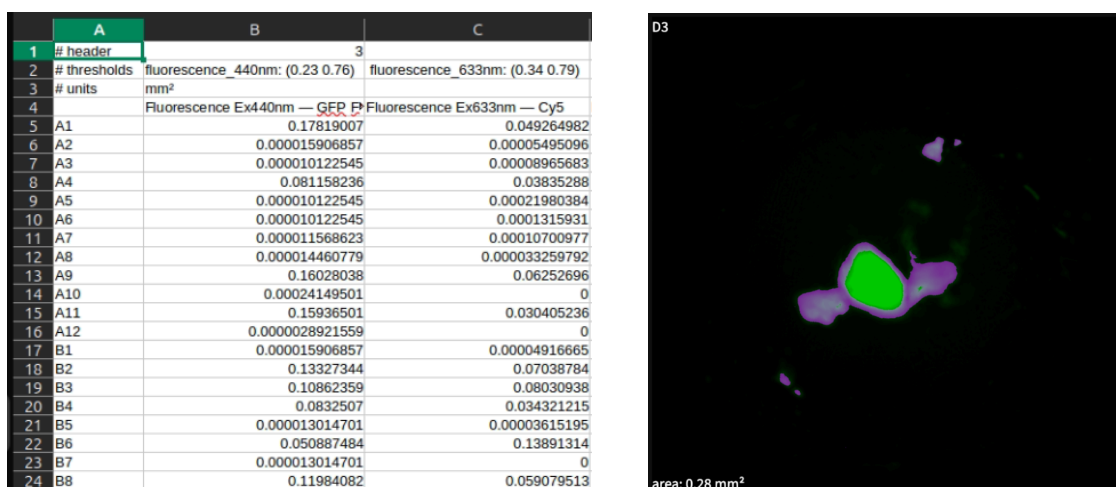


Fig 96: An example of a csv output (left), and a binary mask (purple) overlaying GFP signal within the threshold range defined pre-analysis (right).

Confluence Analysis

The Confluence Analysis module quantifies the fractional area of a well or field-of-view (FOV) that is occupied by cells. Confluence analysis measures surface coverage, making it well suited for applications such as cell growth monitoring, toxicity assays, and quality control of culture uniformity.

Confluence is computed by performing semantic segmentation of cellular regions using a deep-learning model and calculating the ratio of segmented cell pixels to total valid image area. The analysis operates on a single image or a full XYZC stack, with automatic best-focus slice selection for stack datasets. Results are reported as a normalized confluence value (0–1, displayed as a percentage) per FOV or per well. The Confluence assay can be accessed in the MCAM Viewer by navigating to Assays > Confluence (Fig 97).

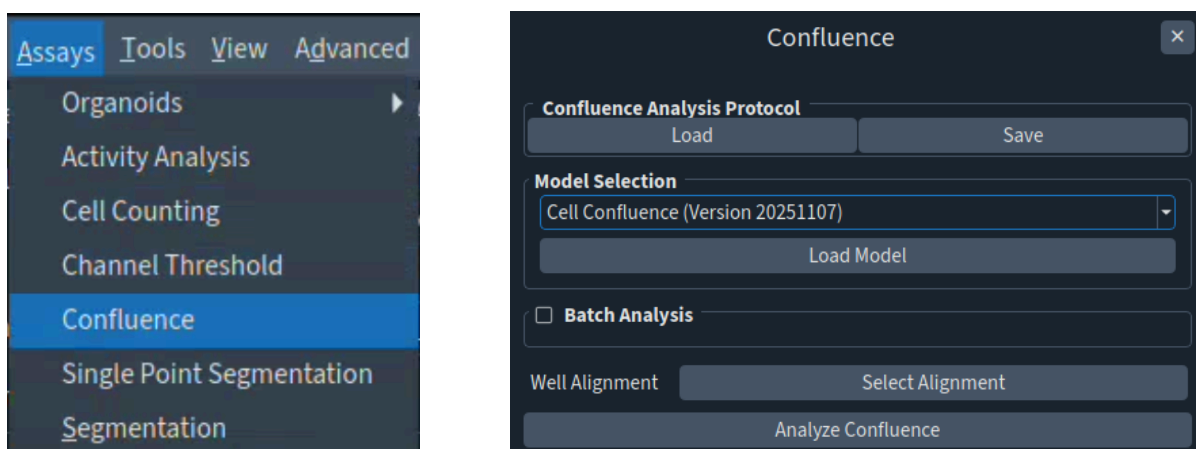


Fig 97: Assays menu (left); Confluence panel (right).



The Confluence panel allows users to configure the segmentation model and channel selection. Users select a segmentation model from the available model registry. Each model corresponds to a trained UNet-based neural network optimized for specific cell types and imaging modalities. If the selected model is not available locally, it is automatically downloaded and cached for future use. The name of the segmentation model used is stored in the analysis metadata and exported with results for reproducibility.

Optionally, users can choose to have the Confluence Analysis compute pooled confluence on a per well basis by combining all FOVs belonging to the same physical well through creating and applying well-plate keypoints. Circular well masks are applied to exclude regions outside the well boundary from the calculation when well keypoints are applied prior to analysis.

Once the user initiates the analysis, the MCAM software performs the following steps:

1. **Focus Determination.** If the dataset contains a Z-stack, the software applies a Laplacian slice-selection projection to determine the best-focus plane for each tile, ensuring consistent segmentation quality across the mosaic.
2. **Segmentation Model Loading.** The selected UNet-based segmentation model is loaded with GPU acceleration enabled by default and automatic fallback to CPU execution if a compatible GPU is not available.
3. **Cell Segmentation.** The segmentation model is applied to each tile, producing a binary segmentation mask where foreground pixels correspond to cellular regions. These masks represent per-pixel classifications rather than discrete objects.
4. **Mask Refinement.** To ensure accurate spatial quantification, segmentation masks are refined using stitching masks, which remove overlap regions introduced during image tiling; and well geometry masks (if keypoints are provided), which exclude pixels outside the physical well area.
5. **Confluence Computation.** Confluence is computed as the ratio of segmented cell pixels to total valid pixels:

Single-field confluence:

$$\text{Confluence} = \frac{\text{number of cell pixel}}{\text{number of total valid pixel}}, \text{ in an FOV}$$

Multi-field (pooled) confluence:

For each well, segmented pixels are summed across all FOVs in that well and normalized by the known well area.

6. All confluence values are stored as floating-point numbers and displayed as percentages in the viewer.
7. All outputs are saved in the `confluence_analysis_results` folder. Two key files are generated: `analysis_metadata.nc`, which contains confluence values, formatted as percentages for display and a binary mask dataset showing segmented cellular regions; and `confluence_results.csv`, where it summarizes confluence values on a per-FOV or a per-well basis, and includes headers for selected segmentation model, assay timestamp and instrument metadata (Fig 98, left).

Visualization settings affect display only and do not alter computed results (Fig 98, right). Users are encouraged to inspect representative wells and tiles to verify segmentation quality.

Example Confluence Workflow

STEP	ACTION
1	Capture an XYZC stack at 4x or 10x magnification.
2	Open the acquired dataset in the MCAM Viewer software.
3	Go to Assays > Cell Confluence. A panel will open on the right.
4	Optional: Select a Well Alignment file specific to the well plate in use for multi-field per well datasets.
5	Click "Analyze Confluence" to run the analysis.

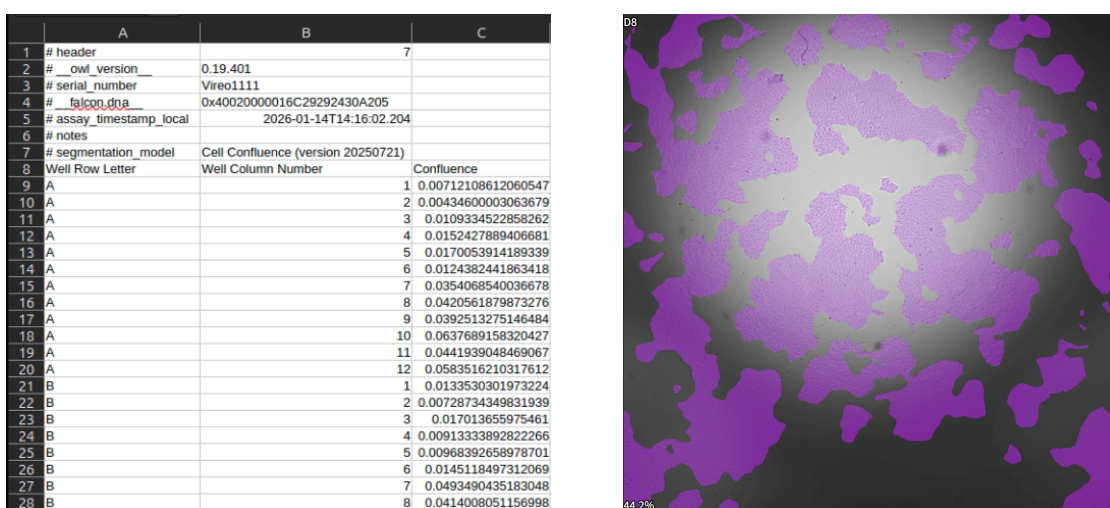


Fig 98: An example of a csv output (left), and confluence with segmentation masks on (right).

Macroview

The Macroview feature automatically generates a high-quality overview image of your MCAM acquisition. It creates a stitched, rendered view of the entire dataset that can serve as both a quick visual reference thumbnail and a publication-ready image depending on your resolution and quality settings.

Macroview is available during XYZC stack acquisitions and generates the overview image automatically after acquisition completes. The output includes optional overlays such as the Ramona logo, acquisition metadata, scale bar, and analysis results (masks, bounding boxes, and labels) if runtime analysis was performed. The Macroview feature can be accessed in the MCAM Acquisition software by navigating to Assays > Acquire XYZC Stack > Advanced Save Options > Macroview tab (Figs 99 + 100).

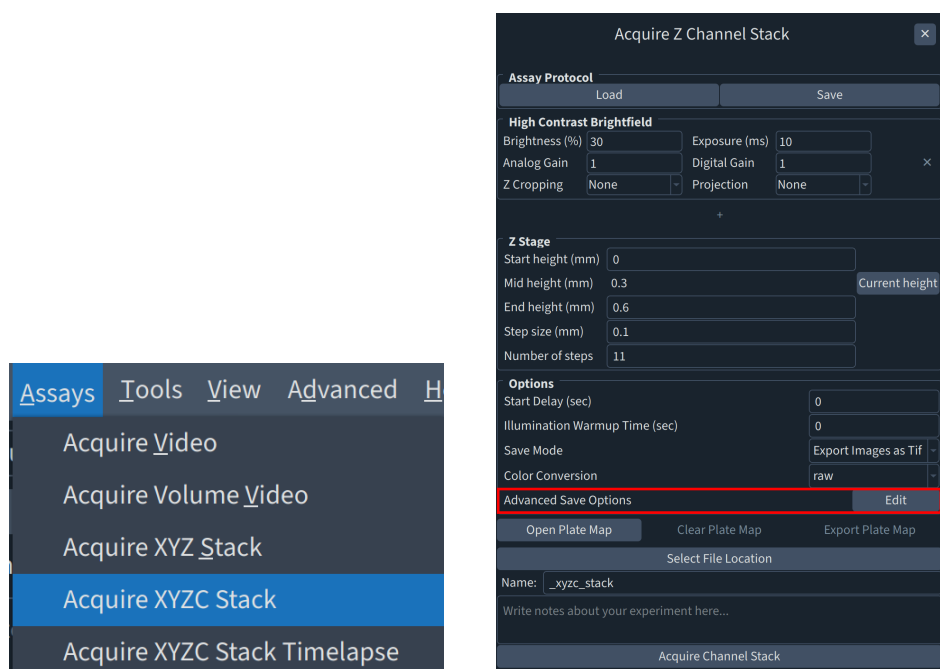


Fig 99: Assays menu (left); XYZC Stack panel with Advanced Save Options highlighted (right).

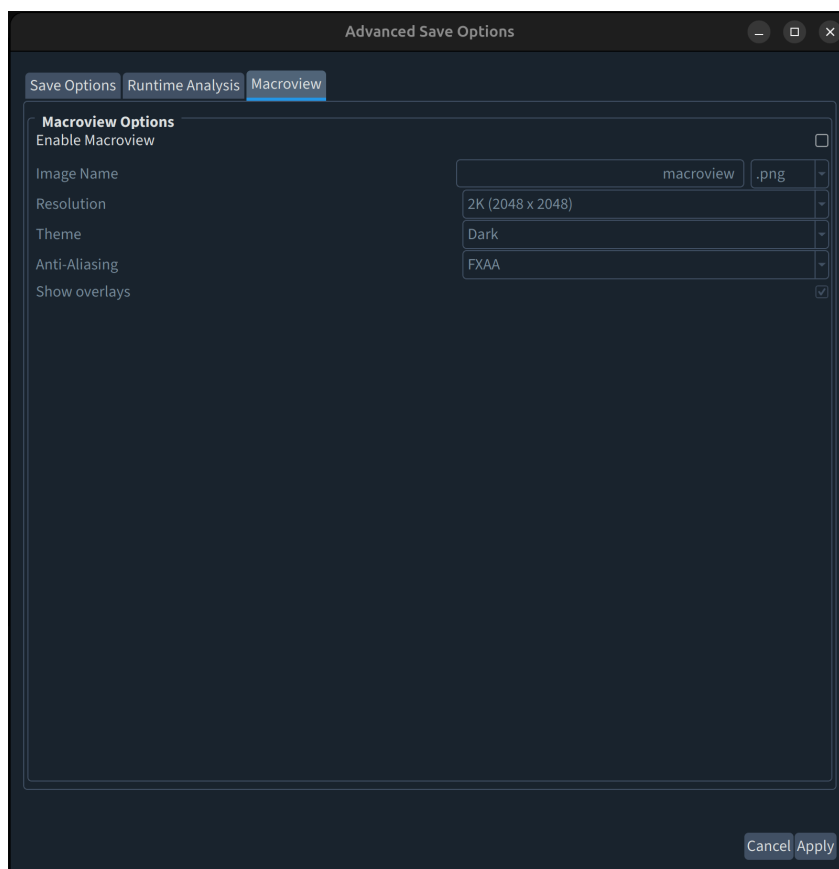


Fig 100: The Macroview tab in Advanced Save Options.

To enable Macroview, check the “Enable Macroview” box. The Macroview panel allows users to configure the following parameters:

- Image Name. The output filename for the Macroview image. You can customize the base name (default: macroview) and select the file extension from the dropdown:

Extension	Description
.png	Lossless compression, best for images with text overlays
.jpg	Lossy compression, smaller file size
.tif	TIFF format, suitable for further processing
- Resolution. Select the output image resolution. Higher resolutions produce more detailed images but take longer to render.

Preset	Dimensions	Use Case
1K	1024 x 1024	Quick preview, thumbnails
2K	2048 x 2048	General purpose (default)
3K	3072 x 3072	Higher quality output
Well Plate 1.5MP	1536 x 1024	Well plate aspect ratio
Well Plate 6MP	3072 x 2048	Well plate, higher resolution
Well Plate 12MP	4032 x 2688	Well plate, publication quality
- Theme. Controls the visual appearance of the rendered scene:
 - Dark (default): Dark background, suitable for fluorescence imaging
 - Light: Light background, suitable for brightfield imaging or printed documents
- Anti-Aliasing. Post-processing anti-aliasing reduces jagged edges in the rendered image:

Mode	Description
None	No anti-aliasing, fastest rendering
DDAA	Data-dependent anti-aliasing
FXAA	Fast approximate anti-aliasing (default), good balance of quality and speed
- Show Overlays. When enabled, displays informational overlays on the rendered image including: - Ramona Optics logo - Acquisition metadata (timestamp, instrument info) - Scale bar - Well labels (for well plate acquisitions)

Macroview generation runs in the background, so you can continue using the application while it completes. If runtime analysis was performed during acquisition, analysis results (masks, bounding boxes, labels) are automatically included in the Macroview output.

Example Macroview Workflow

STEP	ACTION
1	Go to Assays > Acquire XYZC Stack. A panel will open on the right.
2	Configure illumination modes, Z stage parameters, and acquisition settings.
3	Click Edit next to Advanced Save Options. An “Advanced Save Options” window will appear.



- 4 Navigate to the Macroview tab.
- 5 Check Enable Macroview.
- 6 Set desired image name and resolution.
- 7 Adjust Theme to match your imaging modality.
- 8 Enable Show Overlays for metadata display.
- 9 Click Apply to save settings.
- 10 Click Acquire Channel Stack to start acquisition.

After acquisition completes, the Macroview image will be generated automatically and saved to the same directory as your acquisition data.

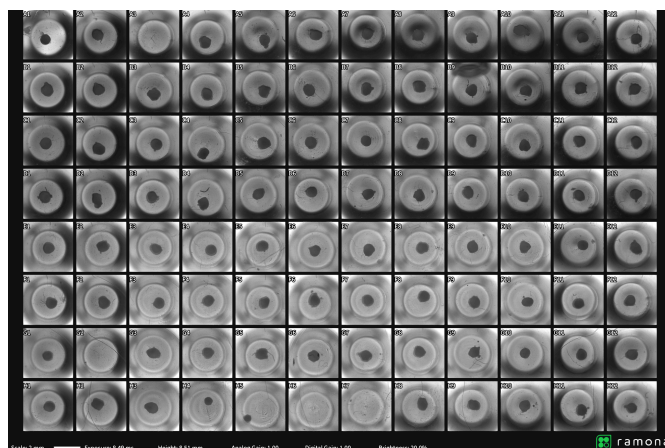


Fig 101: A Macroview of a V-bottom 96-well plate containing single organoids per well.

Region of Interest (ROI) Extraction

The Region of Interest (ROI) Extraction tool enables users to define, visualize, and export fixed-size image regions centered on user-specified locations across all fields of view (FOVs) in an MCAM-acquired dataset. This module is designed for downstream quantitative analysis, visualization, and machine-learning workflows where consistent spatial cropping across the Multi-Camera Array is required and for data size reduction purposes.

The analysis operates on single images or full XYZC datasets and preserves all relevant spatial and photometric metadata during extraction. Results include per-FOV cropped ROI images and updated coordinate metadata. The ROI Extraction tool can be accessed in the MCAM Viewer by navigating to Tools > Regions of Interest (Fig 102).

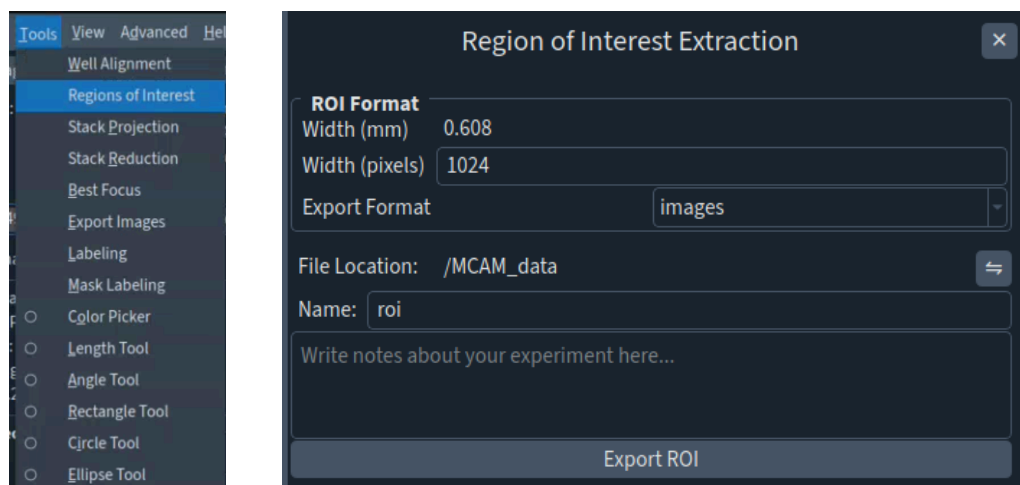


Fig 102: Tools menu (left); Region of Interest Extraction panel (right).

The ROI Extraction panel allows users to define a square ROI size in pixels. The ROI width represents the width of the square region and is constrained to valid values based on the dataset resolution, binning mode, and instrument configuration. The minimum ROI width is set at 64 pixels, while the maximum ROI width is limited by the smallest image dimension after binning. ROI sizes are quantized to valid multiples of the binning factor to ensure pixel-aligned extraction. If pixel size calibration is available, the ROI width is also displayed in millimeters for physical reference.

Each ROI is defined by a center point and a fixed width. ROI centers and extents are visualized directly in the MCAM Viewer using interactive overlays (Fig 103, left):

- Center marker: displayed as a click-and-draggable point
- ROI boundary: displayed as a square outline centered on the marker

Users may drag the center marker to reposition the ROI, and drag the ROI edge to resize the ROI interactively.

During interaction, the software enforces pixel alignment consistent with binning and boundary clamping to prevent ROIs from extending beyond valid image regions, with automatic snapping to defined ROI widths. Hover and active selection states are indicated by color changes to assist precise placement. All changes are applied per-FOV while respecting the dataset's valid-data mask.

After editing, the MCAM software performs boundary checks to ensure that each ROI lies fully within the visible image area. If an ROI would extend beyond image limits, its center is automatically shifted inward by the minimum amount required to keep the ROI valid. This adjustment is applied immediately after interactive edits, automatically when ROI width is changed numerically, and across all FOVs when global ROI parameters are updated. These safeguards ensure consistent, artifact-free ROI extraction across the entire camera array.

Once the user initiates ROI export, the MCAM software executes the following steps:

1. **Keypoint Normalization.** ROI centers are converted from display space into dataset image coordinates, accounting for binning, cropping offsets, and non-contiguous camera layouts.
2. **Subindexed Cropping.** For each FOV, a square ROI is extracted using a subindexed array mapping that avoids unnecessary data duplication and supports large datasets efficiently.
3. **Metadata Reprojection.** Dataset metadata is rewritten into the ROI coordinate system, including:
 - Spatial coordinates

- Well identifiers
 - Stack dimensions
 - Photometric and stitching metadata
4. Export Assembly. A new dataset is created containing only the extracted ROIs, preserving instrument metadata and user notes.

Exported ROIs are saved in the user-selected location, with filename appended with a timestamp. The output dataset contains:

- Cropped ROI image data for each FOV (Fig 103, right).
- Updated spatial coordinates starting at the ROI origin.
- Per-ROI identifiers and indexing metadata.
- Notes and export settings.

All exported data are fully compatible with downstream MCAM analysis modules and external image-analysis pipelines.

ROI overlays are displayed in real time during editing and remain visible until the ROI panel is closed. These overlays are for visualization only and do not modify the underlying image data.

Example ROI Workflow

STEP	ACTION
1	Open the acquired dataset in the MCAM Viewer software.
2	Go to Tools > Regions of Interest. A panel will open on the right.
3	Select the region of interest by click-and-dragging the red point and/or the perimeter of the square box.
4	Determine the export format.
5	Optional: Select the location for the output and provide a name for the folder.
6	Click "Export ROI" to export tiffs/mp4s of the ROIs selected.

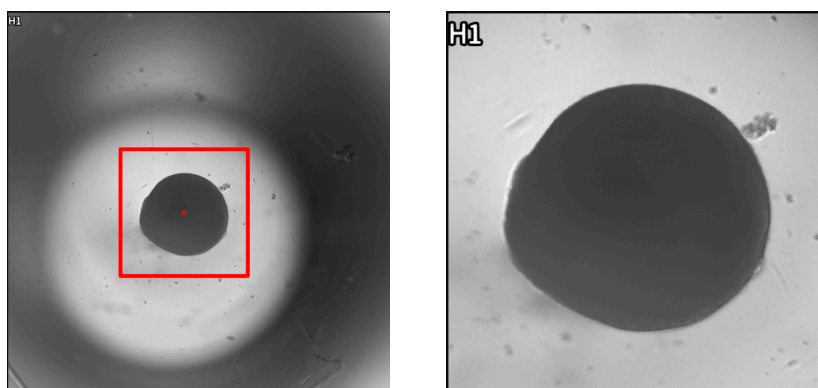


Fig 103: An image with the original image shape, with the red box indicating the ROI boundary and the red dot the center marker of the ROI boundary (left), and the exported ROI (right).

Segmentation

The Segmentation module in the MCAM software provides automated, pixel-accurate delineation of objects within MCAM-acquired datasets. Segmentation produces dense object masks that capture the full spatial extent of each detected object, enabling downstream measurements such as area, confluence, and per-object morphology metrics. This module is designed for applications including organoid analysis and 2D cell segmentation. The Segmentation module can be accessed in the MCAM Viewer by navigating to Assays > Segmentation (Fig 104).

