Supplementary Material

# Detailed steps of the CellProfiler pipelines

The CellProfiler pipeline developed by Reis et al. [[1]](https://sciwheel.com/work/citation?ids=1411272&pre=&suf=&sa=0&dbf=0) includes the following steps. Before the CellProfiler pipeline, the softWoRxTM image analysis software deconvolutes either the middle slice of the z-stack or the z-projections (max intensity). “Identify Prim Automatic” and “Identify Secondary” identify the nuclei and cell locations. Rescale Intensity, Apply Threshold (a global threshold method to define the boundary of mitochondria), Correct Illumination Calculate, Correct Illumination Apply, and Smooth or Enhance (a Gaussian filter with 50 pixels per box) are used for preprocessing. The Identify Prim Automatic module is also used for mitochondrial segmentation and identification with the Otsu adaptive threshold. The Measure Object Area Shape, Measure Object Neighbors, and Relate Object (designed to create links between mitochondria and the cells to which they belong) modules are then applied to extract the features of mitochondrial profiles and relationships among mitochondria, cells, and their neighbors. The features of “Zernike” and “Mitochondrial Area and Shape” are the most relevant features in the random forest classification, where “Zernike” refers to Zernike-polynomial-based matrices [[2]](https://sciwheel.com/work/citation?ids=12196374&pre=&suf=&sa=0&dbf=0) that describe binary information in 2D images.

The CellProfiler pipeline developed by Rees et al. [[3]](https://sciwheel.com/work/citation?ids=11159404&pre=&suf=&sa=0&dbf=0) includes the following steps. In the preprocessing stage, the Correct Illumination Calculate and Correct Illumination modules normalize the illumination over the field of view. After preprocessing, the Identify primary/secondary/tertiary object modules identify the nuclei, cell bodies, and cytoplasm. The Enhance or Suppress Feature module then enhances the mitochondrial speckle signals in the images; the Identify primary objects module recognizes mitochondrial puncta with the “Otsu” thresholding “Per Object.” Afterward, the Measure object size and shape and Measure object intensity modules measure the mitochondrial puncta profiles and intensities. Subsequently, the Relate objects module sorts the linkage relationships between mitochondrial puncta and nuclei.

# Detailed steps of the MitoGraph pipeline

The pipeline of MitoGraph [[4,5]](https://sciwheel.com/work/citation?ids=5206741,4345584&pre=&pre=&suf=&suf=&sa=0,0&dbf=0&dbf=0) includes the following steps. MitoGraph processes high-quality 3D images of a single cell from ImageJ pipelines and extracts mitochondrial surfaces, networks, and node information for further analysis and visualization using ParaView and RStudio. The steps are described in detail below. The ImageJ macro “GenerateFramesMaxProj.ijm” uses the maximum intensity projections as boundary references to mark the regions of interest (ROIs) in single-cell 3D (z-stack) images. The ImageJ macro “CropCells.ijm” is then used to crop the images according to the ROIs, and the cropped images are then subsequently fed into MitoGraph.

Several parameters can be adjusted in MitoGraph, and these include the pixel sizes of the images, the radius of tubules, the scales for tuning noisy images, the threshold types (global vs. adaptive thresholds), and the threshold ratio. The adaptive threshold divides the images into N areas and calculates the local threshold individually rather than the global threshold sharing a single threshold value. After these steps, MitoGraph generates three VTK files for the 3D rendering surface, the skeleton outline, and the information on nodes in the skeleton. These VTK files can subsequently be rendered with ParaView or other visualization tools, such as MayaVi. The results, including the mitochondrial volumes, lengths, total numbers of nodes and edges, average edge lengths, and degree distributions of the networks, are obtained using R scripts from the developer.

# Detailed steps of the MiNA pipeline

The pipeline of MiNA [[6]](https://sciwheel.com/work/citation?ids=3357652&pre=&suf=&sa=0&dbf=0) includes the following steps. Some preprocessing steps are recommended before implementation of the MiNA image processing pipeline, and these include “Unsharp Mask,” “Enhance Local Contrast (CLAHE),” and “Median Filters” to enhance the edges, sharpen the images with unchanged contrast, equalize histograms locally with limited ranges of changes to avoid overamplifying noise, and eliminate salt-and-pepper noise. In the MiNA pipeline, images are converted into binary (black-and-white) images through the default thresholding method (IsoData algorithm of ImageJ) and then transformed into skeletons, a framework representing lines with a one-pixel width, using the default thinning algorithm. [[7]](https://sciwheel.com/work/citation?ids=1100585&pre=&suf=&sa=0&dbf=0) MiNA then categorizes mitochondrial skeletal structures into two types – networks (at least one junction node) and individuals (no junctions) – and produces network parameters such as the mean branch length, number of individuals and networks, and mitochondrial footprint (i.e., total area). Some of the parameters are further evaluated using the R programming language. The mitochondrial morphological changes under different conditions, such as FCCP exposure, hypoxia, and Mfn-2 deletion, were successfully identified for validation of MiNA. During binarization and skeletonization, artifacts may reduce the accuracy, including “merging close branches,” “merging at blurred boundaries,” and “fragmenting dark branches.”

