

MitoMapper User Guide

Quick Start

1. Download **MitoMapper.zip** and the **MitoMapper.ijm** file.
2. The `.zip` file contains the complete data structure and all required subfolders.
3. Drag and drop the **MitoMapper.ijm** file into Fiji (ImageJ) to open it.
4. Click the **Run** button.

A folder selection dialog will appear.

5. Select the unzipped **MitoMapper** folder. You should now see all mask subfolders.

The program will automatically:

- Loop through all images in the main folder
- Load the corresponding annotation masks from the subfolders
- Generate the graphical output
- Save both the output images and the `.txt` results file in the **Results** folder

Image Naming Convention

Your image files must follow this naming format:

`MainImageID_extension`

Where the extension is one of the following:

- `_nucleus`
- `_myo`
- `_mito`
- `_cell`

The only optional mask in the base version of MitoMapper is the **nucleolus** mask. The program is case-sensitive. Ensure that file names exactly match the expected naming convention.

Detailed Information

The MitoMapper macro runs stably in Fiji (ImageJ 1.54p). A Python version is also available upon request. Like other macros, it can simply be dragged and dropped onto the Fiji status bar to open automatically. When executed, the macro prompts you to select a folder containing all required data for the analysis. A `.zip` file with a test image and the complete folder structure is available in the project's GitHub repository to help users understand the required setup. The program analyzes all images in the selected main folder and searches the corresponding subfolders for the necessary masks.

Minimum Requirements

The minimum required binary masks are:

- Cell
- Mitochondria
- Myofilaments
- Nucleus

The corresponding subfolders must be named exactly:

- Cell_Masks
- Mito_Masks
- Myo_Masks
- Nucleus_Masks

If any of these folders are missing, the program cannot proceed. Additionally, a `Results` subfolder is required to store:

- The generated graphical output
- The `.txt` file containing the numerical results

Optional Mask

Support for a nucleolus mask is available. If present, it should be saved in an additional subfolder named:

- Nucleolus_Masks

If no matching nucleolus mask is found for a given image, the program will continue without it.

Mask Naming Scheme

To correctly associate masks with their corresponding images, the masks must be labeled using the main image ID.

Example:

If the original image is:

`Test_Data.tif`

The corresponding cell mask must be named:

`Test_Data_cell.tif`

The same logic applies to all other mask types.

The program uses strict string matching and is case-sensitive.

Performance and Speed Optimization

If all required files are found, the analysis begins. The most time-consuming factor is the number of mitochondria present in each image. In the base version, the processing speed is approximately **1000 mitochondria per hour**. On high-performance machines, the speed can be increased by reducing the waiting time for probe selection. This can be done by modifying the: `wait(10)` command in the script.

- Reduce the value to increase speed.
- Increase the value if you encounter the error: “*no selection present*”, particularly on older computers.

Increasing the wait time allows additional time for data selection and loading.

Mitochondrial Analysis Method

Mitochondrial parameters are calculated by:

1. Determining the center of mass.
2. Measuring properties every 2 degrees around the object.

This approach enables accurate calculation of length and width.

Known Measurement Limitations

The method may produce two non-critical errors that do not crash the macro but require manual review:

1. **Center of mass outside object boundaries**
 - May result in length and width values of 0.
 - Other parameters (e.g., area, shape descriptors) remain correct.
2. **Open mitochondria at image borders**
 - If a mitochondrion is truncated at the image edge, the calculated length may be returned as infinite.
 - In such cases, the associated data should be removed, as the mitochondrion is not fully captured.

Mitochondrial Classification

Mitochondria are categorized into three main groups based on their distance to the nucleus:

- Close
- Intermediate
- Far

An additional annotation, **Intersected**, is assigned if the most direct path to the nucleus is interrupted by a myofilament.

Output Files and Customization

Upon completion, the program saves:

- A comma-separated `.txt` file in the `Results` folder
- A copy of the original image with ROI overlays of all analyzed organelles
- Color-coded mitochondrial categories

Graphical output settings can be modified in the script under the section:

“Generating Graphical Output” (starting at approximately line 621)

Specifically:

- Line width can be adjusted around line 625
- Category colors are defined around lines 632–637

The `.txt` file can be imported into analysis software of your choice (e.g., Excel), using a comma as the delimiter.

If you have any questions, please feel free to contact us.