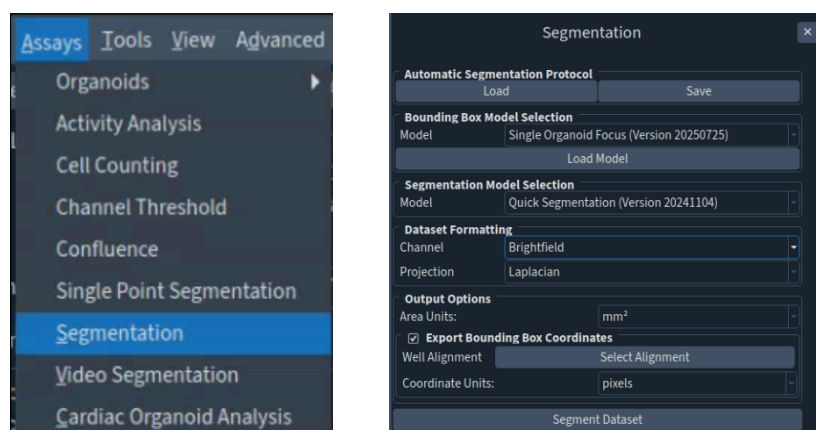


Fig 104: Assays menu (left); Segmentation panel (right).

Segmentation in the MCAM software is a two-stage process: Object Detection (Bounding Box Detection) where a trained object-detection model identifies candidate objects and generates bounding boxes for each field of view (FOV), followed by Pixel-Level Segmentation, where a deep learning segmentation model refines each detected bounding box into a precise object mask, producing per-pixel labels for each object.

Bounding boxes are generated using a dedicated object-detection model prior to segmentation. Depending on the selected model, each FOV may contain either a single object (e.g., one organoid per well), or multiple objects (e.g., multiple organoids per well).

For each bounding box, the MCAM software uses a transformer based segmentation mode to generate a binary mask representing the object footprint. Masks are combined into a labeled mask image where background pixels are set to 0, and each object is assigned a unique integer label (instance index). This representation supports both visualization and quantitative analysis. Masks are stored in a dedicated dataset variable and aligned with the original MCAM coordinate system.

This hybrid approach combines the robustness of object detection with the accuracy of modern segmentation models, ensuring reliable performance across heterogeneous illumination conditions and large datasets.

The Segmentation module operates on either a single 2D image, or XYZC Stack. When a Z-stack is provided, the MCAM software applies the selected projection method (default: Laplacian) to collapse the data into a single best-focus image per image prior to analysis. This ensures consistent segmentation quality across all images.

The Segmentation module automatically computes several quantitative metrics from the generated masks:

- **Area:** total segmented area per FOV, reported in physical units (e.g., mm²) when pixel size metadata is available.
- **Confluence:** fraction of the image area covered by segmented objects, expressed as a percentage.
- **Per-Object Areas:** individual object areas, indexed by instance ID, enabling multi-object analysis within a single FOV.
- **Per-object morphology metrics** such as circularity, eccentricity, and region intensity.

All outputs are saved in the segmentation_analysis_results folder. Two key files are generated: analysis_metadata.nc, where it includes bounding box coordinates, segmentation masks, model selections, and analysis parameters; and segmentation_individual_summary.csv and segmentation_well_summary.csv (for multi-object segmentation), where it summarizes individual object area and total area, circularity, eccentricity, region intensity, and object counts on a per FOV basis, and includes headers for selected segmentation model, assay timestamp and instrument metadata (Fig 105, left). Bounding box coordinates are also exported in the csv format (in pixel coordinates) with support for conversion into well-plate coordinate space and SI unit when keypoints are defined (Fig 106).

During and after analysis, segmentation results can be visualized directly in the MCAM Viewer, including segmentation masks overlaid on the original images, bounding boxes that can be toggled, area or confluence labels displayed on the images per FOV, and instance identifiers for multi-object datasets (Fig 105, right). These display options affect visualization only and do not modify the underlying analysis results.

Example Segmentation Workflow

STEP	ACTION
1	Capture an XYZC stack at 4x magnification in the brightfield channel.
2	Open the acquired dataset in the MCAM Viewer software.
3	Go to Assays > Segmentation. A panel will open on the right.
4	Within Bounding Box Model Selection Options, select the appropriate model.
5	Click "Segment Dataset" to run the analysis.

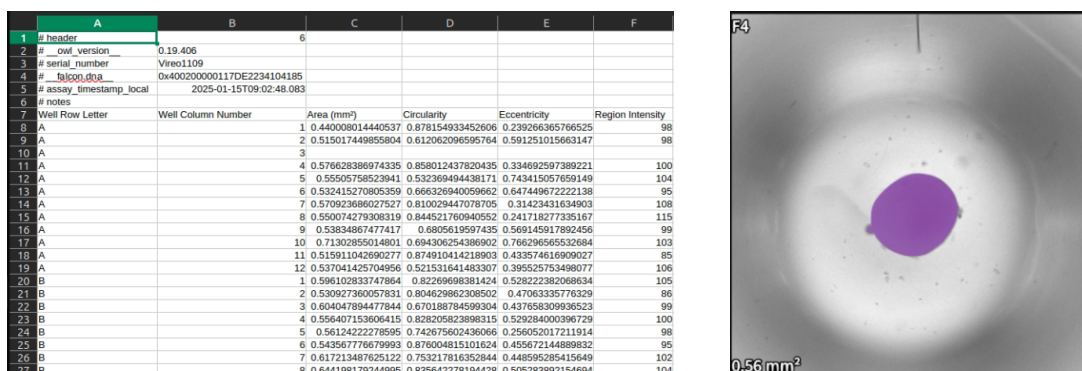


Fig 105: An example of a csv output (left), and a segmentation of an organoid, with the area of the organoid shown on the bottom-left corner (right).

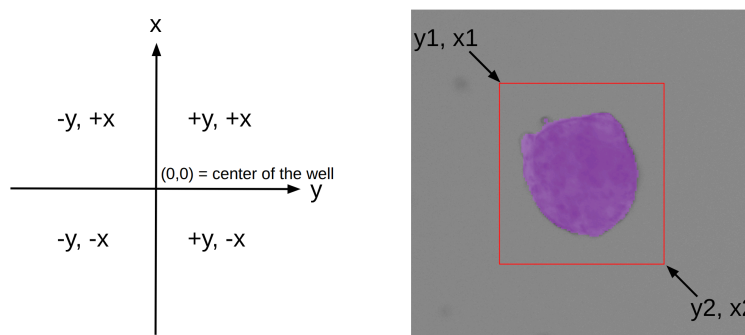


Fig 106: When a well alignment file is loaded during segmentation analysis, bounding box coordinates of the resulting csv file are converted from image space coordinates to image space coordinates. y,x coordinates of the center of the well, with relative locations indicated by plus (+) and minus (-) symbols (left), and the notation indicating the top-left and bottom-right corners of a bounding box (right).

Stack Projection

The Stack Projection tool generates a single representative image slice from an MCAM-acquired full XYZC stack by combining or selecting information across the stack. This tool is designed to reduce volumetric or time-resolved data into a single 2D image per field-of-view (FOV), enabling downstream analysis, visualization, and quantitative assays to operate on uniformly projected mosaics.

Stack Projection is commonly used for best-focus selection, signal aggregation, background suppression, or variance-based contrast enhancement across the full Multi-Camera Array. The tool can be accessed in the MCAM Viewer by navigating to Tools > Stack Projection (Fig 107).

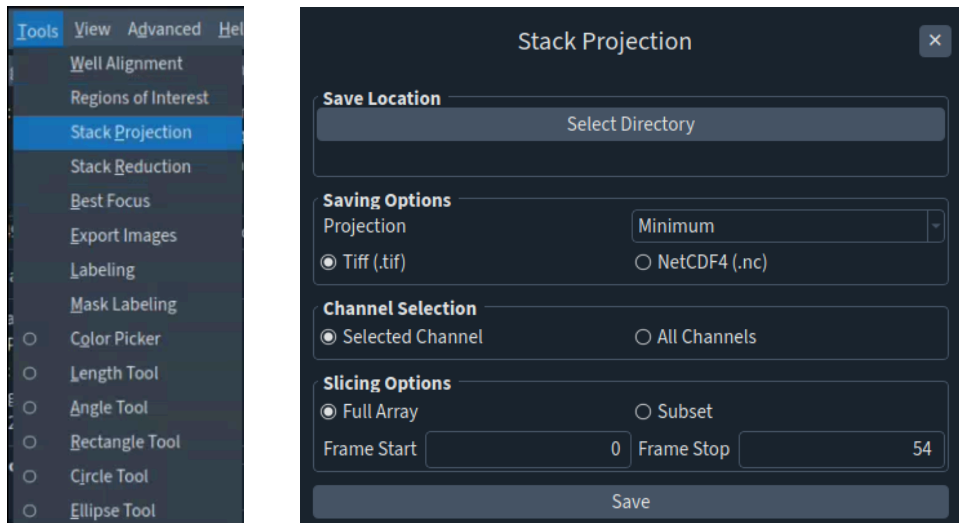


Fig 107: Tools menu (left); Stack Projection panel (right).

The analysis operates on single-channel or multi-channel datasets and supports full XYZ stacks and time stacks. Projection is applied independently to each FOV and, when applicable, per channel.

The Stack Projection panel allows users to select intended projection methods. Each method collapses the stack dimension into a single image using a defined mathematical or model-based operation:

Maximum and Minimum Projections

This method computes the pixel-wise maximum or minimum intensities across all slices. Maximum Intensity Projection is effective for highlighting bright, sparse features (e.g., fluorescent puncta) while Minimum Intensity Projection is useful for suppressing transient bright artifacts or emphasizing dark structures in inverted or brightfield datasets.

Formally, for a stack $I(x, y, z)$, the maximum projection image $P(x, y)$ is defined as:

$$P(x, y) = \max_{z \in Z} I(x, y, z),$$

while the minimum projection image is defined as:

$$P(x, y) = \min_{z \in Z} I(x, y, z),$$

where Z represents the stack dimension.

When Maximum or Minimum Projection is selected, the MCAM software executes the following steps:

1. Stack Dimension Resolution. The software automatically identifies the stack dimension using dataset metadata. If the dataset is not stacked, the input image is returned without modification.
2. Initialization using the first slice. The first slice in the stack is loaded and used to initialize the projection buffer.
3. Iterative Pixel-Wise Comparison. For each subsequent slice in the stack:
 - a. A pixel-wise comparison is performed between the current projection buffer and the new slice.



- b. For each pixel location (x, y) , the larger (Maximum Projection) or smaller (Minimum Projection) of the two values is retained:

$$P_{new}(x, y) = \max(P_{current}(x, y), I_{slice}(x, y)), \text{ or}$$

$$P_{new}(x, y) = \min(P_{current}(x, y), I_{slice}(x, y))$$

4. This process continues until all slices have been evaluated.
5. Stack Dimension Collapse. After processing all slices:
 - a. The stack dimension is removed ("squashed") from the dataset.
 - b. Metadata from the middle slice of the stack is retained to preserve alignment and acquisition context.
 - c. An indexed stack coordinate is added to record the representative slice reference.

Mean projection

This method reduces an image stack into a single 2D image by computing the arithmetic average of intensity values across all slices in the stack for each pixel location.

Formally, for a stack $I(x, y, z)$, the mean projection image $P(x, y)$ is defined as:

$$P(x, y) = \frac{1}{N} \sum_{z=1}^N I(x, y, z)$$

where N is the number of slices along the stack dimension.

When Mean Projection is selected, the MCAM software executes the following steps:

1. Stack Dimension Resolution. The software automatically identifies the active stack dimension using dataset metadata. If the dataset is not stacked, the input image is passed through unchanged.
2. Projection Buffer Initialization. A floating-point accumulation buffer is initialized with zeros:
 - a. The buffer has the same spatial dimensions as a single image.
 - b. A higher-precision floating-point type is used internally to prevent overflow and rounding errors during accumulation.
3. Slice Accumulation. Each slice in the stack is processed sequentially:
 - a. The slice image is loaded into memory.
 - b. Pixel values are added to the accumulation buffer:

$$A(x, y) = A(x, y) + I_{slice}(x, y)$$

- c. This step is repeated until all slices have been accumulated.
4. Normalization by Slice Count. After all slices are processed:
 - a. The accumulation buffer is divided by the total number of slices N:
- b. This produces the arithmetic mean intensity at each pixel location.
5. Data Type Conversion. The resulting floating-point mean image is cast back to the original dataset image data type.

$$P(x, y) = \frac{A(x, y)}{N}$$

Sum Projection

This method computes the pixel-wise sum across the stack. If the summed intensity exceeds an 8-bit range, values are automatically normalized to maintain compatibility with downstream visualization and analysis. Sum



Projection emphasizes cumulative signal strength and is useful when total photon counts or integrated intensity across depth or time are of interest.

Formally, for a stack $I(x, y, z)$, the sum projection image is defined as:

$$P(x, y) = \sum_{z=1}^N I(x, y, z)$$

where N is the number of slices along the stack dimension.

When Sum Projection is selected, the MCAM software executes the following steps:

1. **Stack Dimension Resolution.** The software identifies the active stack dimension using dataset metadata. If no stack dimension is present, the input image is returned without modification.
2. **Projection Buffer Initialization.** A floating-point accumulation buffer is initialized:
 - a. The buffer has the same spatial dimensions as a single image.
 - b. A higher-precision floating-point type is used internally to safely accumulate values without integer overflow.
3. **Slice Accumulation.** Each slice in the stack is processed sequentially:
 - a. The slice image is loaded into memory.
 - b. Pixel values are added to the accumulation buffer:

$$A(x, y) = A(x, y) + I_{\text{slice}(x,y)}$$

- c. This process continues until all slices have been accumulated.
4. **Dynamic Range Normalization.** After accumulation:
 - a. The maximum value in the summed image is evaluated.
 - b. If the maximum exceeds the allowable range of the original image data type (e.g., >255 for 8-bit images), the entire image is uniformly scaled:

$$P_{\text{scaled}}(x, y) = \frac{A(x,y)}{\max(A)/255}$$

- c. This normalization preserves relative intensity relationships while ensuring compatibility with downstream visualization and analysis modules.
5. **Data Type Conversion.** The normalized summed image is cast back to the dataset's native image data type.

Standard Deviation (STD) Projection

This method computes the pixel-wise standard deviation across slices. Regions with strong axial or temporal variation appear enhanced, making this method suitable for detecting motion, structural heterogeneity, or dynamic signal changes.

Formally, for a stack $I(x, y, z)$, the STD projection image $P(x, y)$ is defined as:

$$P(x, y) = \sqrt{\frac{1}{N} \sum_{z=1}^N (I(x, y, z) - \mu(x, y))^2}$$

where N is the number of slices in the stack, and $\mu(x, y)$ is the mean intensity at pixel location (x, y) across all slices.

When STD Projection is selected, the MCAM software executes the following steps:

1. **Stack Dimension Resolution.** The software identifies the active stack dimension. If the dataset is not stacked, STD projection cannot be computed and the input image is returned unchanged.



2. Internal Buffer Initialization. Two floating-point buffers are allocated: a mean accumulation buffer, and a variance accumulation buffer. Both buffers have the same spatial dimensions as a single image.
3. First Pass – Mean Computation. Each slice in the stack is processed sequentially:
 - a. Slice pixel values are added to the mean buffer:

$$M(x, y) = M(x, y) + I_{\text{slice}}(x, y)$$

- b. After all slices are processed, the buffer is divided by N to compute the mean image:

$$\mu(x, y) = \frac{M(x, y)}{N}$$

4. Second Pass – Variance Accumulation. The stack is traversed a second time:
 - a. For each slice, the difference between the slice and the mean is computed:

$$D(x, y) = I_{\text{slice}}(x, y) - \mu(x, y)$$

- b. The squared difference is accumulated in the variance buffer:

$$V(x, y) = V(x, y) + D(x, y)^2$$

5. Standard Deviation Calculation. After processing all slices:
 - a. The variance buffer is divided by N, and the square root is taken to obtain the standard deviation:

$$P(x, y) = \sqrt{\frac{V(x, y)}{N}}$$

6. Data Type Conversion. The resulting STD image is cast back to the dataset's native image data type. No additional normalization or scaling is applied.

Laplacian Focus Selection

Laplacian slice selection is a focus-selection method that reduces an XYZ or time-resolved stack into a single 2D image by selecting the single best-focused slice rather than mathematically combining slices. The selected slice is chosen independently for each FOV based on a Laplacian-based sharpness metric. Unlike mathematics-based projections, Laplacian selection preserves true image intensities and spatial structure by extracting an existing slice that is most likely to be in focus.

The Laplacian operator measures local second-order intensity variation, which is strongly correlated with image sharpness. In-focus images exhibit strong local gradients and edges, whereas out-of-focus images appear smoother and have reduced high-frequency content. By computing the variance of the Laplacian response, the software estimates how sharply focused each slice is.

When Laplacian projection is selected, the MCAM software executes the following steps:

1. Stack Dimension Identification. The software determines the active stack dimension. This dimension defines the candidate slices over which focus selection will occur.
2. Per-Slice Laplacian Computation. For each slice in the stack, the slice image is converted to floating-point precision. The Laplacian response is approximated by summing second-order spatial gradients:

$$L = \nabla_x^2 I + \nabla_y^2 I$$

A threshold is applied to suppress weak gradients:

- Laplacian values below a fixed threshold are set to zero.
- This reduces sensitivity to noise and low-contrast background.

3. Laplacian Variance Calculation. For each pixel (or downsampled region), the MCAM software computes:

$$V = \text{Var}(L)$$

- a. This variance represents the local sharpness score for that slice:
 - i. High variance → strong edges → good focus
 - ii. Low variance → blurred content → poor focus



The variance values are accumulated across all slices into a sharpness score volume.

4. Best-Focus Slice Selection. After evaluating all slices, The slice index with the maximum Laplacian variance is selected:

$$Z_{best} = \operatorname{argmax}_z V(z)$$

This selection is performed independently for each FOV.

5. Slice Extraction and Stack Collapse. Once the best-focus slice is identified, that slice is extracted from the original dataset. The stack dimension is removed, and a new metadata variable records the selected slice index. No intensity modification, averaging, or normalization is performed.

Focus Fusion

Focus fusion projection generates a single 2D image by combining only the in-focus regions from multiple slices in an XYZ stack, rather than selecting a single slice or mathematically projecting all slices uniformly. The result is a composite image that is locally in focus everywhere, even when different structures lie at different axial depths. Unlike Laplacian slice selection, which chooses one best slice per FOV, focus fusion operates spatially and locally, allowing different pixels (or regions) to originate from different slices without intensity averaging or normalization. The core idea behind focus fusion is that in-focus regions contain high local spatial energy (sharp edges, strong gradients), whereas out-of-focus regions are smoother and lower in high-frequency content. By measuring local energy across slices, the algorithm can select, for each pixel or region, the slice that contributes the sharpest information.

When Focus Fusion is selected, the MCAM software executes the following steps:

1. Stack Dimension Validation. Focus fusion currently supports:
 - a. Stack dimension: XYZ or time-resolved stacks only.
 - b. The dataset must be GPU-compatible.
 - c. CUDA-enabled GPU and PyTorch must be available.

If these conditions are not met, the operation fails with a validation error.

2. Multi-Scale Decomposition. Each slice in the stack is decomposed into multiple spatial frequency levels. Low-frequency levels capture coarse structures, and high-frequency levels capture fine edges and texture. This multi-scale representation allows focus estimation to be robust across feature sizes.
3. Local Energy Computation. For each slice and each scale level, local spatial energy is computed over small neighborhoods. Energy reflects the strength of local gradients and edges, where higher energy corresponds to better local focus. This produces an energy map:

$$E(x, y, z, l)$$

where l is the scale level.

4. Per-Pixel Focus Selection. For each pixel location (x, y) , the algorithm compares local energy values across all slices. At each scale, the slice with the maximum local energy is identified. These selections are then combined across scales to determine which slice contributes that pixel. This means:
 - a. Pixel A may come from slice 3
 - b. Pixel B may come from slice 7
 - c. Pixel C may come from slice 5
 all within the same FOV.
5. Fusion and Reconstruction. Once the optimal slice contributions are determined, pixel values are copied directly from their source slices. No averaging or intensity blending is performed. The final image is reconstructed by combining these locally selected pixels. This preserves true intensities, sharp boundaries, and fine structural details of images.



6. Metadata Handling. After fusion, the stack dimension is retained internally only for indexing. A representative slice index (middle slice) is recorded in metadata for compatibility. The output dataset contains a single fused image per FOV per channel (Fig 108). The fused image behaves as a standard 2D MCAM dataset.

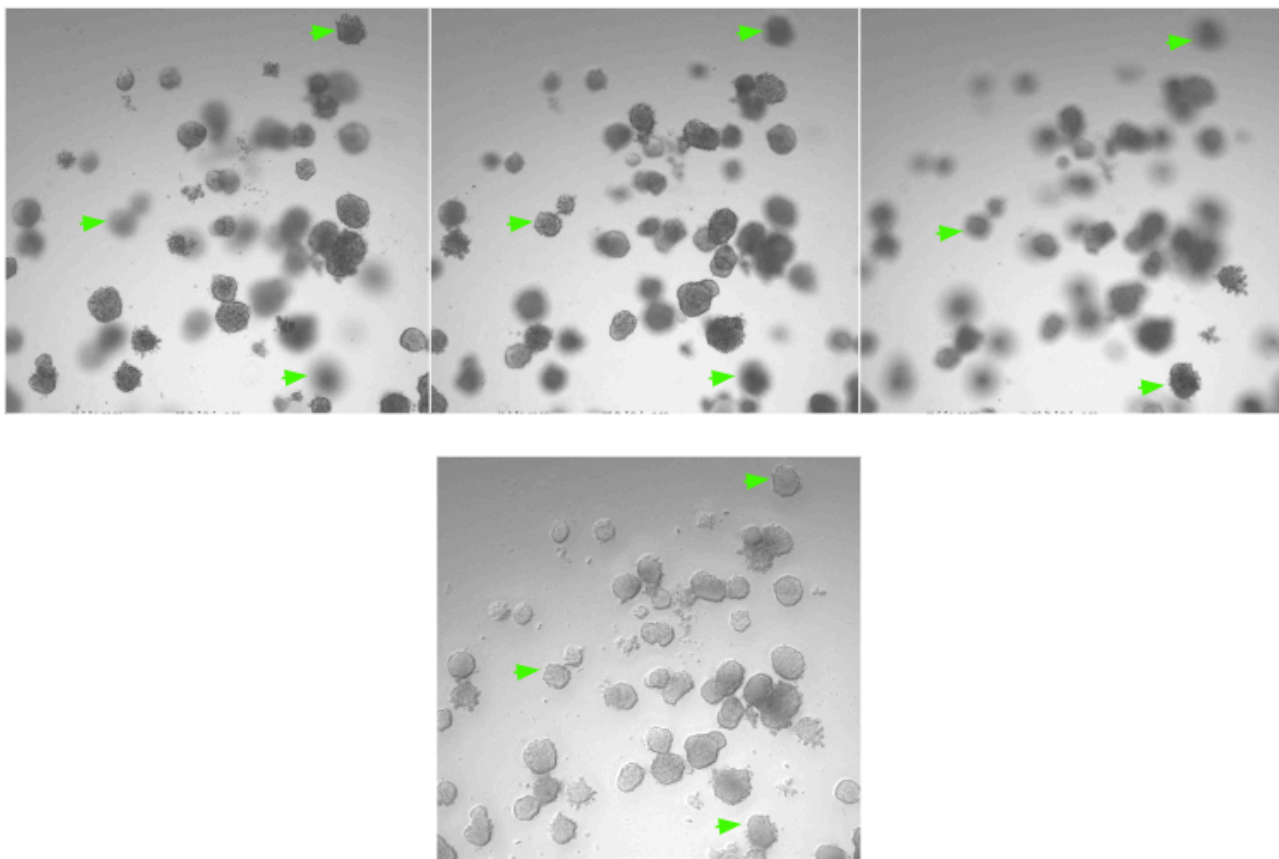


Fig 108. Organoids grown in Matrigel dome, with organoids in- or out-of-focus at different depths that are 300 μ m apart between each depth (green arrowheads, top); focus fusion-derived single-frame image, with all organoids in-focus (green arrowheads, bottom).

Content aware focus projection

This method generates a single representative image from an XYZ stack by using a trained deep-learning model to determine which slice (or region) is most relevant for the specimen being imaged. Instead of relying solely on generic image statistics (e.g., intensity or gradients), this approach leverages semantic understanding of the sample to guide focus selection.

Compared to traditional projections where they answer the question “Which slice is brightest or sharpest?”, model-based projection answers a different question “Which slice best represents the biological object of interest?” This is accomplished by running a trained object-detection or localization model on stack slices, identifying where the target specimen appears most clearly, and selecting or fusing slices based on model confidence, not raw image statistics.

Model-based projection is intended for application-specific specimens whose optimal focus cannot be reliably determined using classical metrics alone (e.g., zebrafish embryos, organoids). Because selection is driven by a



trained model, model-based projection can outperform Laplacian or intensity-based methods when the specimen has complex morphology, low contrast, and confounding background structure.