# Detailed steps of the Mitochondria Analyzer pipeline

Mitochondria Analyzer analyzes both particles and skeletons in binary images and skeleton images[[8]](https://sciwheel.com/work/citation?ids=8048077&pre=&suf=&sa=0&dbf=0). 3D images often require a preliminary deconvolution step, but most preprocessing procedures for 2D and 3D images are similar. The preprocessing steps consist of the ImageJ commands Subtract Background to subtract the background pixel intensity with the Rolling Ball Algorithm (smooth the image with well-preserved edges), Sigma Filter Plus to enhance the local contrast, and Gamma Correction to nonlinearly adjust brightness histograms to increase the brightness of signals from faint objects. After preprocessing steps, the images are binarized using the “Weighted Mean (adaptive threshold)” method, which captures the last noise and exhibits the best performance among all the tested thresholding methods, such as Mean, Median, and MidGray. Residual noise is then removed from the binary images by applying the ImageJ commands Despeckle and Remove Outliers before obtaining the skeleton image using the “Skeletonize (2D/3D)” command.

In the particle analysis, parameters related to the mitochondria count and profile – “Mito count,” “Area,” “Perimeter,” “Form factor,” and “Aspect ratio” – are obtained from 2D binary images by applying “Analyze articles.” In contrast, “Mito count,” “Volume,” “Surface area,” and “Sphericity” are obtained from 3D binary images by applying 3D Object Counter and 3D Particle Analyzer from the “MorphoLibJ” package. In the skeleton analysis, information on the mitochondrial network, such as “Branches,” “Branch Junctions,” “Branch length,” and “Mean branch length”, is obtained from 2D/3D skeleton images by applying the Analyze Skeleton (2D/3D) command. The authors also recommend other ImageJ modules to facilitate the analysis, such as DcceonvolutionLab2 for the deconvolution of 3D images and 3D Viewer and Volume Viewer for visualization.

# Detailed steps of the Momito pipeline

Momito [[9]](https://sciwheel.com/work/citation?ids=6660758&pre=&suf=&sa=0&dbf=0) requires binary images of mitochondria in JPEG format as input. Momito provides a convenient and user-friendly threshold tool for autosegmentation. The segmentation algorithm consists of functions such as “Enhance Filter,” “Median Filter,” “Local Threshold,” and “Erode Count” with several adjustable parameters that help achieve proper segmentation without problems such as granularity and poor connectivity. The algorithm then computes the mitochondrial area, count, length, and connectivity by analyzing the binary and skeleton images generated by the Momito algorithm. Mitochondrial elements are divided into three structural types, ends, tubules and junctions, and the numbers of elements/clusters (ECs), as well as the ratios of junctions/ends (J/E), are calculated to quantify the mitochondrial network connectivity, where higher values for ECs and J/E indicate greater connectivity. Momito includes the “Data Consolidation” and “EMD Comparison” functions for users to compile data obtained from the analysis and compare the distributions.

# Detailed steps of the Mitocluster pipeline

In the Mitocluster pipeline [[10]](https://sciwheel.com/work/citation?ids=5413178&pre=&suf=&sa=0&dbf=0), fluorescent images are first converted to binary images by applying the MATLAB routine im2bw. The skeleton representation is then generated from the binary images using the MATLAB routine bimorph. Finally, the MATLAB routine bwlabel segments the skeleton representation into mitochondrial clusters, and the degree of each connecting point is calculated.

By combining image analysis, percolation theory, and computational model simulations based on the mitochondrial network model developed by Sukhorukov et al. [[11]](https://sciwheel.com/work/citation?ids=4966662&pre=&suf=&sa=0&dbf=0), Mitocluster has been used to estimate the mitochondrial fusion and fission rate ratios c1 and c2 under different cellular conditions, including the induction of fission by adding paraquat (pqt) and enhancing fusion by overexpressing Mfn-1. As a method to determine the parameters, the number of total edges (L) was set to 15000, the estimated average number of edges in the network under the control condition, and c1 or c2 was varied while specific parameters were fixed to fit the network properties of the experimental data. Two additional topological quantities were calculated, the normalized mass of the giant cluster (Ng/N) and the average degree of the nodes (<k>), where Ng is the number of nodes in the largest cluster and N is the total number of nodes in the network, are obtained by analyzing the skeletonized mitochondrial network images to associate experimental data with parameters of the computational model.