The MCAM software currently supports multiple object detection-based focus models, each trained for a specific specimen type:

- Zebrafish projection
- Tiny organoid projection

Each model encodes prior knowledge about the expected object size and shape, the texture and contrast characteristics, and typical focal appearance across Z range.

When a model-based projection is selected, the MCAM software executes the following steps:

1. **Stack Dimension and GPU Validation.** Model-based projection operates on stacked datasets. This method requires GPU acceleration and uses a trained neural network model loaded at runtime. If GPU support is unavailable, model-based projection cannot be executed.
2. **Per-Slice Inference.** For each slice in the stack, the image is passed through the trained object detection model. The model performs object detection or localization. Detection outputs include confidence scores, bounding boxes or spatial masks, and presence/absence of the target specimen. This produces a per-slice relevance score indicating how well the specimen is represented in that slice.
3. **Focus Score Aggregation.** For each FOV, model outputs are aggregated across slices. The slice with the highest detection confidence (or best localization quality) is then selected. If no valid detection is found, fallback behavior may apply (e.g., default slice or no projection). This selection is semantically driven, meaning the chosen slice is the one where the biological object is most clearly identifiable to the model.
4. **Optional Multi-Instance Handling.** Depending on the model configuration, single-instance mode selects one dominant specimen per FOV. Multi-instance mode may evaluate multiple detections but still selects a single optimal slice. For example, the tiny-organoid model is configured for single-instance selection, ensuring consistent projection behavior across the array.
5. **Slice Extraction and Stack Collapse.** Once the optimal slice index is determined, the corresponding slice is extracted from the original dataset without performing averaging, normalization, or pixel blending. The stack dimension is removed from the output dataset. A metadata field records the selected slice index for traceability.

Once the user clicks Run Projection, the MCAM software executes the following steps:

1. **Stack Dimension Identification.** The software automatically detects the stack dimension and validates that it is compatible with the selected projection method.
2. **Per-Slice Processing.** Each slice is loaded sequentially, and intermediate statistics are accumulated as defined by the selected projection.
3. **Projection or Slice Selection.**
 - a. For true projections (max, min, mean, sum, std), the stack is mathematically collapsed.
 - b. For focus-selection methods, the optimal slice index is identified and extracted.
4. **Metadata Preservation.** The software records the selected or representative slice index and collapses the stack dimension while preserving spatial alignment and FOV indexing.
5. **Output Assembly.** The resulting projected images are assembled into a new MCAM dataset, maintaining channel structure, image layout/indexing, and coordinate consistency.

Focus Reconstruction



This method reduces an image stack into a single 2D image by first identifying the approximate in-focus plane using minimum intensity variance, then reconstructing enhanced contrast from the two neighboring defocused images. For transparent, unstained samples, such as 2D cell culture, the in-focus plane is assumed to produce relatively little intensity variation, while images immediately above and below focus produce stronger, approximately opposite intensity contrast.

When Focus Reconstruction is selected, the MCAM software executes the following steps:

1. Intensity Variance Calculation. The intensity variance $V(z)$ is calculated for each image along the Z-stack:

$$V(z) = \text{Var}[I_z(x, y, z)]$$

where:

- $I_z(x, y)$ is the image at Z position z ,
- x, y represent pixel coordinates,
- $V(z)$ is the intensity variance of that image.

The calculation is performed independently for each image tile. By default, images are downsampled by a factor of 4 during focus scoring to reduce computation.

2. Best-Focus Identification. The variance values across the Z-stack are smoothed using a moving-average window (default = 3). The software identifies the most prominent minimum-variance valley, representing the estimated in-focus plane. If no valley is detected, the global minimum of the smoothed variance curve is used.
3. Defocused Plane Selection. The two images immediately adjacent to the selected focus plane are retrieved:

$$\begin{aligned} z_{\text{under}} &= z_{\text{focus}} - 1 \\ z_{\text{over}} &= z_{\text{focus}} + 1 \end{aligned}$$

At least three images are therefore required in the stack.

4. Focus Reconstruction. The two defocused images are subtracted to produce the reconstructed image:

$$R(x, y) = I_{\text{over}}(x, y) - I_{\text{under}}(x, y)$$

Because transparent objects can produce opposite intensity changes on either side of focus, this subtraction enhances specimen contrast while suppressing image components that remain similar through focus, such as background and uneven illumination.

5. Contrast Normalization. The reconstructed image is contrast-stretched using global percentile limits (default 0.5%–99.5%), clipped to the valid output range, and converted to the original image data type. Using global limits keeps contrast consistent across different fields and wells.

Key idea: Minimum intensity variance is used to locate focus, but the in-focus image itself is not used as the final image. Instead, the images immediately above and below focus are subtracted to create an enhanced-contrast reconstruction.

All outputs are stored as derived datasets in the location selected by users. Results include a projected image dataset (one image per FOV), indexed metadata indicating the representative slice used for each image, and projection parameters and method identifiers for reproducibility. For focus-based methods, users may inspect per-image slice indices in the dataset tree to verify focus selection accuracy.

Example Stack Projection Workflow

STEP	ACTION
1	Load image stack
2	Calculate intensity variance for each slice
3	Smooth variance values and identify best focus plane
4	Select images immediately above and below focus plane
5	Subtract images to produce reconstructed image
6	Normalize contrast of reconstructed image
7	Save results as derived datasets



- 1 Open the acquired dataset in the MCAM Viewer software.
- 2 Go to Tools > Stack Projection. A panel will open on the right.
- 3 Click on Select Directory to choose the file location where the Stack Projection will be saved.
- 4 Select the intended projection method.
- 5 Choose whether the Stack Projection will be taken from all the frames (Full Array) or from a specific subset (Subset).
- 6 Click "Save" to run the projection.

Stack Reduction

The Stack Reduction tool reduces high-dimensional MCAM datasets into analysis-ready representations by applying a sequence of z-cropping and projection. It is designed to standardize preprocessing across all images in a dataset while preserving quantitative integrity and reproducibility. Stack Reduction operates on full MCAM-acquired datasets to generate 2D mosaics from volumetric data. The Stack Reduction tool can be accessed in the MCAM Viewer by navigating to Tools > Stack Reduction (Fig 109).

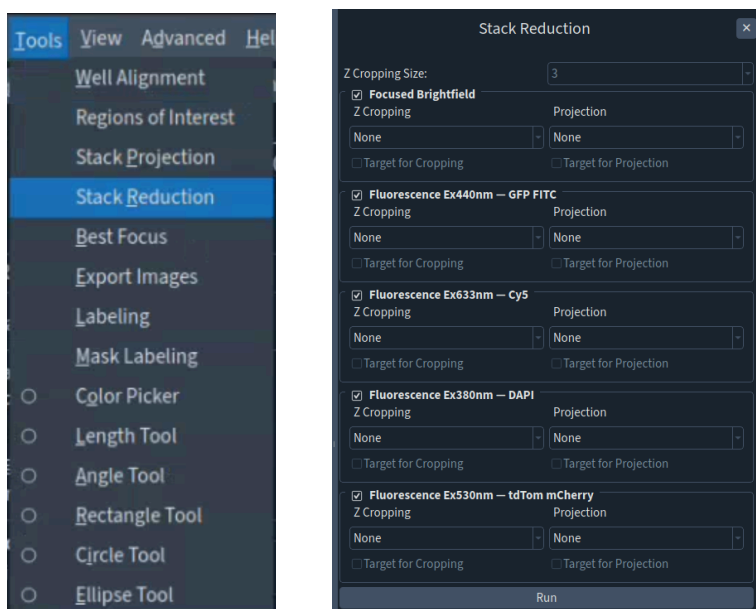


Fig 109: Tools menu with Stack Reduction highlighted (left); Stack Reduction panel (right).

Internally, Stack Reduction is implemented as a chained pipeline, where each processing step operates sequentially on a dataset. The active steps and their execution order are explicitly defined by the user through the Stack Reduction settings model. Only steps with enabled parameters are executed on selected channel(s).

The available processing steps are Z-cropping and projection, with z-cropping occurring before projection when both are enabled. This ensures that projections are computed only on the intended subset of the Z stack.

1. Z Cropping. Z cropping limits the number of focal planes used in subsequent processing by selecting a subset of slices around a focus position when enabled. This reduces noise, improves projection quality, and lowers computational load.

Users may specify:

- Cropping method (e.g., focus-based cropping).
- Cropping size, defined as the number of Z frames retained.
- Per-channel application, allowing cropping to be applied only to selected illumination modes.
- Target cropping, where the channel selected will be used as the target for all other channels in focus selection when enabled.

If no z-cropping method is specified, the full Z stack is passed unchanged to the next step.

2. Projection. Projection reduces a cropped (or full) Z stack into a single 2D image slice per image and channel. Stack Reduction supports multiple projection methods, including maximum/minimum intensities and Laplacian-based focus projections (See Stack Projection for more information). Projection options include per-channel projection mapping, allowing different projections for different illumination modes.

Projection is applied independently to each image, preserving MCAM's per-camera spatial alignment.

The results are saved in a folder named "projection_results_timestamp" in the provided dataset's folder with the metadata and the Tiff files containing the processed images.

Users are encouraged to inspect reduced images across wells to confirm focus quality and projection behavior before proceeding to downstream analysis.

Example Stack Reduction Workflow

STEP	ACTION
1	Open the acquired dataset in the MCAM Viewer software.
2	Tools > Stack Reduction. A panel will open on the right.
3	Optional: Enable Z cropping and set cropping size.
4	Select a projection method per channel.
5	Deselect channel(s) that are not needed.
6	Click "Run" to perform the analysis.

Transfection Efficiency Analysis

The Transfection Efficiency Analysis module quantifies the percentage of transfected cells within each field of view (FOV) of an MCAM-acquired dataset by combining AI-based cell segmentation from a brightfield image with fluorescence-positive detection. The assay is designed for experiments where fluorescent reporter expression in the cytoplasm (e.g., GFP, RFP, mCherry, or other fluorescent proteins) indicates successful transfection, enabling automated quantification across the Multi-Camera Array.

The analysis operates on either a single image or a full XYZC dataset. For image stacks, the MCAM software automatically selects the optimal focal plane using Laplacian-based focus selection before analysis begins,



ensuring that both brightfield and fluorescence channels are evaluated on the same focal slice. The Transfection Efficiency Analysis can be accessed in the MCAM Viewer by navigating to Assays > Transfection (Fig 110).

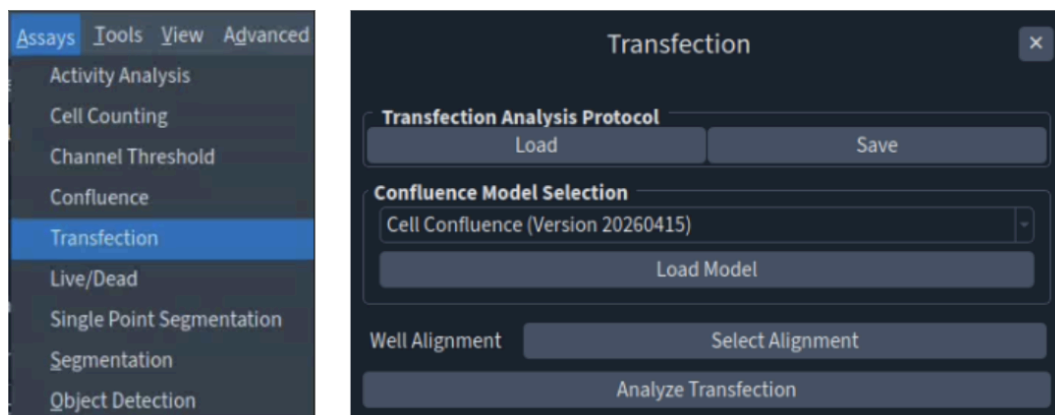


Fig 110: Assays menu (left); Transfection panel (right).

The Transfection panel allows users to configure the following parameters:

- Segmentation Model – Selects an appropriate trained neural-network model used to identify cellular regions from brightfield images.
- Well Alignment – Uses previously generated well keypoints to aggregate measurements at the well level instead of individual FOVs.
- Batch Analysis – Applies identical analysis settings to every dataset in the selected batch.

Transfection Efficiency Analysis combines two complementary measurements:

1. AI-derived segmentation of all cells from the brightfield image.
2. Automatic thresholding of each fluorescence channel to identify fluorescent-positive areas.

The overlap between these two masks determines the fraction of cellular area exhibiting fluorescent signal, reported as transfection efficiency.

Fluorescence-positive regions are identified using dynamic fluorescence thresholding on a per image basis. For every fluorescence channel and every FOV, the MCAM software automatically calculates an adaptive pixel intensity threshold. Pixels above this threshold are classified as fluorescence-positive.

Once the user clicks Analyze Transfection, the MCAM software performs the following steps:

1. Focus Determination. If the dataset contains multiple Z-planes or timepoints, the MCAM software performs Laplacian-based focus selection using the selected brightfield channel. The resulting focal plane is applied consistently to every fluorescence channel to ensure spatial alignment throughout the analysis.
2. Cell Segmentation. The selected segmentation model is loaded and executed on the brightfield images. The neural network generates an segmentation mask identifying cellular regions throughout each FOV. The resulting segmentation masks serve as the reference population for all subsequent transfection calculations.
3. Fluorescence Detection. Each fluorescence channel is analyzed independently using automatic dynamic thresholding. Pixels exceeding the calculated threshold are converted into binary fluorescence masks. Threshold values are stored for every fluorescence channel and every FOV to ensure reproducibility.
4. Transfection Mask Generation. The MCAM software combines the AI-generated cell mask with each fluorescence mask using a logical intersection:

$$\text{Transfected Region} = \text{Cell Mask} \cap \text{Fluorescence Mask}$$

Only fluorescence located within segmented cells contributes to the transfection measurement. Fluorescent artifacts located outside cells are automatically excluded. If stitching masks are available, overlapping image regions are weighted appropriately to prevent duplicate counting. If well masks are available, pixels outside the defined well boundary are excluded from analysis.

5. Transfection Efficiency Calculation. For each fluorescence channel, the MCAM software calculates:

$$\text{Transfection Efficiency (\%)} = \frac{\text{Cell areas positive for fluorescence}}{\text{Total cell areas}} \times 100$$

If no segmented cells are detected within a field of view, the reported efficiency is 0%. For datasets containing multiple fluorescence channels, transfection efficiency is calculated independently for each channel.

6. Well-Level Aggregation. When well alignment information is available, the MCAM software averages the transfection efficiencies from all valid FOVs belonging to the same well. This provides a single transfection efficiency measurement representing the entire well while preserving individual FOV measurements within the analysis dataset.

All outputs are saved in the transfection_results analysis folder. Two primary output files are generated: analysis_metadata.nc that contains cell segmentation masks, fluorescence threshold masks, combined transfection masks, transfection efficiency maps, dynamic fluorescence thresholds, analysis parameters, segmentation model information, instrument metadata; and transfection_results.csv, where it summarizes transfection efficiency for every field of view or, when well aggregation is enabled, every well, and includes headers for selected segmentation model, assay timestamp, and instrument metadata.

After analysis completes, the MCAM Viewer can display multiple visualization layers:

- Confluence Mask – AI-segmented cellular regions identified from the brightfield image.
- Threshold Mask – Fluorescence-positive pixels detected by automatic thresholding.
- Transfection Efficiency Labels – Numerical percentage displayed for each FOV or well.

These visualization layers affect display only and do not alter the underlying analysis results.

Example Transfection Efficiency Analysis Workflow

STEP	ACTION
1	Acquire a brightfield image together with one or more fluorescence channels containing a transfection reporter.
2	Open the dataset in the MCAM Viewer software.
3	Navigate to Assays > Transfection. A panel will open on the right.
4	Select the segmentation model for the bright-field channel.
5	Optional: Select a Well Alignment file specific to the well plate in use for multi-field per well datasets.
6	Click Analyze Transfection.

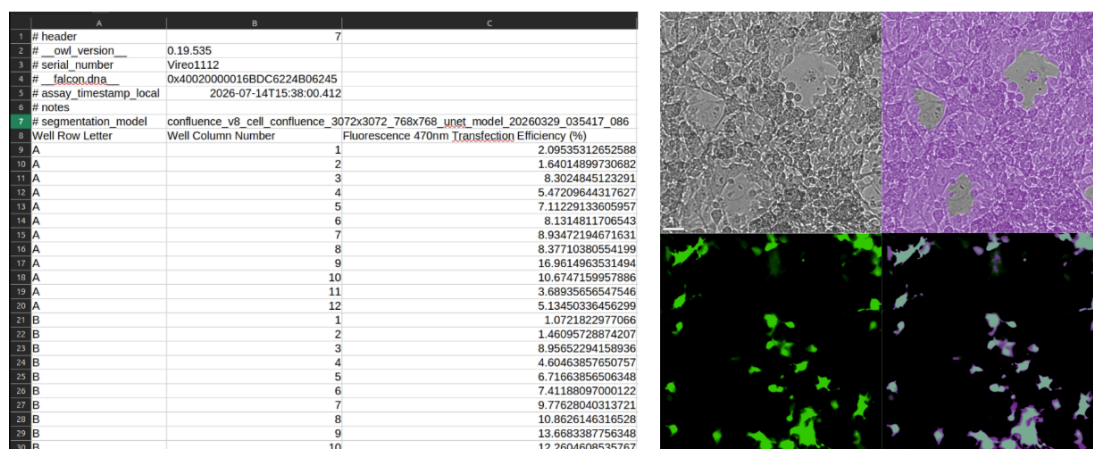


Fig 111: An example of a csv output (left), and binary masks (purple) overlaying segmented cell areas on the bright-field channel (top right) and GFP channel within the threshold range defined through dynamic fluorescence thresholding (bottom right). White scale bar: 50um.

Live/Dead Analysis

The Live/Dead Analysis module quantifies cell viability and cell death by combining AI-based brightfield segmentation with fluorescence-based live and dead staining. The analysis measures the total cell-covered area from the brightfield image and independently quantifies the areas occupied by live- and dead-stained cells from two fluorescence channels. The results provide a rapid assessment of cell viability on a per-field or per-well basis, making the assay suitable for cytotoxicity studies, drug screening, and cell health monitoring across the entire Multi-Camera Array.

The analysis operates on either a single image or a full XYZC dataset. For image stacks, the MCAM software automatically selects the optimal focal plane using Laplacian-based focus selection before analysis begins, ensuring that both brightfield and fluorescence channels are evaluated on the same focal slice. Results include segmentation masks for brightfield, live-stained, and dead-stained regions, quantitative area measurements, and calculated viability ratios. The Live/Dead Assay can be accessed in the MCAM Viewer by navigating to Assays > Live/Dead (Fig. 112).

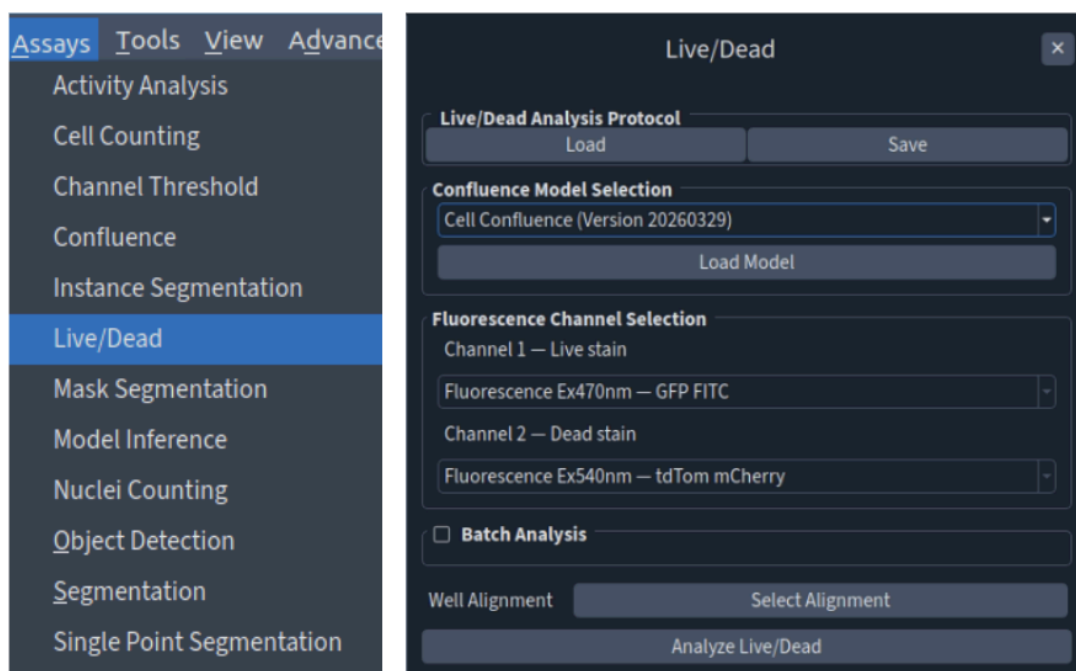


Fig 112: Assays menu (left); Live/Dead Assay panel (right).

The Live/Dead Assay panel allows users to configure the following parameters:

- Segmentation Model – Selects an appropriate pretrained neural network model to identify the cellular area in the brightfield image.
- Live Channel – Selects the fluorescence channel containing the live-cell stain. By default, the software automatically selects the first GFP/green fluorescence channel when available.
- Dead Channel – Selects the fluorescence channel containing the dead-cell stain. By default, the software automatically selects the first mCherry/red fluorescence channel when available.
- Well Alignment Keypoints – Used during result compilation to aggregate measurements into individual wells when imaging multi-field plates when applied.
- Batch Analysis – Applies identical analysis settings to every dataset in the selected batch.

The software requires the Live and Dead channels to be different fluorescence channels. Selecting the same channel for both measurements is not permitted.

Brightfield images are segmented using a pretrained neural network model. Live and dead fluorescence segmentation are identified using dynamic fluorescence thresholding. For every fluorescence channel and every FOV, the MCAM software automatically calculates an adaptive pixel intensity threshold. Pixels above this threshold are classified as fluorescence-positive. Dynamic thresholding is performed independently for both the Live and Dead fluorescence channels, allowing each stain to adapt to variations in staining intensity or illumination across the dataset.

Once the user clicks Run Live/Dead Analysis, the MCAM software performs the following steps:

1. Channel Extraction and Focus Selection. The selected brightfield, live fluorescence, and dead fluorescence channels are extracted. If the dataset contains a Z-stack, the software performs Laplacian slice-selection to automatically identify the best-focused image before analysis.
2. Brightfield Segmentation. The selected brightfield image is processed using the pretrained neural network segmentation model to generate a binary mask representing the cellular area.



3. Fluorescence Processing. The Live and Dead fluorescence channels are independently background-corrected, dynamically thresholded, and converted into binary masks representing live- and dead-stained regions.
4. Area Measurement. The software calculates the area occupied by:
 - Brightfield cellular region
 - Live-stained fluorescence
 - Dead-stained fluorescence
5. Well Compilation and Stitching Correction. For multi-field acquisitions, individual image fields are stitched into complete wells. Overlapping image regions are removed to prevent duplicate measurements, and optional well masks restrict analysis to the interior of each well.
6. Result Generation. Segmentation masks, quantitative measurements, analysis parameters, and metadata are assembled into the output dataset for visualization and export.

All analysis results are saved in the `live_dead_analysis_results` folder. Two primary output files are generated: `analysis_metadata.nc` that contains the brightfield, live, and dead segmentation masks, calculated area measurements, analysis parameters, selected segmentation model, and all metadata required for visualization and reproducibility; and `live_dead_results.csv`, where it summarizes measurements for every field-of-view or well, including brightfield area, live-stain area, dead-stain area, green area / brightfield area, red area / brightfield area, dead channel area / live channel area, instrument metadata, software version, selected live and dead fluorescence channels, segmentation model, measurement units, and assay timestamp.

After analysis completes, the MCAM Viewer displays multiple visualization layers:

- Brightfield Mask – AI-segmented cellular regions identified from the brightfield image.
- Live-Stain Mask – Fluorescence-positive pixels detected by automatic thresholding.
- Dead-Stain Mask – Fluorescence-positive pixels detected by automatic thresholding.
- Per-field or per well brightfield, green, red area labels.

The segmentation masks can be toggled independently using the Display Layers panel, allowing users to visually inspect segmentation and threshold quality before exporting results. Area measurements are displayed directly on the image, and users may inspect individual fields or wells to verify segmentation accuracy before re-running the analysis with modified parameters if necessary. These visualization layers affect display only and do not alter the underlying analysis results.

Example Live/Dead Assay Workflow

STEP	ACTION
1	Acquire images in brightfield, GFP, and mCherry channels.
2	Open the dataset in the MCAM Viewer software.
3	Navigate to Assays > Live/Dead. A panel will open on the right.
4	Select the segmentation model for the bright-field channel.
5	Select the Live and Dead fluorescence channels.
5	Optional: Select a Well Alignment file specific to the well plate in use for multi-field per well datasets.
6	Click Analyze Live/Dead.