# Detailed steps of the Mitometer pipeline

Mitometer [[12]](https://sciwheel.com/work/citation?ids=11560010&pre=&suf=&sa=0&dbf=0) takes TIF image stacks with time series as input and uses a diffuse background removal algorithm to enhance the mitochondrial boundary, Gaussian filter convolution to remove high-frequency noise, and Otsu’s threshold to create a binary mask of semantic segmentation. An optimized algorithm for parameter searching is applied to regulate the preprocessing algorithms in various images and achieve the stability of the automatic analysis.

A frame-by-frame global assignment algorithm is used for mitochondria tracking, where 6 (7 in the case of 3D) morphological parameters, such as area, volume, major lengths, and mean intensity, are calculated for each mitochondrion and compared with those of mitochondria in existing tracks to either follow it or assign a new track. In addition to motion tracking, mitochondrial fusion and fission events were also identified by detecting changes in the mitochondrial velocity and volume. Different conditions were validated with images of breast cancer cells, primary breast cells, and various breast cancer cell lines captured under diverse types of microscopes and both in silico simulated and in vitro cellular mitochondrial networks, which indicates the high flexibility of Mitometer. Mitometer extracts morphological features, such as the mitochondrial area, solidity, and perimeter, and mitochondrial dynamics parameters, such as fission/fusion rates, movement speed, and distance between mitochondria.

# Detailed steps of the MitoWave pipeline

At the beginning of the MitoWave [[13]](https://sciwheel.com/work/citation?ids=11415364&pre=&suf=&sa=0&dbf=0)pipeline, time-series cellular images are first aligned by applying the StackReg plugin of Fiji. Afterward, masks of fluorescent images are generated with the functions “Median Filter,” “Enhance Contrast,” and “Auto Local Threshold (Niblack)” for preprocessing and binary processing, and the module “Analyze Particles,” which omits particles with a radius less than 60 pixels, is then applied. These masks are subsequently used to obtain the fluorescence intensity of each cell in the next step. The procedures of the analysis are constructed with Fiji macros and MATLAB codes: image processing such as binary, segmentation, and extracting information (ex. area, intensity) from images and generating differential stacks for oscillation analysis are applied using Fiji macros, and the calculations of intensity oscillation are implemented with MATLAB functions and codes. Wavelet transform is used to implement maximal overlap discrete wavelet transform (MODWT-MRA) and continuous wavelet transforms (CWT) in MATLAB to determine the features of oscillations in time-series images. One of the key features of the MitoWave analysis is that it classifies the patterns of membrane potential oscillations into five distinct types and obtains the transition time points of inner mitochondrial membrane potentials during ischemia, the predominant frequencies of mitochondrial clusters, and the size of mitochondrial clusters to compare these five oscillation patterns.

# Detailed steps of the Mytoe pipeline

In the Mytoe workflow [[14]](https://sciwheel.com/work/citation?ids=2919814&pre=&suf=&sa=0&dbf=0), a median filter and top-hat filter are used to remove the noise and identify the signal of mitochondria from the background to enhance the separation of mitochondria. Afterward, binary masks and their skeleton images are obtained by applying Otsu’s algorithm and a two-iteration thinning procedure, respectively. Endpoints and branch points are then detected from skeleton images, and optical flows are computed between image sequences to implement the motion analysis. In addition to the indicators of motion analysis, morphological parameters such as the cell area, mitochondrial area, and mitochondrial count are also provided in the Mytoe results.

# Detailed steps of the MitoSPT pipeline

MitoSPT [[15]](https://sciwheel.com/work/citation?ids=12014872&pre=&suf=&sa=0&dbf=0) provides a customized ImageJ micro of preprocessing steps for time-lapse image stacks, including convolution, fast Fourier transform (FFT), and filtering. After visual thresholding, a sequence of individual binary images is inputted to the MATLAB program with related image information (image resolution, time frame and minimum lifetime of the objects). The MALAB algorithm (built-in functions *bwconncomp* and *regionprops*) tracks centroid locations of all identified mitochondrial objects as they travel over time. The net distances traveled by mitochondria are recorded and lognormally distributed on a cell-to-cell basis.