	A	B	C	D	E	F	G	H
1	header							
2	#_owl_version	0.19.520						
3	#_serial_number	Vmeo1111						
4	#_biopon_00a	0x40020000010C20202430A205						
5	#_assay_timestamp_local	2026-06-16T15:31:25.501						
6	notes							
7	#_segmentation_model	confluence_vt_cell_confluence_307x3072_768x768_ugst_model_20200329_035417_086						
8	#_live_channel	fluorescence_440nm						
9	#_dead_channel	fluorescence_530nm						
10	#_units	mm²						
11	well_id	Plate Map						
12	A1	CTBL	3.3393348883875	0.896790747351907	0.0238233192292	0.268556509695048	0.00713684910742105	0.0265748505165883
13	A2	CTBL	3.45706212101504	0.884357291397464	0.017656404385952	0.255811802172014	0.00510734366943079	0.0199652386092667
14	A3	CTBL	3.47000013789511	0.909213724753499	0.0362836338752004	0.262021235914136	0.0104563782228579	0.0399066059908383
15	A4	CTBL	3.47221725860436	0.87852941987876	0.0342166117661991	0.2520168346182	0.00985439827573528	0.0389475992402066
16	A5	CTBL	3.46354549595691	0.845810465414389	0.0337413723627933	0.272054007776584	0.0097409893936647	0.035674220311878
17	A6	CTBL	3.4458998254018	0.93345727237377	0.0338619372541871	0.270889260128584	0.00982673272879827	0.036275829245004
18	A7	CTBL	3.4457432320778	0.924591518014495	0.0653118342458802	0.268328617584474	0.0152445663474158	0.0717201412233165
19	A8	CTBL	3.42642229087539	0.90501049496259	0.085737617000442	0.264126880470171	0.0300132136050199	0.075771233769004
20	A9	CTBL	3.4295842397114	0.91497093990256	0.0522998107044259	0.266787714180642	0.0152496066720982	0.05716007450137
21	A10	CTBL	3.23532117363357	0.897288316537015	0.0522069818487125	0.277341342136143	0.018136585342699	0.0581830621067258
22	A11	CTBL	3.33161960952566	0.840135101043416	0.036975968953566	0.252170175322935	0.011084861336967	0.044011932018395
23	A12	CTBL	3.3046214002332	0.8923346686039	0.076840296001019	0.27038669678042	0.0114705238217734	0.0454284181770949
24	B1	3.125uM_1000x800	3.41518193636148	0.867765663770115	0.0312457082429773	0.254090610673184	0.0091490611158232	0.036007080625674
25	B2	3.125uM_1000x800	3.4368599695191	0.810087669833592	0.028101352362024	0.235705753774114	0.00817646124715275	0.0346892730288994
26	B3	3.125uM_1000x800	3.45330365153364	0.82914244410849	0.045496813517825	0.242999990148205	0.0131612169603881	0.0541620575314906
27	B4	3.125uM_1000x800	3.42824682775245	0.841801238493645	0.0392611951838262	0.245548608600486	0.0114522661746517	0.0466395075090194
28	B5	3.125uM_1000x800	3.4325573778915	0.866893230977439	0.0352144340354244	0.252550252054327	0.0102589498611835	0.0406214200054595
29	B6	3.125uM_1000x800	3.45925195688324	0.843453278775996	0.034449693873859	0.243825338335847	0.00995688754769882	0.0408435301091703
30	B7	3.125uM_1000x800	3.409878617275244	0.850475373681547	0.103077738344837	0.248415142091932	0.03022915133962398	0.122100144688237
31	B8	3.125uM_1000x800	3.37282540385786	0.836717049423896	0.103741598422857	0.248076004308874	0.0307580695710478	0.123986470666535
32	B9	3.125uM_1000x800	3.40565429723938	0.850770941692746	0.0822743899675915	0.249799559039902	0.0241581742557496	0.0967102357930531
33	B10	3.125uM_1000x800	3.27547218146538	0.83948123119626	0.084142352151743	0.256293195204475	0.02568942912979	0.100231466984473
34	B11	3.125uM_1000x800	3.27305019709456	0.805709930773446	0.0531326840302881	0.246164856090263	0.0162333850172702	0.0509451770431613
35	B12	no_stain	3.24424627251574	0.0026554320643226	0.0234421406730689	0.00081850599282488	0.00722575868289149	8.82799477514974
36	C1	6.25uM_1000x800	3.09443839796586	0.862923400291038	0.0511440667504343	0.27886269502171	0.0165277378903565	0.0592683739277263
37	C2	6.25uM_1000x800	3.3502695254416	0.801319114570381	0.038677686394724	0.238841377994184	0.0118740545931387	0.0497152318621342
38	C3	6.25uM_1000x800	3.35994536726503	0.794329650943837	0.0701699818250745	0.238411478199247	0.02088426273496	0.0883386157644968
39	C4	6.25uM_1000x800	3.3365199697881	0.764579169754143	0.064212272832647	0.229154681320088	0.019245283549257	0.083983811451354
40	C5	6.25uM_1000x800	3.18552270073269	0.798799135282025	0.050258703963646	0.250759420353614	0.0157757062598328	0.0628117736996221
41	C6	6.25uM_1000x800	3.10354035588925	0.816714020857035	0.050730390158034	0.263154438871447	0.01635397751633329	0.0621417506247484
42	C7	6.25uM_1000x800	3.17322110277019	0.764527044339047	0.145303545195929	0.240930908870997	0.0457905517737422	0.190056775979126
43	C8	6.25uM_1000x800	2.99071348308644	0.754149425574724	0.141141399012668	0.252163716062976	0.0471932198824371	0.18715309489073
44	C9	6.25uM_1000x800	3.1049022861407	0.800655698185437	0.13664894312251	0.241854119146769	0.0412823736288341	0.17691215739775
45	C10	6.25uM_1000x800	3.13227542392269	0.755005601149605	0.12168906487729	0.241040617112807	0.0388500540400539	0.161176375754587
46	C11	6.25uM_1000x800	2.8400092781147	0.731313434698677	0.0689513655061091	0.25745025552115	0.0242735138889618	0.094284286647242

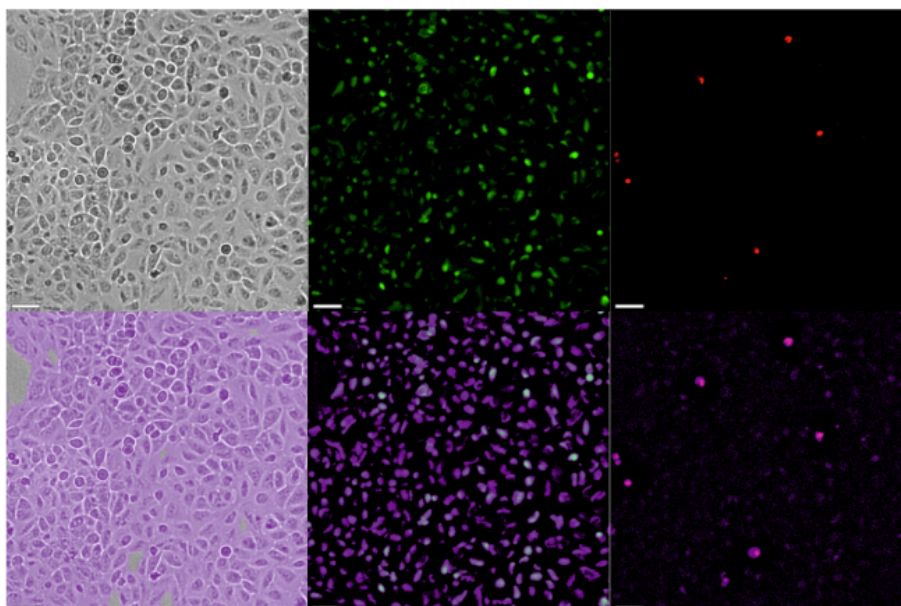


Fig 113: An example of a csv output (top), and binary masks (purple) overlying segmented cell areas on the bright-field channel (bottom left), GFP (bottom middle) and mCherry (bottom right) channels within the threshold range defined through dynamic fluorescence thresholding. White scale bar: 60um.

Fluorescent Nuclei Segmentation Analysis

The Nuclei Segmentation module identifies and quantifies individual nuclei in each well of a fluorescence dataset, such as cells stained with DAPI or Hoechst. Where the Cell Counting module detects bright spots and the Confluence module measures the fraction of the field covered by cells, Nuclei Segmentation performs *instance segmentation of nuclei*: a deep-learning model draws a separate mask around every nucleus, so touching,



clustered and irregularly shaped nuclei are separated and measured individually. The analysis operates on either a single image or a full XYZC stack (using automatic focus-slice selection). Results include a mask and a bounding box for every detected nucleus, and per-well nucleus counts, total nuclear area and average nucleus area. The Nuclei Segmentation assay is available on the Vireo and can be accessed in the MCAM Viewer by navigating to Assays > Nuclei Segmentation (Fig 114).

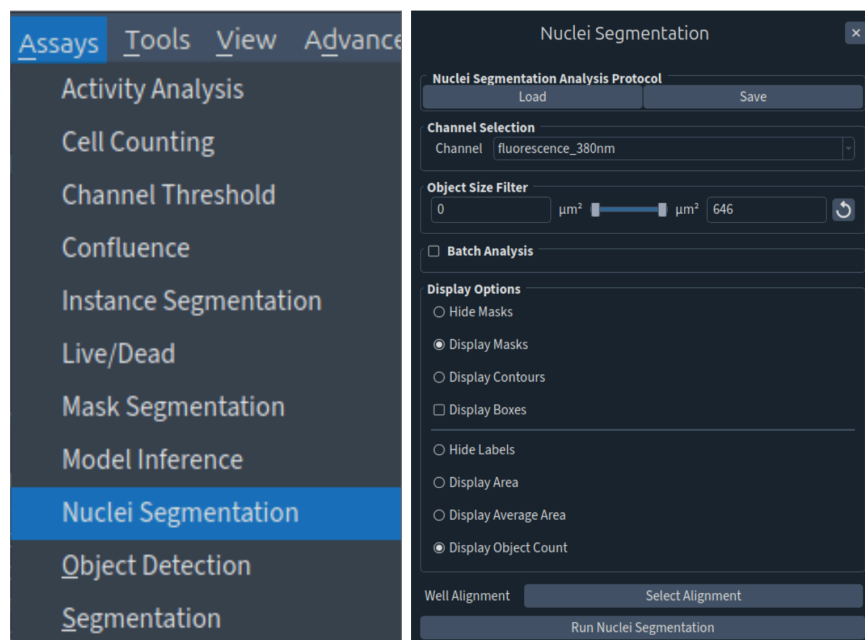


Fig 114: Assays menu (left); Nuclei Segmentation Assay panel (right).

The Nuclei Segmentation panel allows users to select the analysis channel, and the filters applied to the detected nuclei.

- **Channel Selection.** For multi-channel datasets, the channel the model is run on is selected here. Nuclei are segmented on a nuclear stain, so only fluorescence channels are offered; brightfield channels are never listed. The default follows the emission colour of the stain: a blue channel (DAPI, Hoechst) is preferred, then green, then red. A dataset that contains only brightfield channels leaves the selector empty and the Run button disabled.
- **Object Size Filter.** After the analysis has run, a two-ended slider sets the minimum and maximum allowed nucleus area (in μm^2 or mm^2 for calibrated datasets, pixels^2 otherwise). Nuclei outside the range are hidden from the masks, boxes and per-well labels immediately, without re-running the model. This can be used to reject small debris or large clumps of nuclei. Clicking "Save Changes" writes the filtered results to disk so the exported csv files match what is shown on screen.
- **Well Alignment.** Optionally, select a well alignment file for the well plate in use. Nuclei outside the physical well boundary are excluded from all counts and areas, and all fields of view belonging to the same well are pooled so the count, total area and average area are reported per well. If the dataset already carries well plate information, the well boundaries are applied automatically.



- **Batch Analysis.** Additional datasets can be added to the file list (or dragged onto the panel) and analyzed with the same settings one after the other.
- **Assay Protocol.** The channel and filter settings can be saved to a protocol file and loaded again later, so the same analysis can be repeated across experiments.
- **Display Options.** These settings affect display only; they do not change the underlying analysis results. Masks can be hidden, displayed as filled regions or displayed as contours; bounding boxes can be toggled; and the per-well label can show the nucleus count, the total nuclear area, or the average nucleus area, or be hidden.

The nuclei segmentation model is an embedding-based instance segmentation network. A conventional segmentation network answers one question for each pixel, is this pixel part of a nucleus or not, which is enough to measure coverage but not to count, because two nuclei that touch become one connected region. The nuclei segmentation network instead learns *which* nucleus each pixel belongs to.

The network is a U-Net style encoder–decoder. The encoder progressively downsamples the image to capture context, and the decoder upsamples back to full resolution, so every output pixel is predicted at the resolution of the input. For each pixel the network predicts three things:

1. An embedding. A small vector that places the pixel at a location in a learned coordinate space. During training the network is rewarded for mapping all the pixels of one nucleus onto the same point, the nucleus centre, and pixels of different nuclei onto different points. In effect each pixel votes for where the centre of its own nucleus lies.
2. A seed map. A per-pixel score of how likely that pixel is to be the centre of a nucleus.
3. A size estimate (sigma). How far, in the embedding space, a pixel may sit from a centre and still belong to that nucleus. A large nucleus tolerates a larger spread than a small one, so this is learned per object rather than fixed.

Turning these maps into a labelled image is a deterministic post-processing step that also runs on the GPU:

- Seed detection. Local maxima of the seed map above a fixed score are taken as candidate nucleus centres.

One seed is kept per neighbourhood so a single bright nucleus does not produce several candidates.

- Pixel assignment. For each seed, a window is cropped around it and every pixel in the window is scored by how close its embedding lies to the seed's embedding, scaled by the predicted sigma. Pixels whose score exceeds 0.5 are assigned to that nucleus. Because the comparison happens in the learned embedding space rather than in image space, two nuclei that are adjacent in the image but map to different centres are separated cleanly.
- Clean-up. Small fragments and objects whose pixels were assigned with low confidence are dropped, holes inside a nucleus are filled, and pairs of candidate objects that overlap by more than half their area (two seeds that landed in the same nucleus) are merged into one.



Before the image is passed to the network its intensities are rescaled so that the 0.1th percentile maps to zero and the 99.9th percentile maps to one. This makes the model insensitive to exposure and stain brightness, so the same model works across plates imaged with different settings. The one consequence of this normalization is discussed under *Signal filtering* below.

Whole MCAM fields (3072×3072 pixels) are processed in a single pass at native resolution, which is the fastest path. Larger images are split into overlapping tiles, each tile is processed separately, and nuclei that straddle a tile boundary are matched and merged across the overlap so they are not counted twice.

The model is trained on MCAM images with hand-labelled nucleus masks. Training presents random crops of these images with augmentation (flips, rotations, intensity changes) and adjusts the network so that the embedding, seed and sigma predictions reconstruct the labelled nuclei. The trained model is compiled into a single portable package that includes the post-processing, so results are identical on the GPU and on the CPU.

Once the user clicks Run Nuclei Segmentation, the MCAM software goes through the following steps:

1. **Channel Extraction and Focus Determination.** The selected fluorescence channel is extracted from the dataset. If the dataset is a Z-stack or a timelapse, a Laplacian slice-selection projection picks the best-focus frame per tile so that segmentation operates on a uniformly focused mosaic.
2. **Model Loading.** The model is loaded with GPU acceleration enabled by default and automatic fallback to CPU execution if a compatible GPU is not available or runs out of memory mid-run.
3. **Nucleus Segmentation.** The model is applied to each valid field of view in well order, producing a label image in which every pixel carries the number of the nucleus it belongs to (0 for background). Labels are renumbered so that a field with N nuclei holds exactly the labels 1 to N. As each field completes its masks are drawn on the screen in real time.
4. **Well Masking.** If well alignment keypoints are available, every mask pixel outside the physical well area is set to background, so nuclei and debris outside the well never contribute to the outputs. Nuclei cut by the well boundary are measured over the part that lies inside the well.
5. **Measurement.** For each field the software computes the nucleus count, the total nuclear area (the number of mask pixels times the calibrated pixel area), the average nucleus area (total area divided by count), the area of every individual nucleus, the signal of every nucleus (see *Signal filtering*), and one bounding box per nucleus in full-sensor coordinates. Areas are stored in m^2 for calibrated datasets, or pixels^2 otherwise.
6. **Filtering.** The signal filter is applied, followed by any object size limits. Every per-nucleus output is compacted together so that the masks, areas, signals, boxes and counts always describe the same set of nuclei.
7. **Output Assembly.** The masks, per-well labels, per-nucleus tables and bounding boxes are stored in the analysis results dataset together with the analysis parameters (model name, channel, filter values, well keypoints) for reproducibility and metadata export.

Signal Filtering

Because the model normalizes the intensity range of every image before it runs, it has no notion of absolute brightness. In a well that holds no stain at all, the few detections of camera noise are stretched across the full range and the model returns hundreds or thousands of confident but spurious “nuclei”. Suppressing these before inference is not possible, because the normalization simply re-stretches whatever remains, and blanking the image would move it away from the data the model was trained on. The rejection therefore happens *after* segmentation, per nucleus, on a measurement the model never sees.

For every detected nucleus the software measures its mean intensity above the background of its own field, where the background is the median intensity of the pixels that hold no nucleus. The result is expressed as a fraction of the camera’s full-scale range. By default a nucleus must sit at least 5% of full scale above its background to be kept. On test plates this rejected every object in the unstained wells while keeping more than 99% of the nuclei in the stained wells: real nuclei, even faint ones, were several times brighter than the brightest false detection. The measured signal of every nucleus is exported in the individual csv beside its area.

Outputs

All outputs are saved in the nuclei_segmentation_analysis_results folder inside the dataset’s folder. Three key files are generated (Fig 115, top):

- analysis_metadata.nc, which contains the nucleus masks, per-field counts, total and average areas, per-nucleus areas and signals, bounding boxes, and the analysis parameters and model name.
- nuclei_segmentation_well_summary.csv, which summarizes the Object Count, Total Area and Average Object Area on a per-well (or per-field-of-view) basis, and includes headers for the model used, assay timestamp and instrument metadata. Each area column is reported in the unit that suits its magnitude (μm^2 or mm^2) and the unit is named in the column header; uncalibrated datasets report pixels².
- nuclei_segmentation_individual_summary.csv, which lists one row per nucleus with its well, its Object Area and its Object Signal.

When the analysis completes, nucleus masks appear as overlays on the image and a count label appears in the corner of each well. The Display Options group box switches between filled masks, contours and hidden masks, toggles bounding boxes, and selects which per-well value is shown (count, total area or average area). Per-well area labels share one unit across the plate so they can be compared at a glance. Users are encouraged to inspect representative wells and tiles to verify segmentation quality, adjust the size filter if needed, and re-run the analysis if required (Fig 115, bottom).

Example Fluorescent Nuclei Segmentation Assay Workflow

STEP	ACTION
1	Capture an XYZC stack at 10x magnification in a nuclear stain fluorescence channel (for example 380nm for DAPI or Hoechst).
2	Open the dataset in the MCAM Viewer software.
3	Navigate to Assays > Nuclei Segmentation. A panel will open on the right.
4	For multi-channel datasets, select the nuclear stain channel under Channel Selection. The blue (DAPI / Hoechst) channel is selected by default. Brightfield channels are not offered.

- 5 (Optional) Under Well Alignment, select the alignment file for the well plate in use. Nuclei outside the well boundary are then excluded. If the dataset already contains well plate information this is applied automatically.
- 5 Click "Run Nuclei Segmentation" to run the analysis. A progress bar shows the wells as they are processed; click "Stop Analysis" to cancel.

	A	B	C	D	E	F
1 # header		7				
2 # __owl_version__	0.19.601					
3 # serial_number	Vireo1112					
4 # __falcon_dna__	0x40020000016BDC6224B06245					
5 # assay_timestamp_local	2026-09-09T09:39:53.726					
6 # notes						
7 # nuclei_counting_segmentation_model	cell_segmentation_3072x3072_3072x3072_instanse_model_20260828_175640_954.zip					
8 Well Row Letter	Well Column Number	Plate Map	Object Count	Total Area (mm²)	Average Object Area (µm²)	
9 A		1 A549-Hoechst	9598	1.17730057239532	122.661018371582	
10 A		2 A549-Hoechst	9610	1.20750689506531	125.651084899902	
11 A		3 A549-DAPI	10169	1.28601133823395	126.463897705078	
12 A		4 A549-DAPI	10057	1.30828177928925	130.086669921875	
13 A		5 A549-NucSpot488	0	0		
14 A		6 A549-NucSpot488	0	0		
15 A		7 HT1080-Hoechst	7791	1.13878357410431	146.166549682617	
16 A		8 HT1080-Hoechst	7653	1.13286828994751	148.029327392578	
17 A		9 HT1080-DAPI	8005	1.17966425418854	147.365921020508	
18 A		10 HT1080-DAPI	7694	1.13226902484894	147.16259765625	
19 A		11 HT1080-NucSpot488	0	0		
20 A		12 HT1080-NucSpot488	0	0		
21 B		1 A549-Hoechst	6575	0.876273930072784	133.27360534668	
22 B		2 A549-Hoechst	6406	0.877188742160797	136.932373046875	

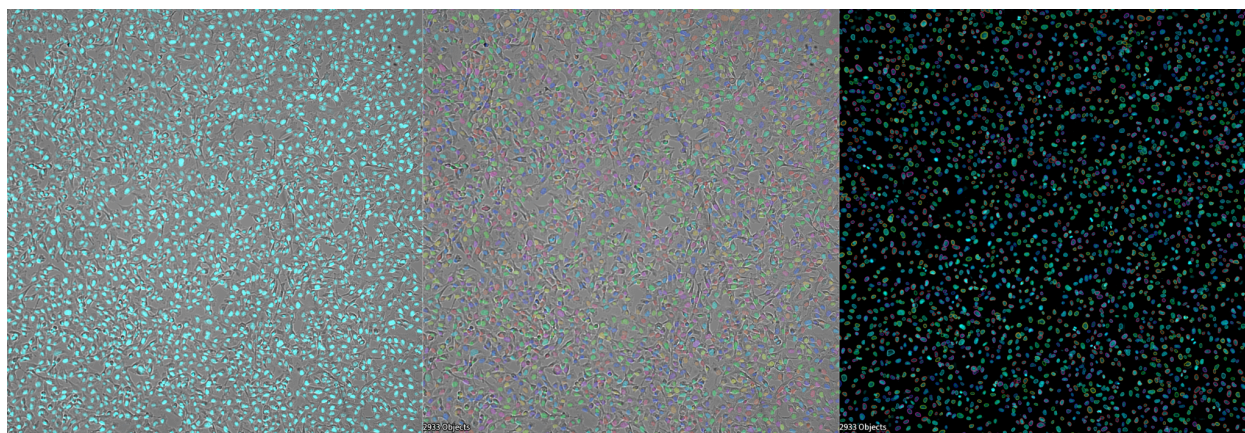


Fig. 115: An example of a well summary csv output (top), and cells with DAPI stained nuclei and nuclei counting analysis (bottom). DAPI stained nuclei (bottom left) are individually segmented and visualized with multi-color masks overlaying each segmented nucleus on the bright-field channel (bottom center), and multi-color contours encompass the DAPI stained nuclei in the blue channel (bottom right). Object count labels display the number of nuclei found. White scale bar: 70µm.



Loading your data with other software

Often it might be necessary to load the raw data in another programming language for custom analysis. In the MCAM Viewer, Ramona Optics applies many image corrections which are not automatically applied when loading data using the manual data loading functionalities described below.

Python®

For our supported platforms, we recommend you utilize the "mcam_data.load" and "mcam_data.save" functions to read in the MCAM's data, along with the available metadata into an Xarray DataArray. However, should you need to export the data as individual camera images, we provide the functions "mcam_data.export".

ImageJ

ImageJ accepts both ".jpg" and ".tiff" file types (as well as others) so we recommend exporting images from the GUI into one of these formats for use with this software. ImageJ documentation is available at:
<https://imagej.net/ij/docs/guide/user-guide.pdf>.

Matlab®

Ramona Optics MCAM data is stored in a netcdf4 compatible data format. Data and metadata can be read using HDF5 functions in Matlab®. Please follow Matlab's® documentation to learn more about how to read the data:
<https://www.mathworks.com/help/matlab/ref/h5read.html>



R

The [CRAN Package ncdf4](#) can aid in loading the raw data in R.

```
library('ncdf4')
nc <- nc_open('mcam_gigapixel_microscopy_image_20201103_134639_857.nc')
raw_data <- ncvar_get(nc, 'mcam_data',
                      start=c(1, 1, 1, 1), count=c(-1, -1, 1, 1))
```

Loading Image Files and EXIF Orientation

EXIF (Exchangeable Image File Format) orientation is a type of image rotation that is encoded in the metadata of digital images. EXIF is a standard format used by many digital cameras and other imaging devices as well as the MCAM. EXIF orientation specifically refers to the rotation of an image that has been taken with a camera held in a different orientation than the default. For example, if you hold your camera sideways to take a portrait photo, the resulting image will appear rotated 90 degrees when viewed on a computer screen. EXIF rotations are important because they ensure that your photos appear in the correct orientation when viewed on different devices. Without EXIF rotations, portrait photos would always appear sideways on computer screens, which can be confusing and frustrating for viewers. By using EXIF rotations, the image is automatically corrected so that it appears as it was intended to be viewed. The various EXIF configurations and their respective integer flags are shown below (Fig 114).

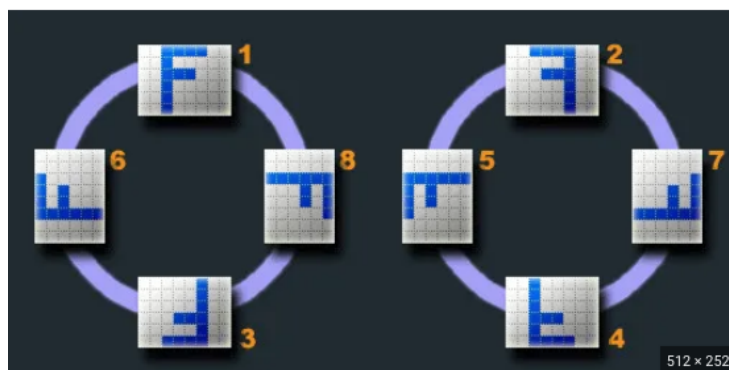




Fig 114: EXIF orientation diagram.

The Kestrel currently uses an EXIF 8 orientation when it is configured in the upright imaging configuration. This orientation can change depending on the user case in the Advanced Settings menu. This may be important to keep in mind during analysis because the orientation can make it seem like coordinates are rotated, or inverted. For example, for EXIF 8, images are displayed with a rotation 90 degrees counterclockwise from how they are acquired so that they are oriented as expected to the user. Quantities that refer to the coordinate system of the image data, such as the tracking data outputs, specifically angles and coordinates, remain in the coordinate frame of raw data. Therefore there is likely a 90 degree rotation between displayed images and videos and the resulting tracking data. The MCAM software will take this into account when generating final outputs and ensure that all analysis remains consistent.

If you are attempting to load the MCAM data as part of an external pipeline, be sure to read the `exif_orientation` tag in the metadata of the dataset (stored in the associated `metadata.nc` or `metadata.json` file) to orient the images correctly for display.

TIFF Files

Due to the large size of the ".tiff" images, we recommend viewing them in either Adobe Photoshop or GNU Image Manipulation Program (GIMP). GIMP can be found here: <https://www.gimp.org/downloads/>

Detailed Technical Performance Limits

MINIMUM EXPOSURE TIME	APPROXIMATELY 160 MICROSECONDS
Maximum frame rate with region of interest selection and binning.	345 frames per second

Appendices

Appendix A: Maximum Recording Time

To ensure maximum transfer rate between the MCAM and the workstation, we included a workstation with 128 GB of RAM. Each video recording session is first stored to RAM, and then transferred to the included solid state drive (SSD) for permanent storage. The information below represents the longest video recording permitted before a pause must be taken in order to transfer the data from RAM to permanent storage on the SSD.

The table below is constructed assuming that 100 GB of the 128 GB of RAM is used for video recording.

Kestrel2850 -- Well plate Imaging

	MEGAPIXELS PER SNAPSHOT	FRAME RATE (FRAMES PER SECOND (FPS)) TOTAL FRAMES (COUNT) VIDEO RECORDING (MINUTES (MIN) SECONDS (SEC))		
		ULTRA RESOLUTION	HIGH RESOLUTION	STANDARD RESOLUTION
MICROCAMERA CONFIGURATION (FIELD OF VIEW)	ULTRA (HIGH) [STANDARD] RESOLUTION	MICROCAMERA PIXELS: (2048, 2048)	MICROCAMERA PIXELS: (1024, 1024)	MICROCAMERA PIXELS: (512, 512)
1 × 1 (18 x 18 mm)	4.2 (1.05) [0.25]	45 fps 25k frames 9 minutes 15 seconds	90 fps 100k frames 18 min, 30 sec	180 fps 400k frames 37 min
2 × 2 (36 x 36 mm)	16.77 (4.2) [1.05]	45 fps 6.25k frames 2 min, 20 sec	90 fps 25k frames 4 min, 20 sec	180 fps 100k frames 9 min, 15 sec
3 × 3 (54 x 54 mm)	37.7 (9.4) [2.35]	45 fps 2.8k frames 1 min, 3 sec	90 fps 11.2k frames 2 min	180 fps 45k frames 4 min, 10 sec
6 × 4 (108 × 72 mm)	100 (25) [6.3]	45 fps 1050 frames 23 sec	90 fps 4.2k frames 45 sec	180 fps 17k frames 1 min, 32 sec



Kestrel2850 -- Free-swim Imaging

	MEGAPIXELS PER SNAPSHOT	FRAME RATE (FRAMES PER SECOND (FPS)) TOTAL FRAMES (COUNT) VIDEO RECORDING (MINUTES (MIN) SECONDS (SEC))		
		ULTRA RESOLUTION	HIGH RESOLUTION	STANDARD RESOLUTION
MICROCAMERA CONFIGURATION (FIELD OF VIEW)	ULTRA (HIGH) [STANDARD] RESOLUTION	MICROCAMERA PIXELS: (2560, 2560)	MICROCAMERA PIXELS: (1280, 1280)	MICROCAMERA PIXELS: (640, 640)
1 × 1 (18 x 18 mm)	4.2 (1.05) [0.25]	30 fps 16k frames 8 minutes 50 seconds	75 fps 64k frames 14 min, 10 sec	140 fps 260k frames 30 min
2 × 2 (36 x 36 mm)	16.77 (4.2) [1.05]	30 fps 4k frames 2 min, 12 sec	75 fps 16k frames 3 min, 35 sec	140 fps 65k frames 7 min, 40 sec
3 × 3 (54 x 54 mm)	37.7 (9.4) [2.35]	30 fps 1.8k frames 1 min	75 fps 7.2k frames 1 min, 36 sec	140 fps 28k frames 3 min, 20 sec
6 × 4 (108 × 72 mm)	100 (25) [6.3]	30 fps 660 frames 20 sec	75 fps 2.6k frames 35 sec	140 fps 10.5k frames 1 min, 15 sec

Appendix B: Dataset size for extended recordings

When operating in high-speed mode (High Frame Rate, bin x4), extended recordings are possible and limited by the SSD size. Figure 116 can help estimate the size of the generated dataset generated in these operating modes. The Figure below can help users estimate the size of a recording on an SSD based on the chosen frame rate.

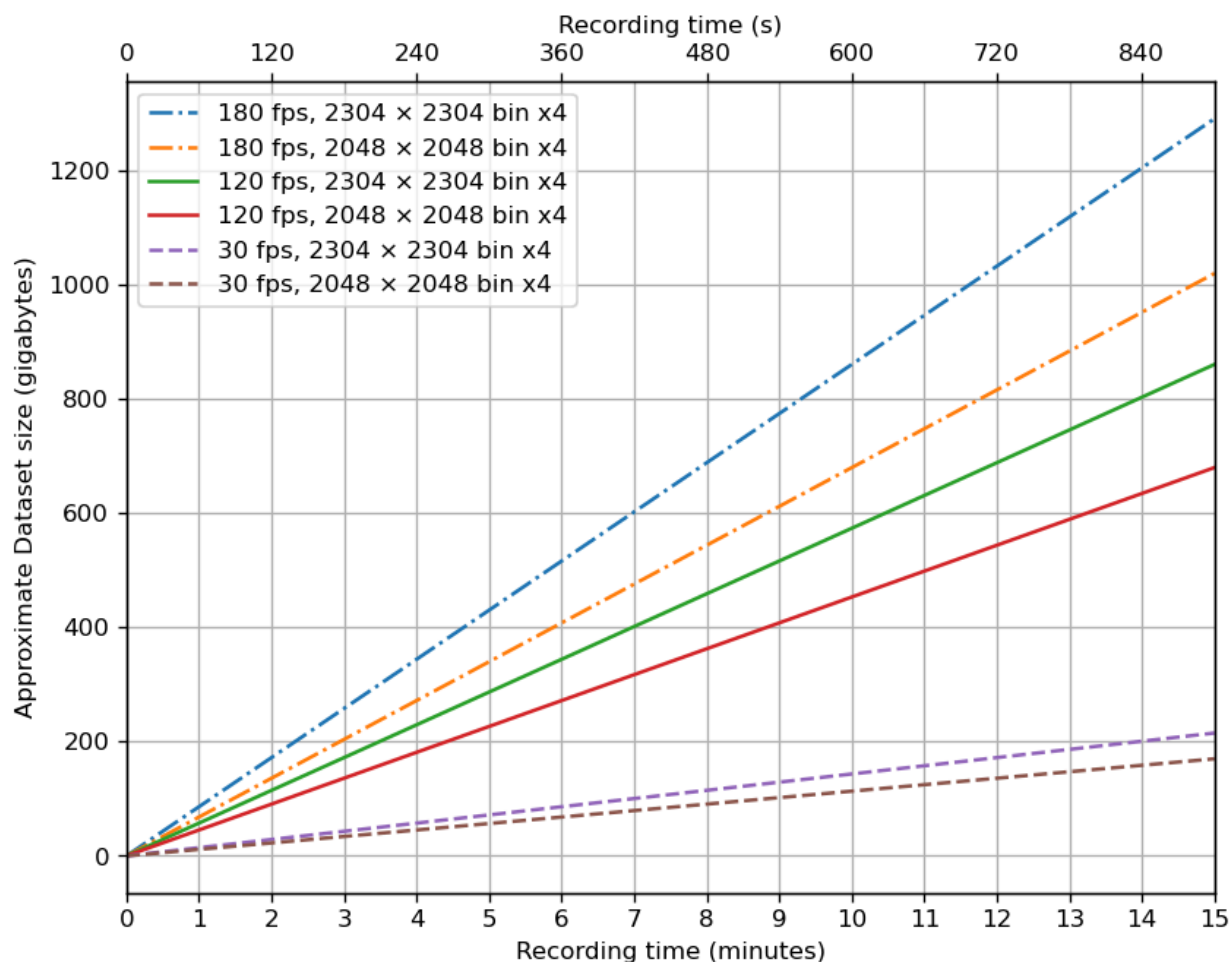


Fig 116: Plot of Approximate Data Set Size proportional to Recording Time.

Appendix C: System Disassembly by End User

High-Speed Data Cable Removal

The installation technician will assist with connecting the MCAM to the MCAM workstation, however the user may need to disconnect and reconnect the instrument for instance to move the workstation and MCAM. Instructions for successful connection are given below.

STEP	ACTION
1	Power down and then disconnect power cords from both the MCAM workstation and MCAM.
2	By pulling perpendicularly to the MCAM using the green flaps, disconnect the SFF-8644 cables (see Fig 117 for reference) from both the MCAM workstation and MCAM.

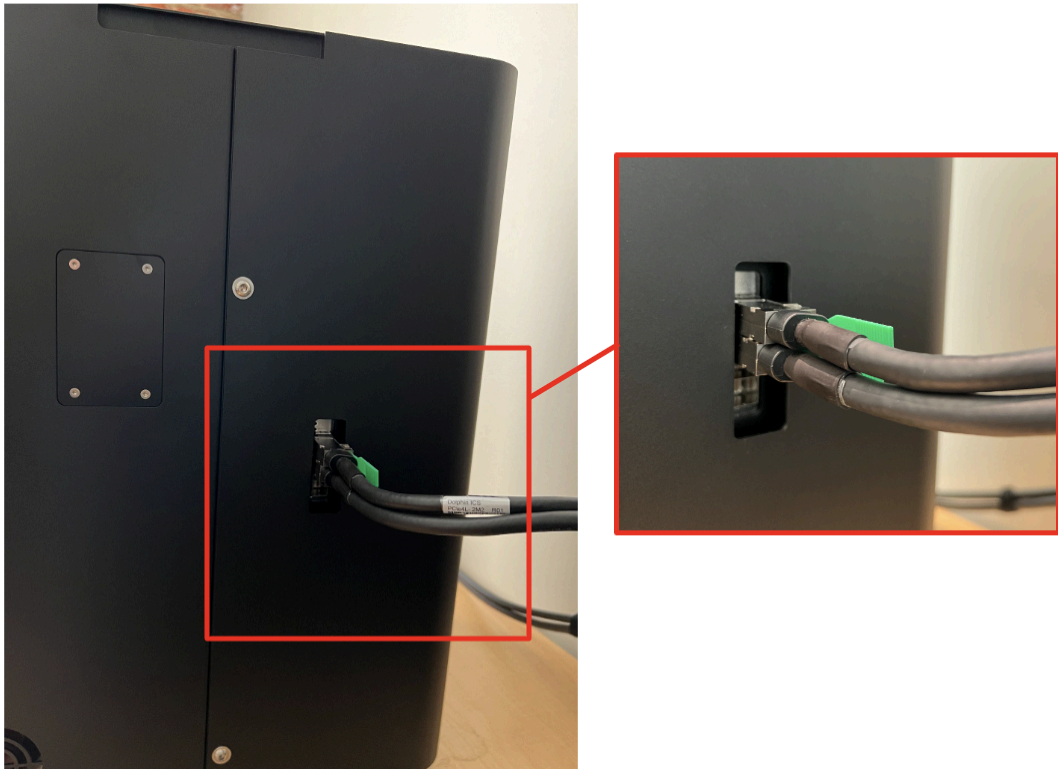


Fig 117: Dolphin cables with green flaps for removal.

PCIe Ribbon Cable Installation - Kestrel (2022 Revision)

The installation technician will assist with connecting the MCAM to the MCAM workstation using the PCIe Ribbon Cable, however the user may need to disconnect and reconnect the instrument for instance to move the workstation and MCAM. Instructions for successful connection are given below.

STEP	ACTION
1	Power down and then disconnect power cords from both the MCAM workstation and MCAM.

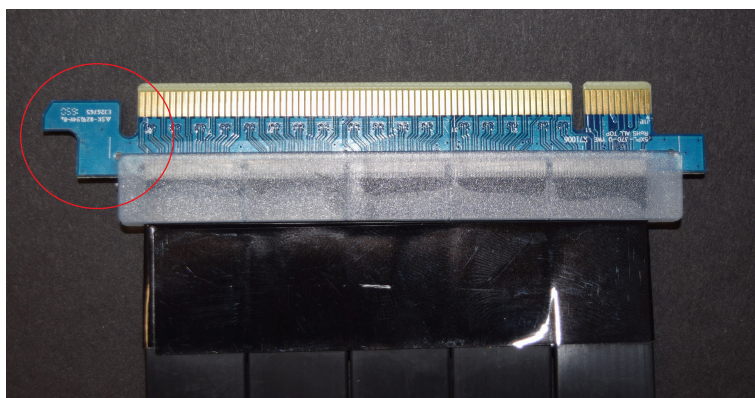


Figure 118: PCIe Ribbon Cable Workstation end with locking flap highlighted on the left side.

2	Align the male end of the PCIe Ribbon Cable with the port on the motherboard of the MCAM workstation with the locking flap on the left side (Figure 119). The PCIe port is located just below the graphics card and above and to the right of the wifi card in the workstation. Gently press the PCIe connector into the socket making sure the connector lock engages the locking flap on the left side.
---	---



Figure 119: Connecting the PCIe Ribbon Cable to the MCAM workstation with connector lock highlighted.

Reflection Illumination Board Installation -- Falcon (Discontinued)

This section is only applicable to MCAMs with 54 multi-cameras sold under the brand name Falcon, and not Kestrel.

The reflection illumination board can be mounted to the upper surface of the MCAM to enable reflection lighting. Follow these directions to install this board.

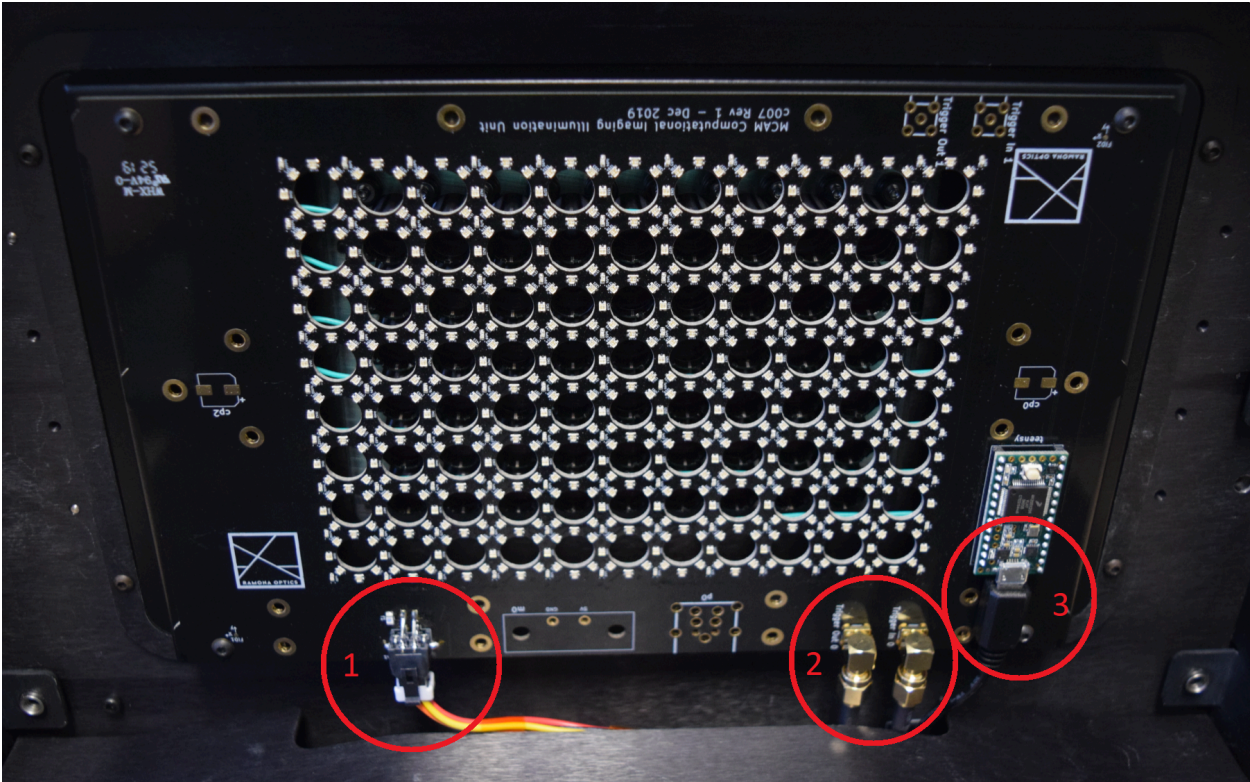


Figure 120: Mounted reflection illumination board on the upper surface of the MCAM with 1) Power source cable, 2) Trigger Out and In, and 3) Microcontroller Cable and Teensy Microcontroller highlighted.

STEP	ACTION
1	Power off and disconnect power from the MCAM.
2	Align the reflection illumination board with the upper surface of the MCAM.
3	Locate the reflection illumination board connection wires at the rear of the upper mounting surface.
4	Connect the power source cable (Figure 120, #1) to the corresponding port on the illumination board.



-
- 5 Connect the Trigger Out and Trigger In cables to their respective ports. Note that each cable is labeled In or Out and their ports are labeled on the illumination board (Figure 120, #2).
 - 6 Connect the microcontroller cable to the Teensy Microcontroller (Figure 120, #3).
 - 7 Mount the reflection illumination board to the MCAM by lining up the back edge of the board into the inset and gently pressing the board into the inset. The board will be held in place by magnets and should sit flush with the inset guide (Figure 121).
-

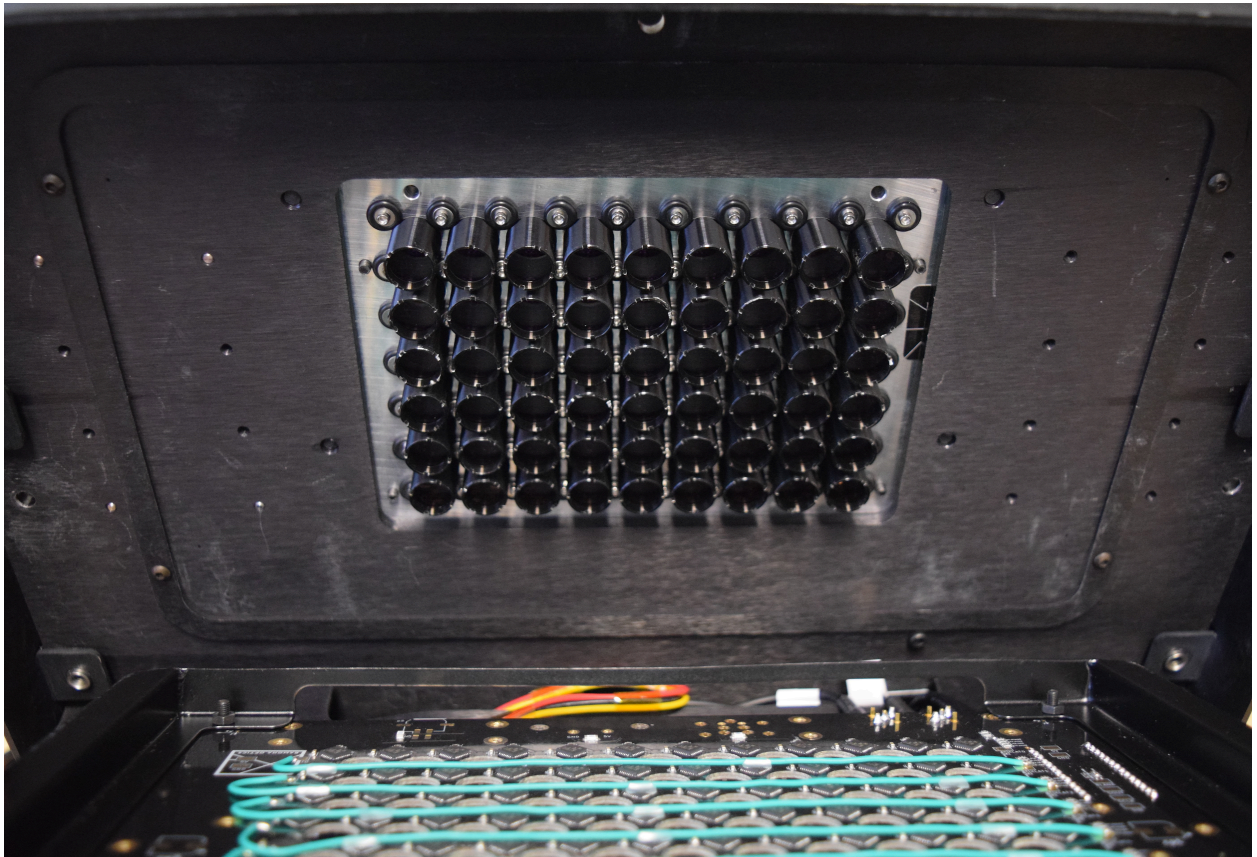


Figure 121: Upper surface of the MCAM showing the system optics and reflection board inset.



Appendix D: Trigger connectivity

Output Trigger

- Signal voltage level: 5 V
- Trigger pulse width: 1 ms (nominal)

The MCAM will output a pulse on the following events:

1. Start of frame or video acquisition.
2. At the start of image data transmission from the image sensors.

The width of each pulse takes on a value of:

$$\text{pulse width} = \min(1 \text{ ms}, \text{exposure} \cdot 0.4)$$

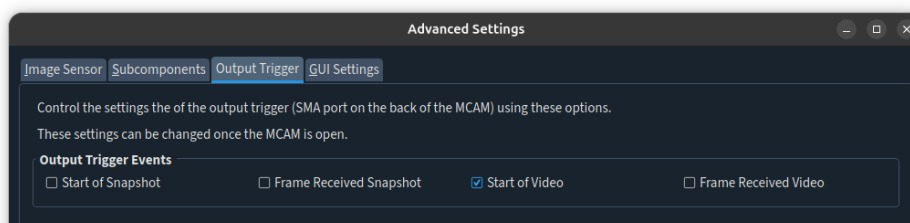
The following options are available in the GUI to help control when the output trigger is generated. They may be accessed after the MCAM is open and will be saved within the assay configuration files.

- Start of Snapshot: This will generate a pulse on the output port every time a new single frame is requested.
- Frame Received Snapshot: This will generate a pulse every time a frame is received by the electronics within the MCAM after requesting a snapshot.
- Start of Video: This will generate a single pulse at the start of every "video" acquisition.
- Frame Received Video: This will generate a pulse every time a frame is received by the electronics within the MCAM after a video is requested.