References

[1.    Reis Y, Bernardo-Faura M, Richter D, Wolf T, Brors B, Hamacher-Brady A, Eils R, Brady NR. Multi-parametric analysis and modeling of relationships between mitochondrial morphology and apoptosis. *PLoS ONE* (2012) **7**:e28694. doi:10.1371/journal.pone.0028694](https://sciwheel.com/work/bibliography/1411272)

[2.    Bhatia AB, Wolf E. On the circle polynomials of Zernike and related orthogonal sets. *Math Proc Camb Phil Soc* (1954) **50**:40–48. doi:10.1017/S0305004100029066](https://sciwheel.com/work/bibliography/12196374)

[3.    Rees DJ, Roberts L, Carla Carisi M, Morgan AH, Brown MR, Davies JS. Automated quantification of mitochondrial fragmentation in an in vitro parkinson’s disease model. *Curr Protoc Neurosci* (2020) **94**:e105. doi:10.1002/cpns.105](https://sciwheel.com/work/bibliography/11159404)

[4.    Harwig MC, Viana MP, Egner JM, Harwig JJ, Widlansky ME, Rafelski SM, Hill RB. Methods for imaging mammalian mitochondrial morphology: A prospective on MitoGraph. *Anal Biochem* (2018) **552**:81–99. doi:10.1016/j.ab.2018.02.022](https://sciwheel.com/work/bibliography/5206741)

[5.    Viana MP, Lim S, Rafelski SM. Quantifying mitochondrial content in living cells. *Methods Cell Biol* (2015) **125**:77–93. doi:10.1016/bs.mcb.2014.10.003](https://sciwheel.com/work/bibliography/4345584)

[6.    Valente AJ, Maddalena LA, Robb EL, Moradi F, Stuart JA. A simple ImageJ macro tool for analyzing mitochondrial network morphology in mammalian cell culture. *Acta Histochem* (2017) **119**:315–326. doi:10.1016/j.acthis.2017.03.001](https://sciwheel.com/work/bibliography/3357652)

[7.    Zhang TY, Suen CY. A fast parallel algorithm for thinning digital patterns. *Commun ACM* (1984) **27**:236–239. doi:10.1145/357994.358023](https://sciwheel.com/work/bibliography/1100585)

[8.    Chaudhry A, Shi R, Luciani DS. A pipeline for multidimensional confocal analysis of mitochondrial morphology, function, and dynamics in pancreatic β-cells. *Am J Physiol Endocrinol Metab* (2020) **318**:E87–E101. doi:10.1152/ajpendo.00457.2019](https://sciwheel.com/work/bibliography/8048077)

[9.    Ouellet M, Guillebaud G, Gervais V, Lupien St-Pierre D, Germain M. A novel algorithm identifies stress-induced alterations in mitochondrial connectivity and inner membrane structure from confocal images. *PLoS Comput Biol* (2017) **13**:e1005612. doi:10.1371/journal.pcbi.1005612](https://sciwheel.com/work/bibliography/6660758)

[10.   Zamponi N, Zamponi E, Cannas SA, Billoni OV, Helguera PR, Chialvo DR. Mitochondrial network complexity emerges from fission/fusion dynamics. *Sci Rep* (2018) **8**:363. doi:10.1038/s41598-017-18351-5](https://sciwheel.com/work/bibliography/5413178)

[11.   Sukhorukov VM, Dikov D, Reichert AS, Meyer-Hermann M. Emergence of the mitochondrial reticulum from fission and fusion dynamics. *PLoS Comput Biol* (2012) **8**:e1002745. doi:10.1371/journal.pcbi.1002745](https://sciwheel.com/work/bibliography/4966662)

[12.   Lefebvre AEYT, Ma D, Kessenbrock K, Lawson DA, Digman MA. Automated segmentation and tracking of mitochondria in live-cell time-lapse images. *Nat Methods* (2021) **18**:1091–1102. doi:10.1038/s41592-021-01234-z](https://sciwheel.com/work/bibliography/11560010)

[13.   Ashok D, O’Rourke B. MitoWave: Spatiotemporal analysis of mitochondrial membrane potential fluctuations during I/R. *Biophys J* (2021) **120**:3261–3271. doi:10.1016/j.bpj.2021.05.033](https://sciwheel.com/work/bibliography/11415364)

[14.   Lihavainen E, Mäkelä J, Spelbrink JN, Ribeiro AS. Mytoe: automatic analysis of mitochondrial dynamics. *Bioinformatics* (2012) **28**:1050–1051. doi:10.1093/bioinformatics/bts073](https://sciwheel.com/work/bibliography/2919814)

[15.   Kandel J, Chou P, Eckmann DM. Automated detection of whole-cell mitochondrial motility and its dependence on cytoarchitectural integrity. *Biotechnol Bioeng* (2015) **112**:1395–1405. doi:10.1002/bit.25563](https://sciwheel.com/work/bibliography/12014872)