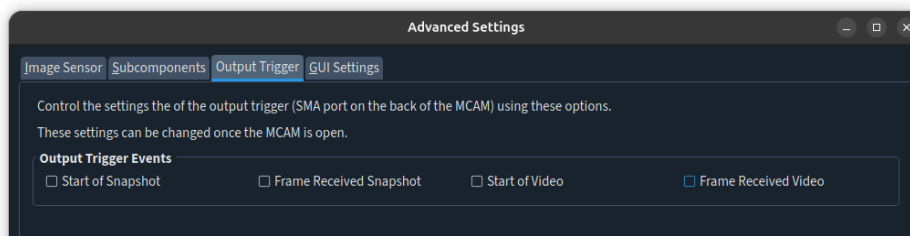
When the GUI is streaming, the hardware is capturing snapshots from the MCAM hardware, one at a time, as such if the options for generating triggers on snapshots are enabled, output triggers are expected when using the GUI.

APPLICATION NOTE - VIDEO

In many cases, users may wish to use the output trigger to synchronize external hardware with video acquisitions. We recommend that users configure their MCAM with the following settings so that a single pulse is generated every time a video acquisition is started.



For video acquisitions completed with the **Animal Tracking** panel (detailed below), we recommend that users retain maximum flexibility of the output trigger by completely disabling the output triggers generated by the sensor hardware

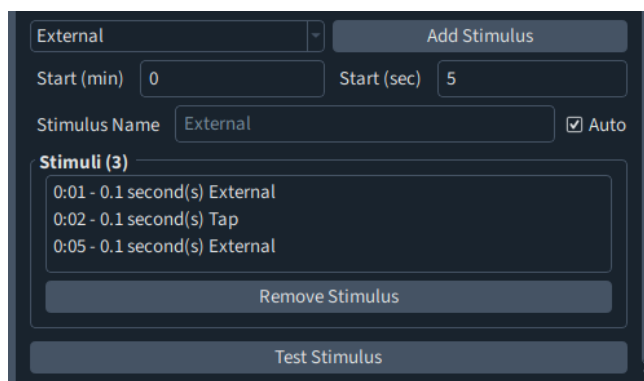


Output Trigger for usage with Animal Tracking

The **Animal Tracking** panel in the MCAM GUI provides an additional interface to generate output trigger pulses with precise control during the video acquisition timeline. This mode is particularly useful for behavior experiments requiring synchronization between the Kestrel and external actuators and is available to Kestrel systems equipped with the external trigger option and software version 0.19.258 or above. These pulses will be generated in addition to any other triggers enabled in the GUI, for example at the start of video acquisition, or the start of data received. It may be desirable to disable all other events for the output trigger.

Users can specify one or more pulse times (in seconds) relative to the start of video acquisition, and the MCAM system will emit pulses at the desired times, independent of the video capture settings. To configure:

1. Navigate to the Animal Tracking panel.
2. Select your desired assay settings.
3. Navigate to the stimulus section and select "External"
4. For each desired pulse, add the time corresponding to the rising edge.



These pulses will be automatically emitted during acquisition and also recorded in the exported CSV data and behavior plots and can be intermixed with other stimuli. The pulses generated by the external stimuli are always **100 ms** long. The hardware receiving the pulse may decide to trigger on the rising edge of the pulse to trigger at the start of the programmed event. It is possible to start the stimuli at time "0 s" allowing the external hardware to receive one pulse at the start of the video recording.

Known issues

The output trigger may at times emit more pulses than the number of frames requested when the MCAM is used in "software trigger mode" (default).

Input trigger

The use of the external input trigger capabilities are currently quite limited. Please contact Ramona Optics to discuss your use case for more information.

Appendix E: Sensor Optical Characteristics

Sensor Core Characteristics - Single Sensor

PARAMETER	SYMBOL	UNIT	VALUE
Pixel Size	Ps	um	1.1
Vertical Pixels	Vp	#	3120
Horizontal Pixels	Hp	#	4096
Analog Gain	Ag	Max	2x Typical 7.75 Absolute Maximum
Digital Gain	Dg	Max	4x Typical 7.98x Absolute Maximum
Chief Ray Angle	CRA	degree	11
Frame Rate (Bin 1)	Fr1	fps	15 (3072×3072)



Frame Rate (Bin 2)	Fr2	fps	65 (3072×3072)
Frame Rate (Bin 4)	Fr4	fps	120 (3072×3072)

High-speed Fruit Fly Configuration

SENSOR CHARACTERISTICS - ARRAY

PARAMETER	SYMBOL	UNIT	VALUE
Pixel Size	Ps	um	1.1
Vertical Pixels	Vp	#	2048
Horizontal Pixels	Hp	#	2048
Analog Gain	Ag	Max	7.75x
Digital Gain	Dg	Max	7.98x
Chief Ray Angle	CRA	degree	11
Frame Rate (max)	Fr	fps	45
Data Rate (single sensor max) @ Fr	D1	MB/s	188
Max Data Rate (sensor array max) @ Fr	Dm	MB/s	6100
Recording Duration per 100 GB @ Dm	Tr	sec	17.7 sec

Sensor Characterization Parameters

Parameter	Symbol	Unit	Min	Typ	Max	Test Conditions	Gain	Integration Time (ms)	Temp.
Green Response	S_G	LSB	449	507	545	Test Setup 1	G_{min}	16.7	60°C
Relative Response	R_R	ratio	0.57	0.60	0.72	Test Setup 1	G_{min}	16.7	60°C
	R_B	ratio	0.43	0.44	0.54	Test Setup 1	G_{min}	16.7	60°C
Readout Noise	σ_{l_2a}	LSB	2.2	2.3	2.7	Test Setup 2	G_{max}	1	0°C
	σ_{l_2b}	LSB	0.56	0.60	0.65	Test Setup 2	G_{min}	1	0°C
Dark Current	I_{dk}	LSB/s	4.6	5.9	7.9	Test Setup 4	G_{max}	125–500	60°C
Photo response Non-Uniformity	PRNU	%	0.78	0.82	2.3	Test Setup 1	G_{min}	16.7	60°C
Dark Signal Non-Uniformity	DSNU	LSB	1.2	1.4	1.7	Test Setup 3	G_{max}	67	60°C
Low-Light Fixed-Pattern Noise	LLFPN	%	1.10	1.14	2.19	Test Setup 5	G_{max}	33	0°C
SNR at Mid-Level	SNR_{MIDL}	dB	30.5	33.7	34.0	Test Setup 1	G_{min}	16	60°C
Max Signal-to-Noise Ratio	SNR_{max}	dB	35.1	36.7	36.7	NA (calculated)	NA	NA	60°C
Dynamic Range	DR	dB	69.1	70.3	70.9	NA (calculated)	NA	NA	60°C

NOTE: Min, Typ, and Max refer to 0.5%, 50%, and 99.5% percentile points in the sample distribution, respectively.

Sensor ADC Bit depth: 10 bit

MCAM Data Transmission: 8 bit

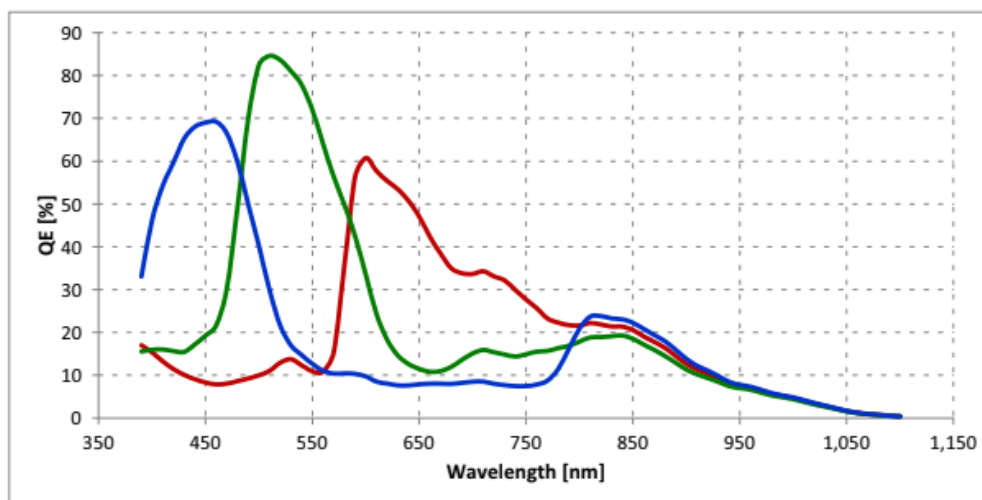
The 8 bits returned to the computer are the 8 MSBs of the sensor's digitized 10 bit val.

Example:

Sensor pixel value 301 (Maximum value 1023).

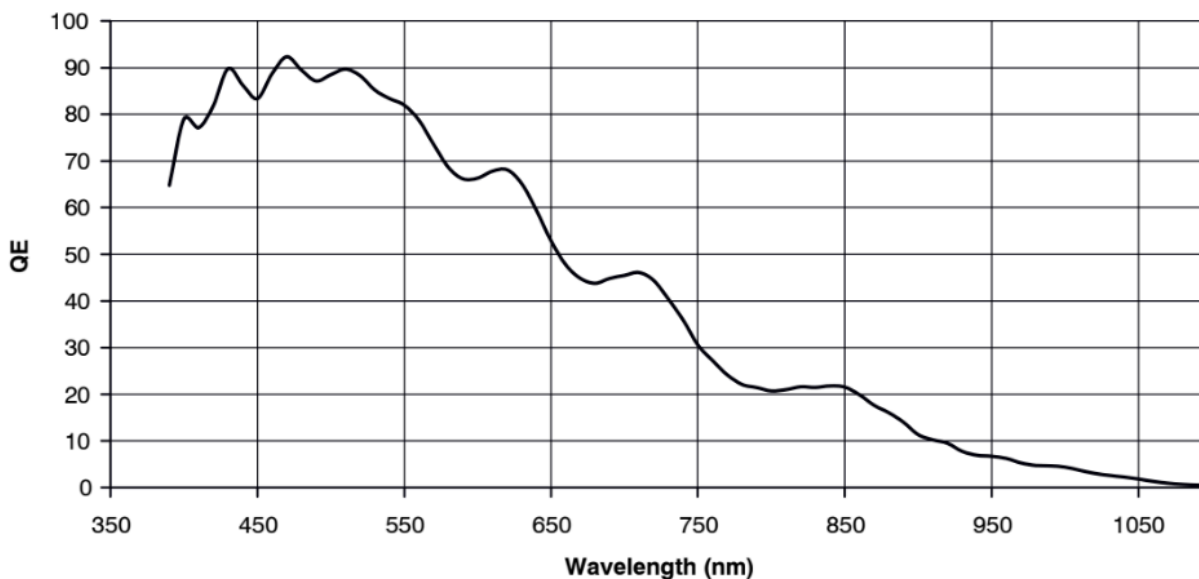
MCAM transmission value to computer 75 (Maximum value 255).

Sensor Spectral Characteristics



NOTE: QE is for Bare Die version.

RGB (Bayer) CMOS: Nominal quantum efficiency of each color pixel in the sensor array



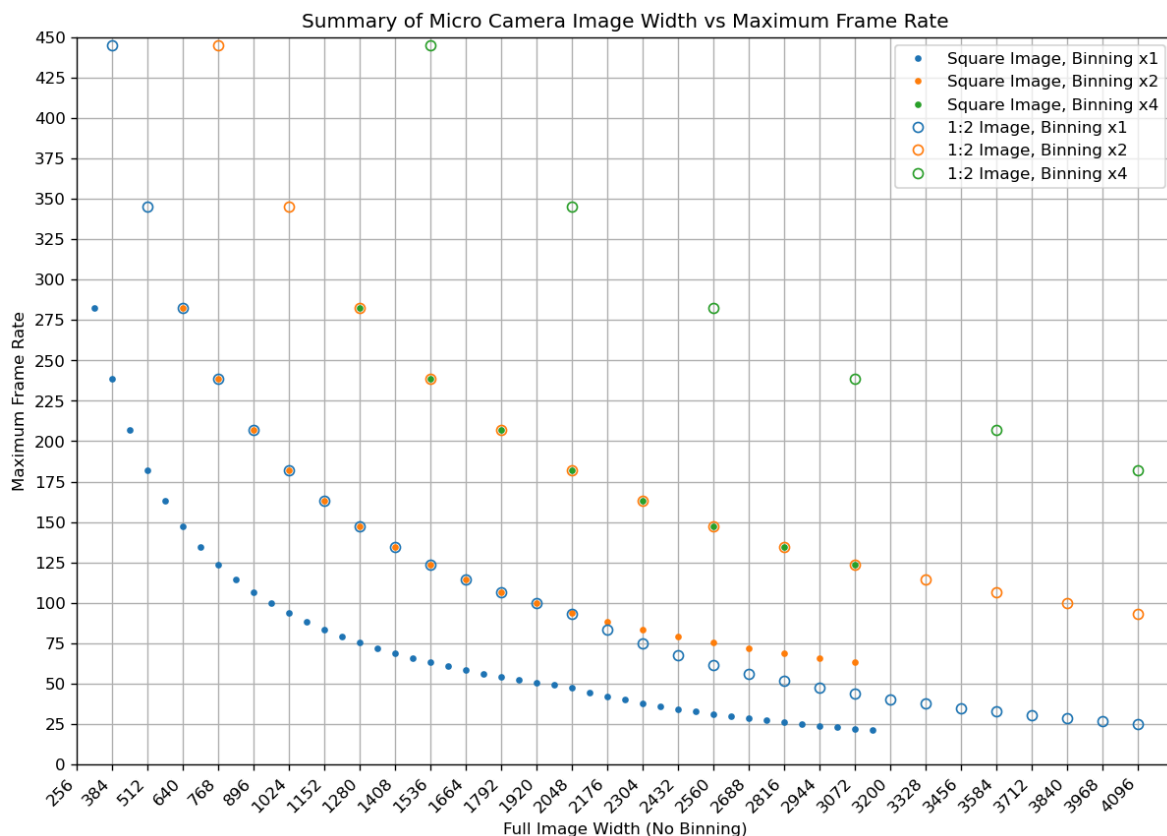
Monochrome CMOS: Nominal quantum efficiency of each pixel in the sensor array

Appendix F: Frame rate control and limitations

We encourage users to benchmark the frame rate of their particular acquisition based on their final imaging parameters. The number of cameras enabled, exposure time, and region of interest within the sensor will all affect the maximum frame rate.

We attempt to list a few different factors that limit the frame rate of the system.

- 1 Each sensor is limited to a bandwidth of 195 Megapixels per second. This limitation is summarized in the figure below for square images and for images with a height to width ratio of 1:2.
- 2 The total bandwidth from the frame grabber to the computer is limited to approximately 6 GB/s but depends on the exact hardware configuration of the connected computer.



The maximum frame rate when the image settings are set to square images and to a height to width ratio of 1:2. The x-axis corresponds to the unbinned image width. Note that not all image sensor shapes are allowed in all bin modes.

Appendix G: Thermal Monitoring

Temperature information is included in the MetaData for all data acquisitions as outlined below.

OPERATING MODE	REPORTING PERIOD	TEMPERATURE A	TEMPERATURE B
Image	1	Ambient	Image Chamber
Video	1 per second	Ambient	Image Chamber
Timelapse	1 per minute	Ambient	Image Chamber

The MCAM is equipped with (2) two thermistors that are used to report Temperature A and Temperature B as defined above. One (1) thermistor is positioned on the exterior of the MCAM while the other is positioned inside the imaging chamber to provide temperature readouts near the sample.

Image Chamber temperature and the temperature of the water in a 96-well plate inside the Image Chamber over time at different IR settings (given an ambient temperature of 21-24 C) is characterized within the tables below.

Well Plate Water Temperature Table

	TIME = 0MIN	TIME = 5 MIN	TIME = 30 MIN	TIME = 60 MIN	TIME = 90 MIN
0% IR	27	26	25	24	24
25% IR	27	27	26	25	25
50% IR	27	27	25	24	24
75% IR	27	26	25	25	25
100% IR	27	26	26	26	26

Kestrel Chamber Temperature Table

	TIME = 0 MIN	TIME = 5 MIN	TIME = 30 MIN	TIME = 60 MIN	TIME = 90 MIN
0% IR	24.9	25.01	25.09	25.23	25.35
25% IR	25.46	25.66	25.89	25.93	25.76
50% IR	23.96	24.23	25.09	25.51	25.62
75% IR	25	25.4	26.37	26.6	26.48
100% IR	25.15	25.53	26.76	27.25	27.52



Appendix H: Environmental Control

The MCAM can integrate with third-party stage-top type environmental systems units when active environmental control is required.

When working with samples that require sustained elevated temperatures, or when granular control of temperature and/or environmental gas concentrations is needed, we encourage the user to work with knowledgeable vendors of environmental control systems. Such systems can be easily integrated with the Ramona Optics Kestrel in both the Enclosed and Open operating condition as shown below.

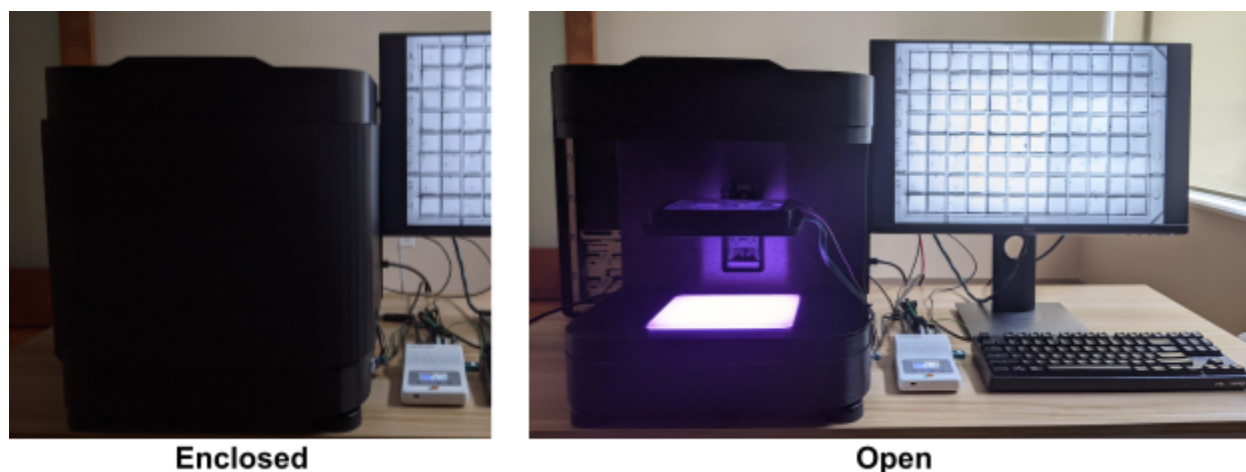


Fig 122: Active control of the environment using a stage top incubator in the Kestrel.

The anodized aluminum drop-in type stage provided by Ramona Optics accepts most active environmental chambers from third-party vendors that are designed to integrate with a 160 mm x 100 mm opening. Ramona Optics has verified adequate fit and cable routing in the Kestrel for the stage-top type environmental chambers provided by Tokai Hit as shown below.



Fig 123: Close up view of the elements in the active control module.

Note to User: ensure the stage-top environmental control unit engages the integrated spring clips in the Ramona Optics drop-in type stage. Engaging the clips ensures level seating on the stage and snug-fit during operation.

For additional information on pricing and availability of systems from Tokai Hit, please contact:



Tokai Hit

306-1, Gendoji-cho, Fujinomiya-shi,

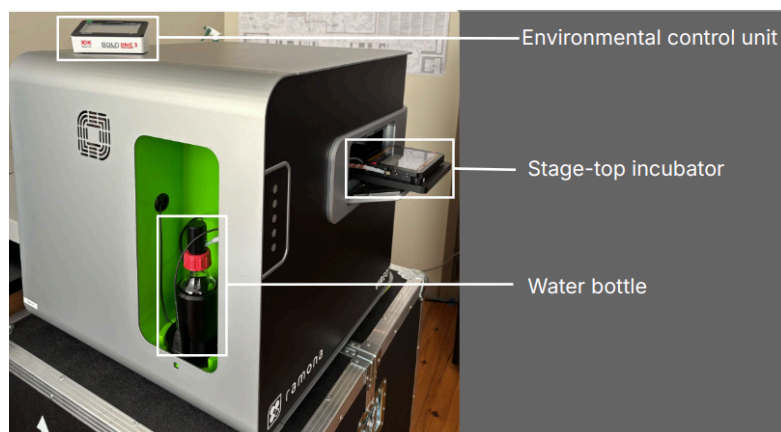
Shizuoka-ken, Japan 418-0074

Phone: +81 544 24 6699 FAX: +81 544 24 6641

Email: solutions@tokaihit.com

Reference: Ramona Optics, MCAM - drop-in type stage integration

For the Vireo, we have integrated the Okolab stage-top incubator (H303-T-H-CO2) within the Vireo nest where users can control humidity (relative humidity 50% - 95%), temperature (3°C above ambient temperature to 60°C), CO₂ (0% - 20%) and optionally, O₂. Incubation can be integrated in your initial purchase or retrofitted into your Vireo unit (though this is a more complex process). Please contact Ramona Optics directly for more information.



Appendix I: Pixel Unit Conversion between GUI and Python

The MCAM reports the acquired region of interest within each sensor in the dataset’s start_pixel and end_pixel properties. The start_pixel is reflected in the dataset’s x and y coordinates.

The MCAM method `select_pixels` assumes the coordinates are in the reference frame of the sensor.

Unfortunately, when exporting the images in a standard image format such as a biap, an offset exists between the pixel location reported by the image in an external image editor, and the pixel coordinates reported by sensors. This offset is reflected in the metadata.json, but may be confusing to the user.

To assist with these workflows, Ramona Optics has enabled a special acquisition mode for the full image from the sensor. This acquisition mode must be enabled in the GUI (Version 0.6.14 and above) before the MCAM is open. To do so, follow the instructions below:

STEP	ACTION
1	Open the MCAM GUI.
2	Open the “Additional Settings” under the “Advanced” menu (Fig 124).
3	Select the acquisition mode that is labeled 3136 x 4224 pixels.

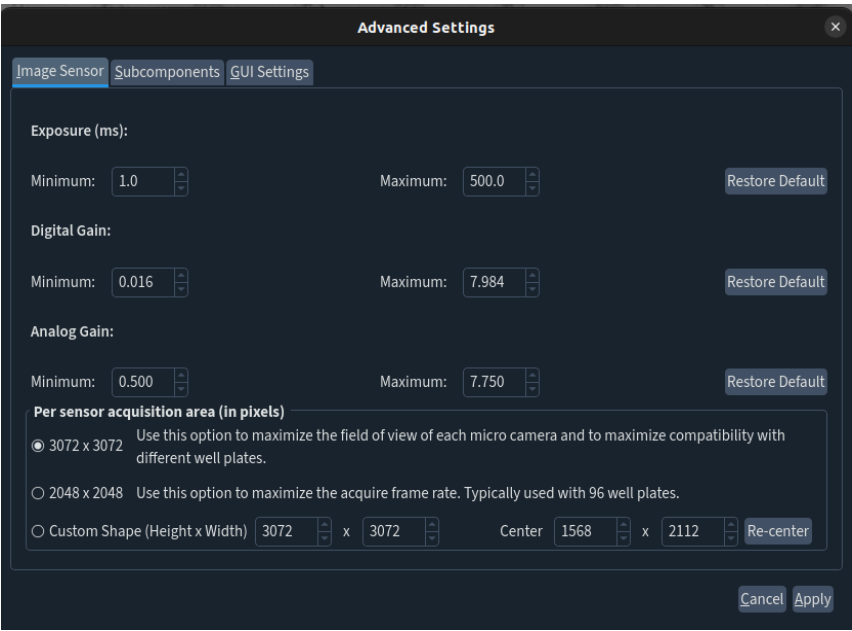


Fig 124: Advanced settings panel.



4	Apply the settings.
---	---------------------

5	Open the MCAM.
---	----------------

With these settings, the MCAM will be acquiring all pixels from the imaging sensor starting at the (0, 0) coordinate.

This mode does not support high frame rate mode.

Acquiring an image and exporting it will ensure that the coordinates of the exported BMPs are the same as those of the imaging sensor.

Appendix J: Optical Specs of Objective Lenses: high-speed Fruit Fly

OBJECTIVE CHARACTERISTICS - ARRAY

PARAMETER	SYMBOL	UNIT	VALUE
Numerical Aperture	NA	-	~0.02
Working Distance	Wd	mm	260-270
Depth of Field	DOF	um	~2000
Objective Pitch	p	mm	13.5
Objective Field-of-View	FOV	mm x mm	15 x 15



Appendix K: MCAM Configuration

The MCAM can be configured with different illumination modes and stages for optimal image acquisition and analysis. To ensure smooth operation of the MCAM after it is started, we enforce that all subcomponents are connected at startup.

The current options for subcomponents include:

- Reflected High-power Excitation Stimulus Module
- The Transmission Illumination module.
- The Fluorescence Illumination module.
- A stage that moves along the z direction (moves along the optical axis).
- A stage that moves along the x direction (In the plane of the sample) - optional.
- A stage that moves along the y direction (In the plane of the sample) - optional.
- A stage that moves along the x axis, but only for the sample.
- A Temperature monitoring module.
- A Plate Tapper module.

The page also allows for configuration of the orientation of the microscope.

- Exif Orientation 8 should be used for:
 - Upright Microscope
 - Well A1 in the top left corner
- Exif Orientation 6 should be used for:
 - Upright microscope
 - Well A1 in the bottom right corner
- Exif Orientation 7 should be used for:
 - Inverted Microscope
 - Well A1 in the top left corner
- Exif Orientation 5 should be used for:
 - Inverted Microscope
 - Well A1 in the bottom right corner



Appendix L: Transmission Illumination Module

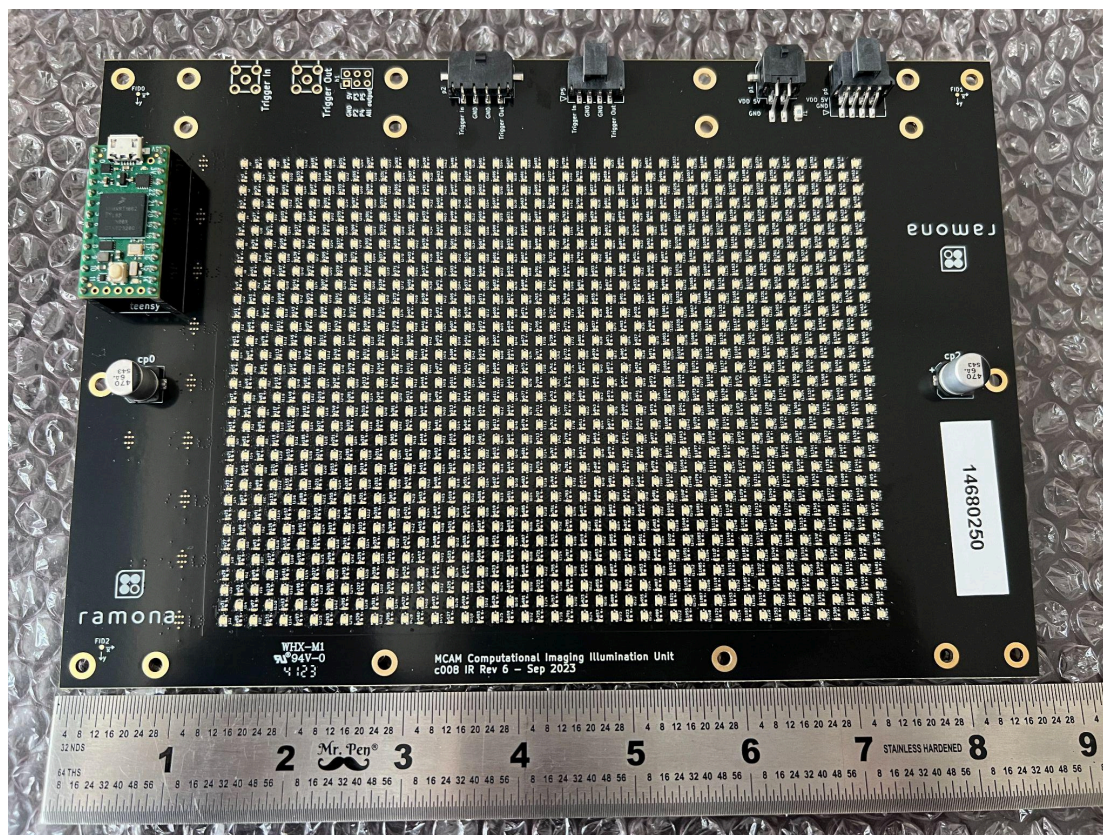


Fig 125: Transmission Illumination Module

33 rows

45 columns

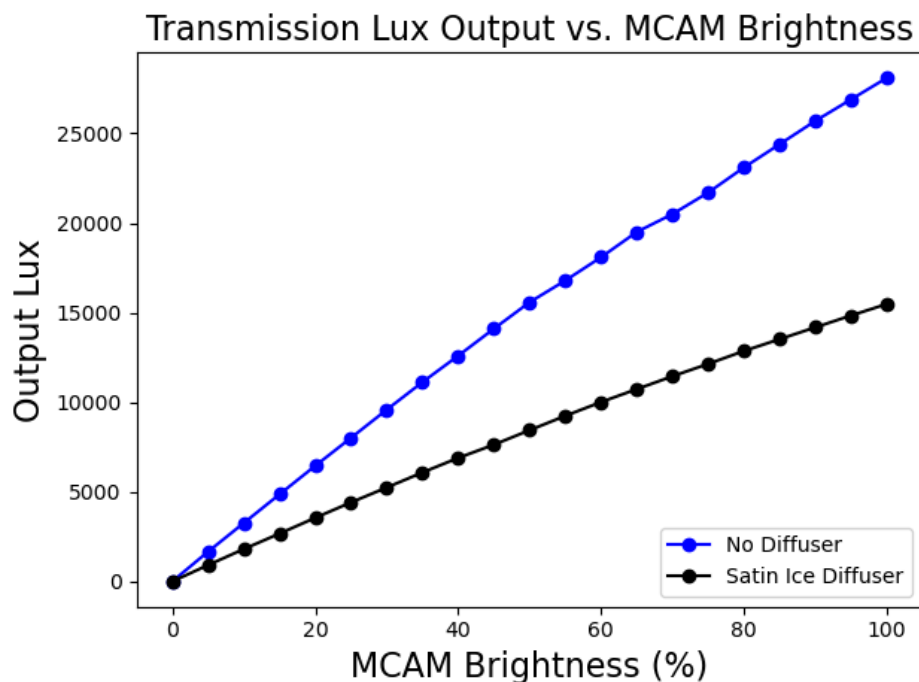
Total = $33 \times 45 = 1485$

The spacing between each LED in both X & Y is 3.375mm

IR and RGB checkerboard pattern

RGB = 743 Total LEDs emitting at 630 nm (R), 525 nm (G), 470 nm (B)

IR-A: 742 Total LEDs emitting at 850 nm



Note: The above lux output per brightness assumes that only transmission RGB lighting is activated. Using another light source at the same time may reduce the power output available to the transmission lighting board and thus decrease the expected output lux.

WARNING: IR illumination causes a reduction in pixel resolution by about 5%.



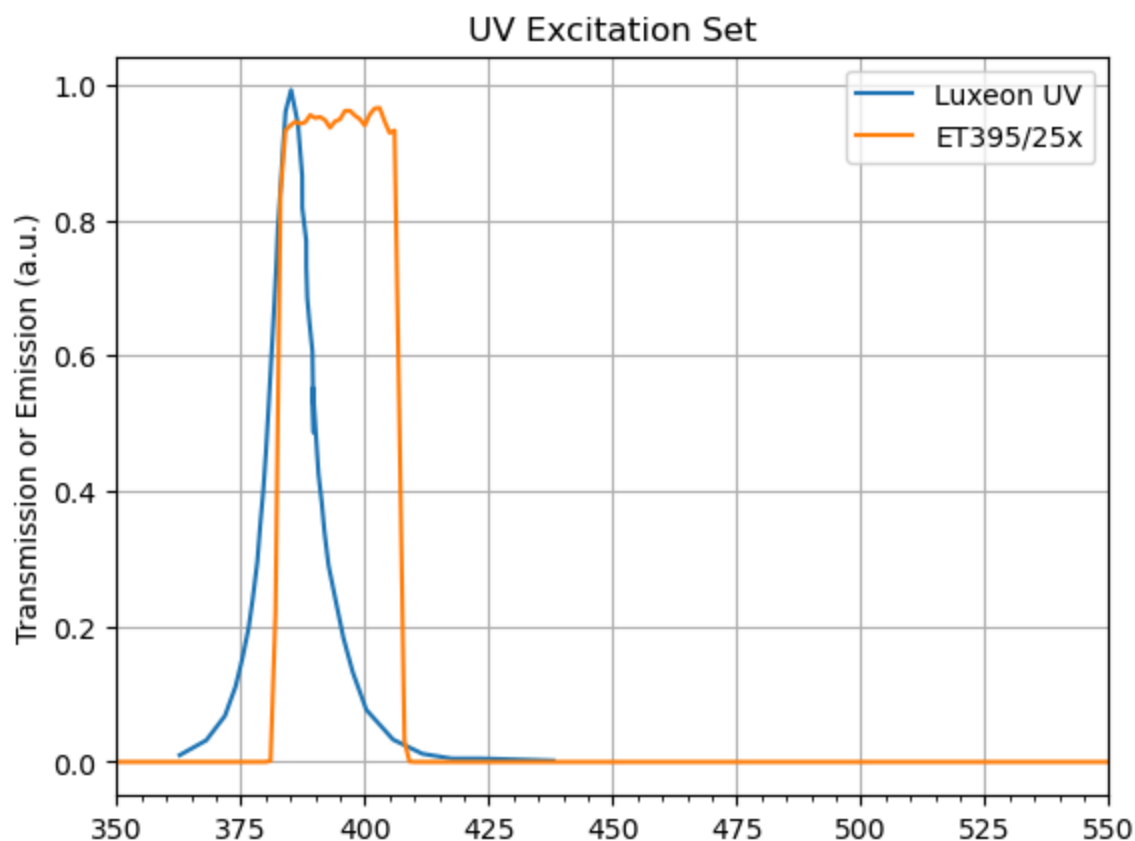
Appendix M: Fluorescence Illumination Spectra

Kestrel

The Kestrel may be equipped with either a single channel fluorescence or multi-channel fluorescence controller. The following plots show the expected spectra of the LEDs emission before any excitation filter.

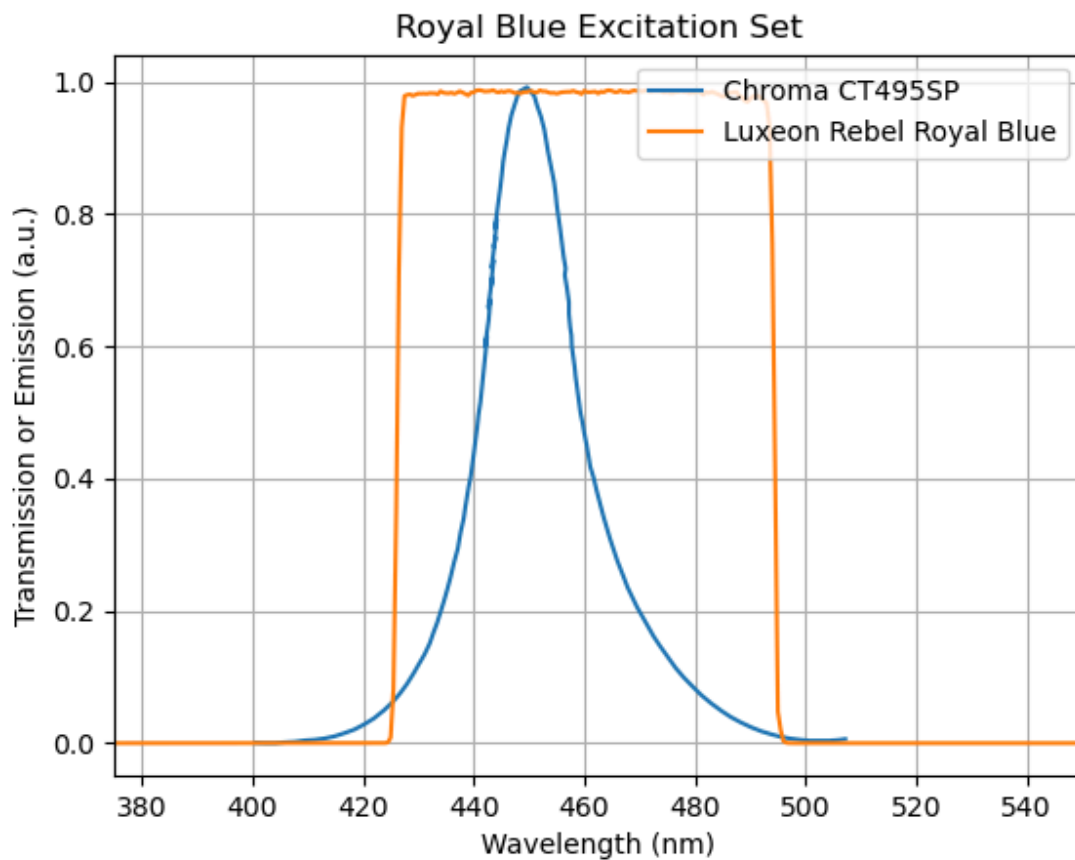
UV LED - 385 nm Excitation

Luxeon L1F3-U380200008000. For more information about the LEDs used please refer to their specific datasheet found online <https://lumileds.com/products/uv-leds/luxeon-uv-u-line/>.



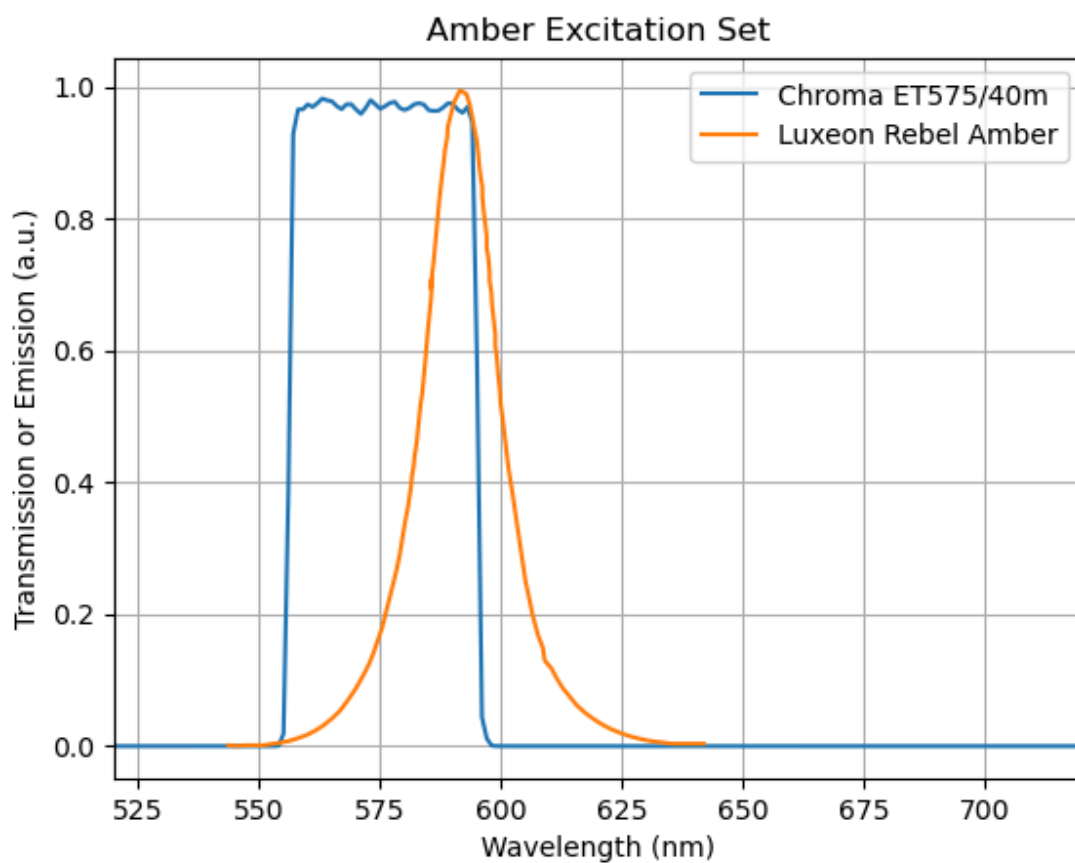


ROYAL BLUE - 440 nm Excitation





AMBER - 590 nm Excitation

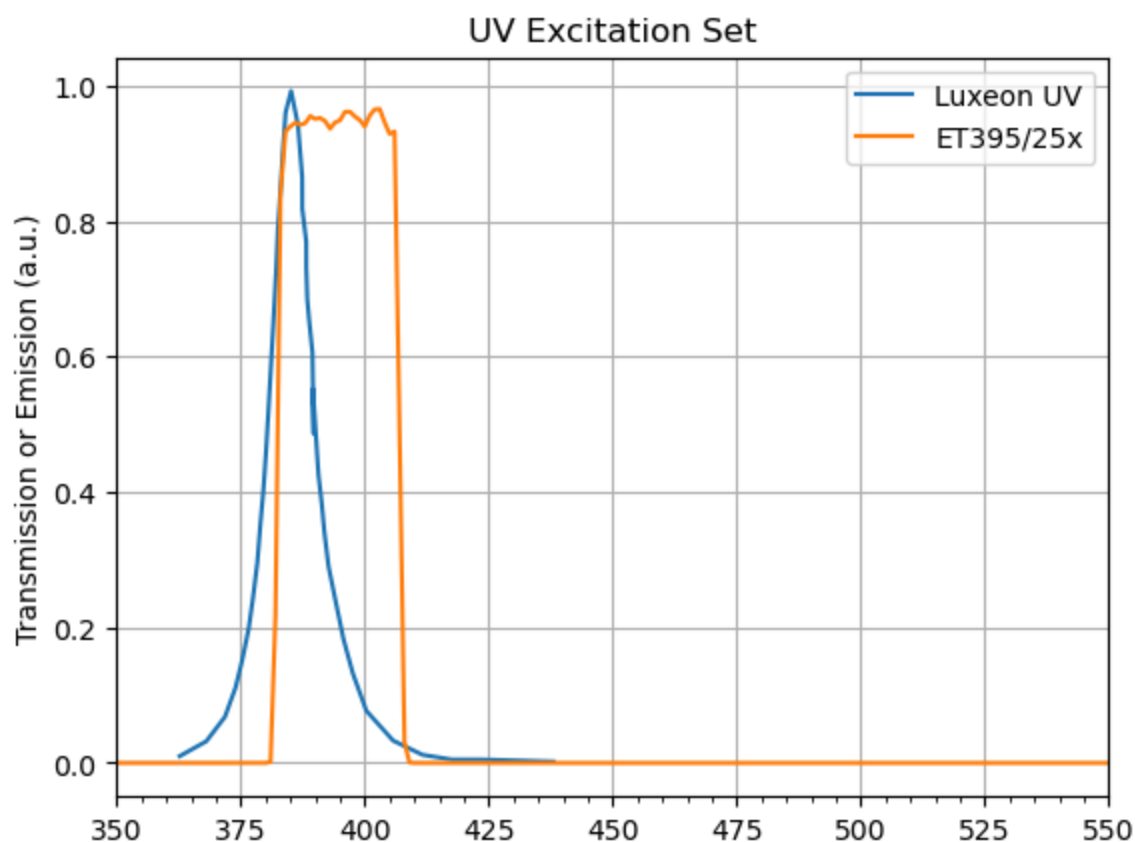




Vireo

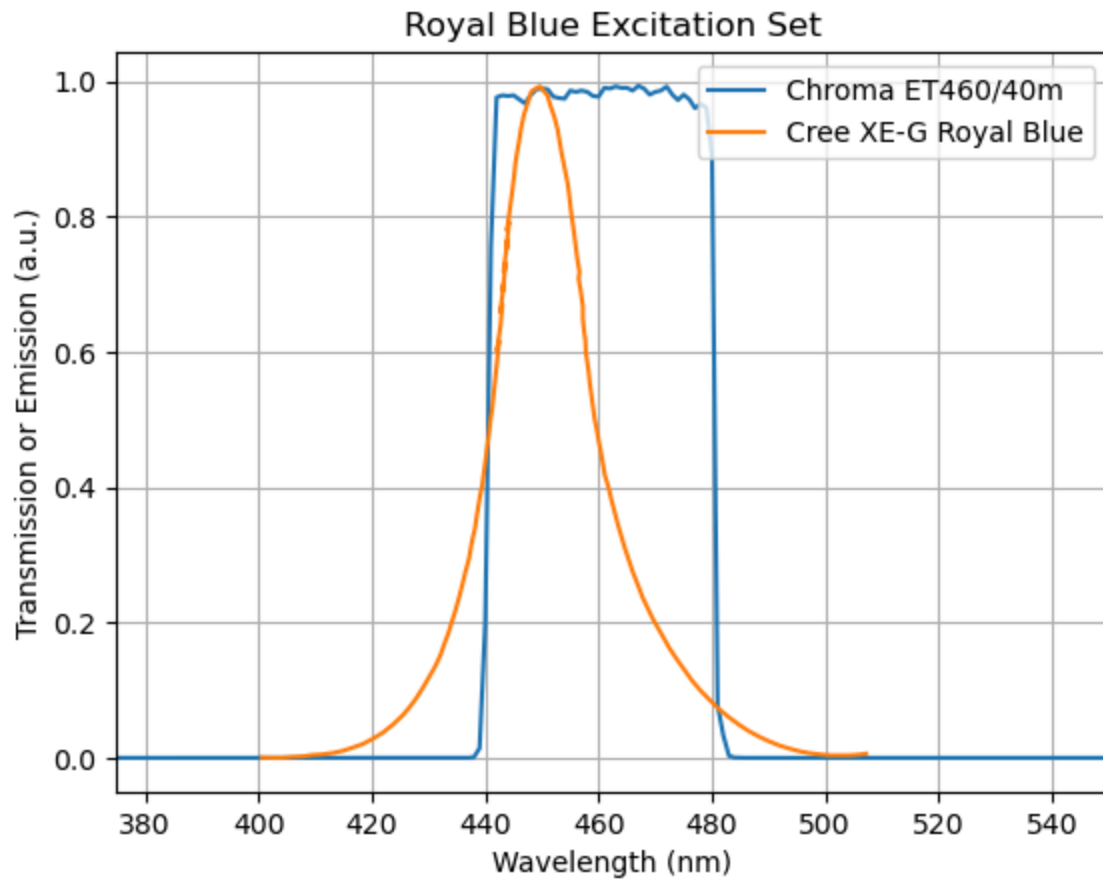
The Vireo may be equipped with either a single channel fluorescence or multi-channel fluorescence controller. The following plots show the expected spectra of the LEDs emission before any excitation filter.

UV LED - 385 nm Excitation



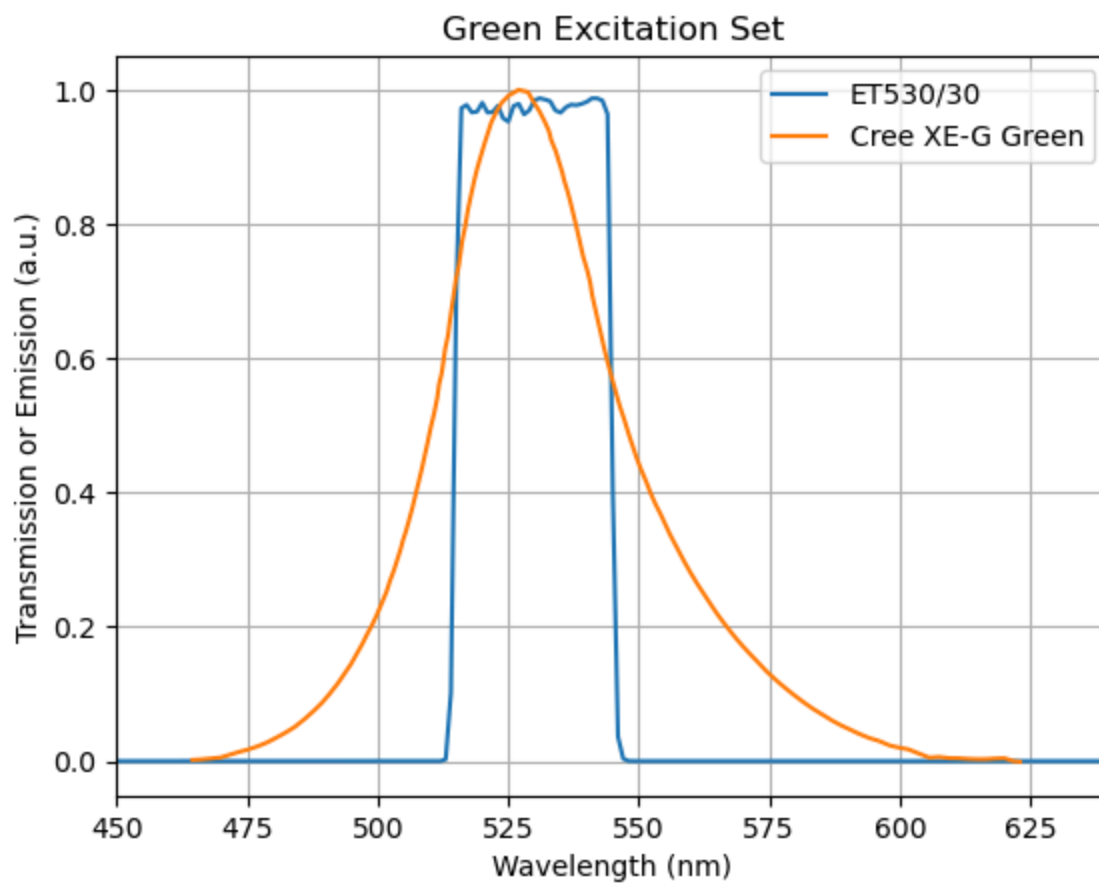


ROYAL BLUE - 440 nm Excitation



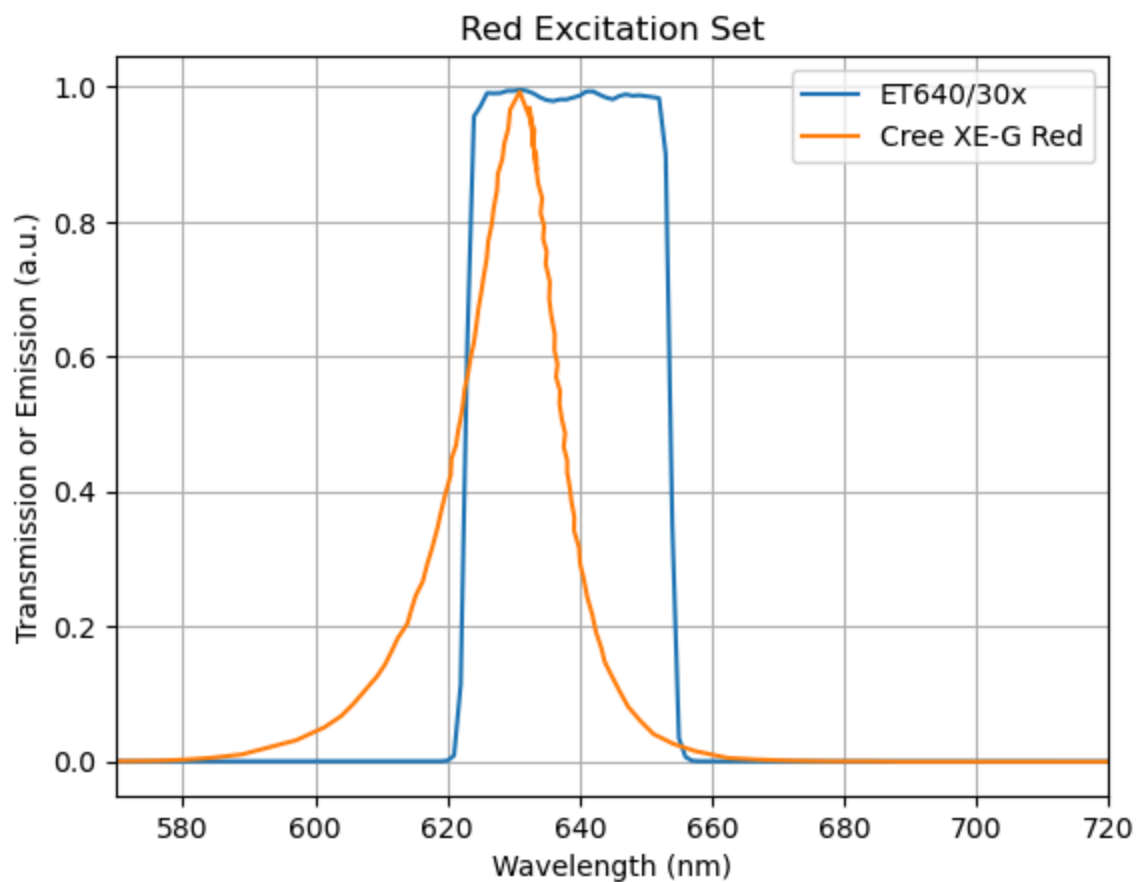


Green - 530 nm Excitation





Red - 633 nm Excitation





Appendix N: Imaging Orientation

The MCAM GUI orients the images acquired by the MCAM in a way that resembles how the user inserts the sample in the sample holder. This orientation should be set at delivery time, but is configurable by the end user in the Advanced Menu. To access this menu, from the GUI, navigate to Advanced → Additional Settings → Subcomponents. To modify the EXIF unselect the “Default Subcomponents” option which will open a new panel where this value can be accessed and changed. The orientation can be set to one of 8 orientations. We suggest:

1. Kestrel, upright: EXIF 8
2. Kestrel, inverted: EXIF 7
3. Vireo: EXIF 7

The advanced menu window allows one to change the display orientation of the MCAM. The orientation can be changed by selecting the desired number for the Exif Orientation

With the correct settings, when a standard well plate is placed under the MCAM with the A1 well in the top left corner, the well should appear in the same orientation in the GUI.

No matter the chosen orientation, the datasets remain in an order that is native to the underlying hardware. This corresponds to the image that would appear if Exif Orientation 1 is chosen. This means that, without any kind of rotation, images displayed by other analysis programs may appear in the wrong orientation.

To index the correct cameras, one can use the maps included below.



For EXIF Orientation 8 (Upright Kestrel)



Fig 126: EXIF orientation 8 example.

To use these indices in code, for example, if we want the image from wells C3, C4, D3, D4 from the Exif Orientation 8 dataset, we choose the micro camera index (1, 2) as shown in the figure above (Fig 126).

```
>>> from owl import mcam_data
>>> from matplotlib import pyplot as plt
>>> dataset = mcam_data.load('/path/to/dataset.nc')
>>> image = dataset.image[1, 2].data
>>> plt.imshow(image)
```



For EXIF Orientation 7 (Inverted Kestrel)



Fig 127: EXIF orientation 7 example.

To use these indices in code, for example, if we want the image from wells C3, C4, D3, D4 from the Exif Orientation 7 dataset, we choose the micro camera index (4, 2) as shown in the figure above (Fig 127).

```
>>> from owl import mcam_data
>>> from matplotlib import pyplot as plt
>>> dataset = mcam_data.load('/path/to/dataset.nc')
>>> image = dataset.image[4, 2].data
>>> plt.imshow(image)
```




Other Questions

If you have any other questions regarding the MCAM, please contact help@ramonaoptics.com.

Troubleshooting Guide

Did Not Detect MCAM

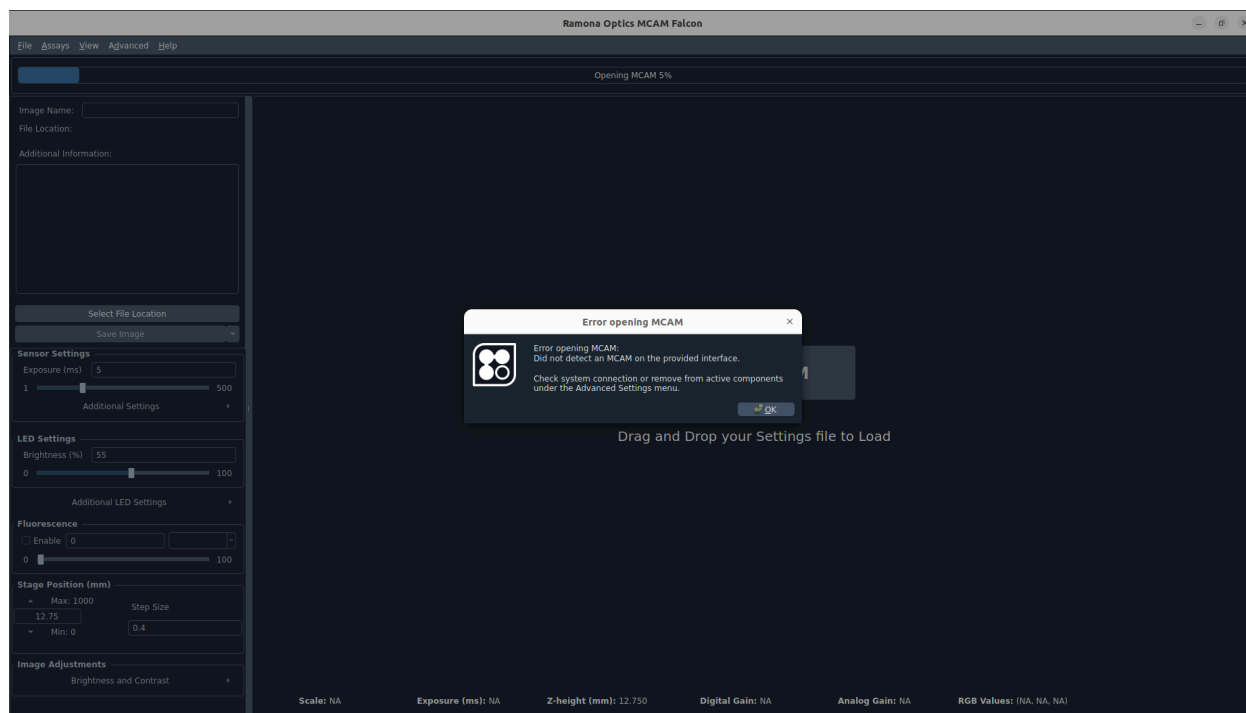


Fig 128: Example "Error opening MCAM" pop up.

This is very likely due to the MCAM being powered off. Please verify the following

- The MCAM is connected to AC power.
- Verify that the power switch is in the on position.
- If both the above are true, the MCAM indicator light should be active.

Once everything is verified to be connected, try connecting to the MCAM again.

Did not detect a subcomponent

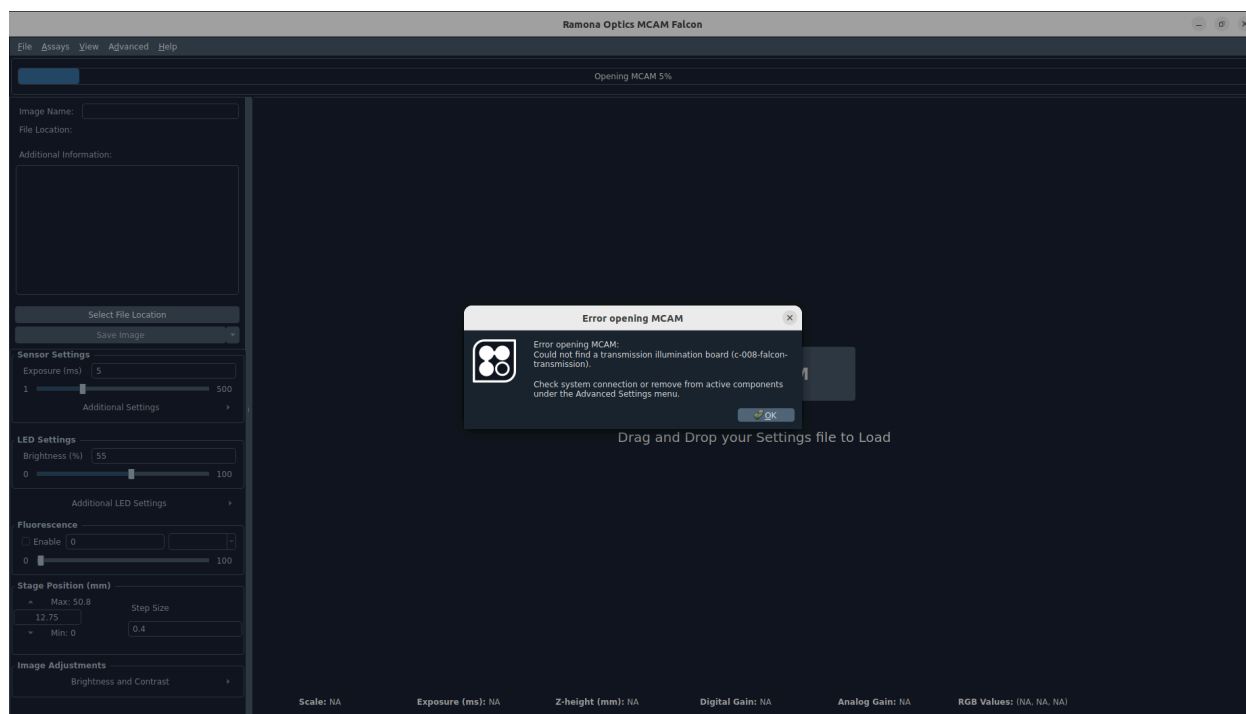


Fig 129: Example "Error opening MCAM" pop up.

Error messages like the one above are typical when the connection is loose between the USB cable connecting the computer and the MCAM (Fig 129). Ensure that the USB cable is connected to both the provided desktop computer, and the MCAM.

If the error persists, verify that the options in the Menu: Advanced ▾ Additional Settings ▾ Subcomponents. Verify that the customization settings match what the Ramona Team has communicated with you at the time of delivery.

Limited Recording Duration

Symptoms: The video acquisition GUI does not allow you to record for the desired duration (Fig 130).

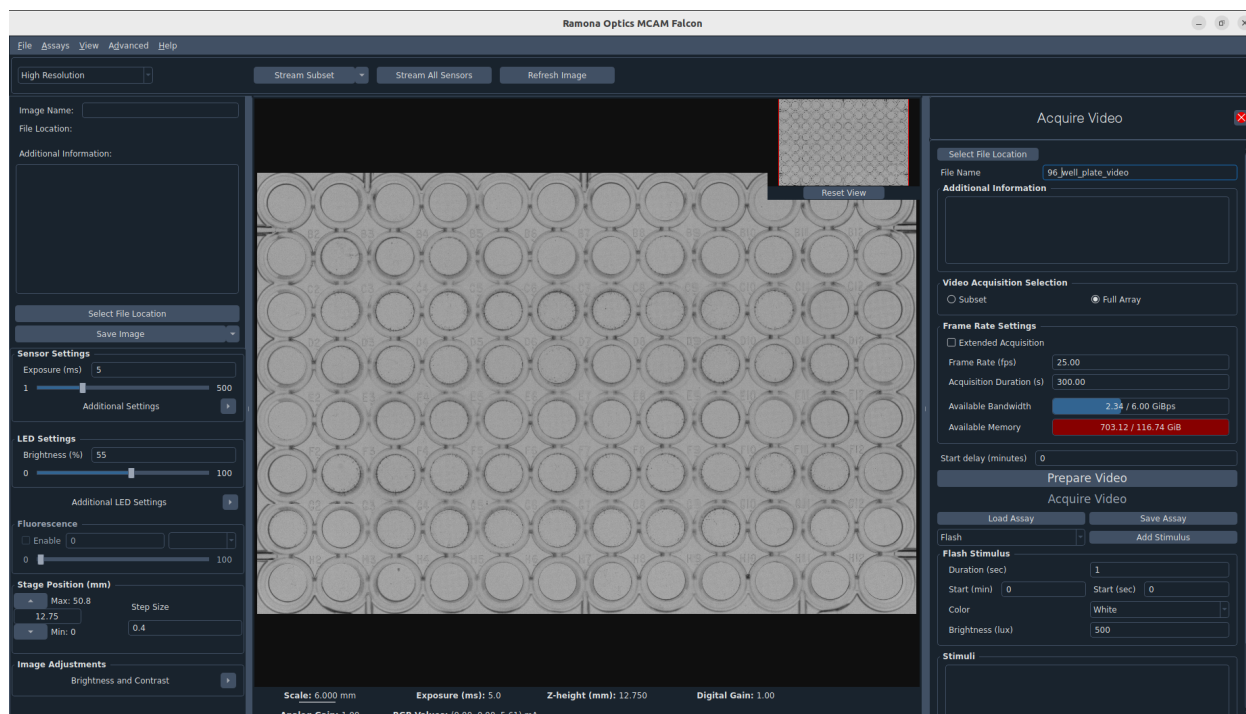


Fig 130: GUI recording time limit exceeded due to high resolution.

The possible causes for this are:

1. The resolution of the system is too large for the desired recording.
 - a. Try changing the resolution of the system on the upper right hand side.
 - i. Selecting "High Frame Rate" will enable the longest recordings.
 - ii. Selecting "Standard (bin x2)" will enable a good compromise between the highest resolution spatial settings and the highest achievable frame rate.
 - iii. Selecting "High Resolution" maximizes spatial resolution at the expense of recording duration.
 - b. Note the system always starts in "High Resolution" Mode.
2. The recording duration must be done in "Extended Acquisition" mode (Fig 131).

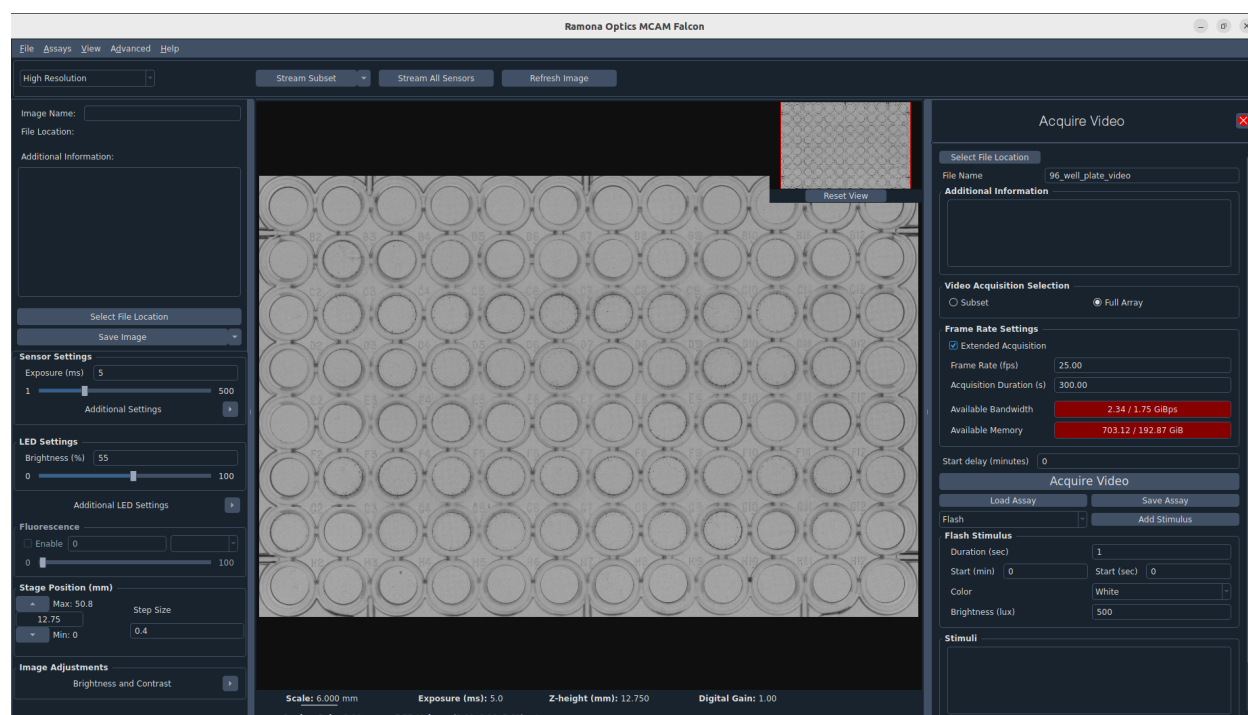


Fig 131: "Extended Acquisition" mode activated on "Acquire Video" panel.

- a. The extended acquisition mode enables recording video directly to the SSD of the MCAM.
- b. The appropriate storage directory for recording
 - i. This is setup by your Ramona team to be /MCAM_data



Cold Boot

The following procedure is one that will force a cold boot to occur. This is the slowest, but most reliable boot up procedure.

1. Ensure that your computer is off.
 - a. To ensure it is really off, power down the computer, and remove the power cable from the computer.
 - b. Keep the cable unplugged for 30 seconds (in all seriousness, please wait).
 - c. Plug the power cable back into the computer.
2. Ensure that the MCAM is off (switch on the right hand side).
3. Turn on the MCAM (not the Computer).
4. Wait 30 seconds.
5. Turn on the Computer.
6. Open the MCAM GUI and open the MCAM.

The connection issues should be resolved.



Connection issues

Common error messages that may appear when connecting to the are divided into three:

1. Connecting to the Multi-Camera Array
2. Connecting to the Illumination Modules
3. Connecting to the motion subsystem



Ensuring a good connection to the Multi-Camera Array

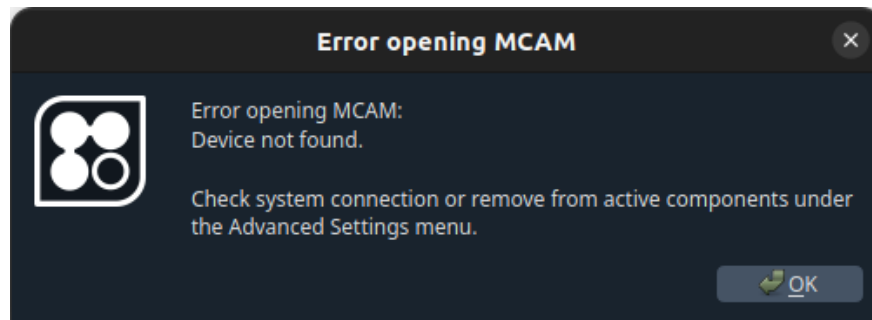


Fig 132: Error pop up indicating connection issues.

This error message typically describes a missing connection from the MCAM body to the computer (Fig 132). To fix this issue, please ensure that the:

1. MCAM is powered on. The indicator light on the side of the MCAM should be illuminated to indicate that the system is powered on.
2. The Power on sequence of the system was followed.

If your issue is still not resolved, please do not hesitate to contact your Ramona Optics representative or contact us by email at help@ramonaoptics.com

Fluorescence Illumination Modules Turned Off or Not Turning On

If the fluorescence LEDs turn off unexpectedly or do not turn on when enabled, it is likely a result of overheating. When driven for long periods of time, and especially at higher power levels, the LEDs heat up quite a bit. For the safety of the user and the hardware, over-temperature protection circuits are in place to automatically turn off the LEDs when they reach 70 °C. To recover from an over-temperature shutdown event:

STEP	ACTION
1	Disable Fluorescence in the GUI by clicking the switch below the Enabled label under the Illumination Settings (Fig 133).

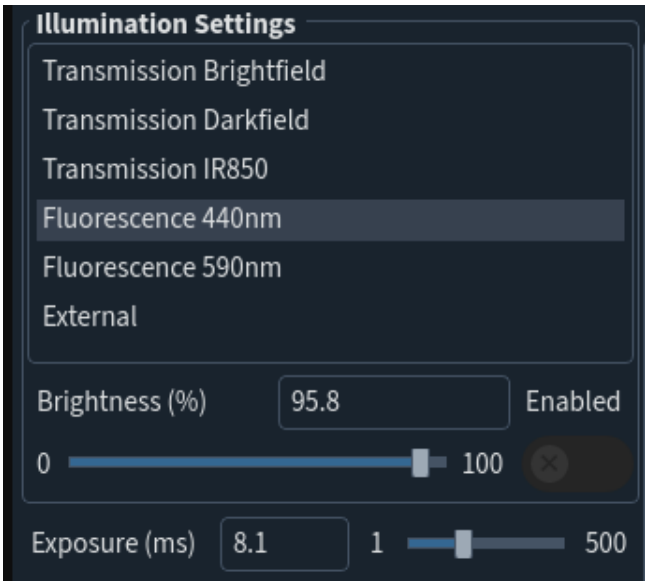


Fig 133: Fluorescent light switch when lights are off.

2	Wait for the LEDs to cool down. The temperature of the fluorescence LEDs should reach room temperature in about 15 minutes.
---	---



- 3 When you are ready to use the fluorescence modules again, re-enable Fluorescence in the GUI by clicking on the switch below the Enabled label again (Fig 134).

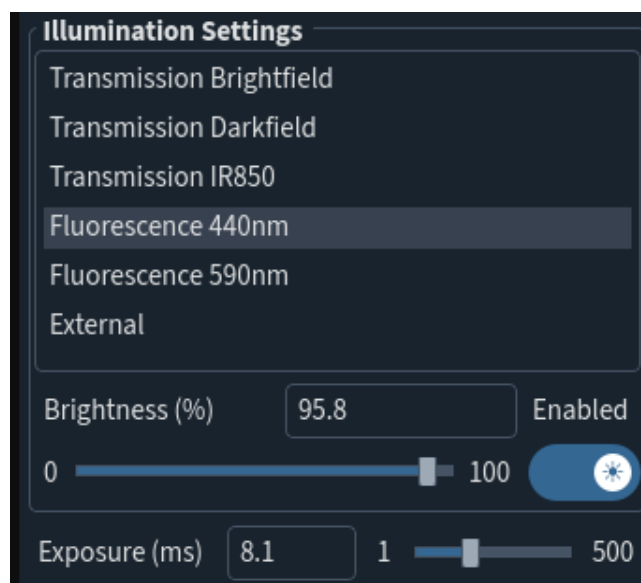


Fig 134: Fluorescent light switch when lights are on.