

MCcalc: analysis of membrane contacts from transmission electron microscopy images

text by Ilya Belevich

Electron Microscopy Unit, Institute of Biotechnology,
Helsinki Institute of Life Science,
University of Helsinki, Finland

Contents

Introduction and overview of the procedure.....	3
Expertise needed to implement the protocol	3
Requirements	4
Limitations	4
Details of the procedure.....	5
Sample size.	5
Magnification.....	5
Strategies for random sampling.	5
Saving images.	6
Basic image processing.....	6
Image calibration.....	6
Contrast normalization.....	6
Conversion to the 8-bit format.....	6
Saving images ready for the analysis.....	7
Blind segmentation.....	7
Calculation of contacts and distance distributions.	8
Main and secondary organelles.....	9
Probing distance.	11
Calculate in pixels.	11
Smoothing factor.....	11
Show result for object.	11
Histogram and Hist binning.	13
Averaged distribution histogram for all main objects.....	15
Extend rays/precision factor.	16
Detect contacts.....	16
Contact cut-off.....	16
Allow gaps, px.....	16

Export results to Matlab.....	16
Save results in Matlab or Excel format.....	17
Generate images for each object.	17
Highlights contacts using the Selection layer.....	17
Continue.....	17
Full procedure as a list.....	18
Supplementary information	24
Supplementary Table 1.....	24
Section 1: MCcalc settings and cumulative results	24
Section 2: MCcalc results for each main organelle of interest.....	25
Supplementary Table 2. Organization of results in the Excel format.....	27
Acknowledgments	28
References.....	28

Introduction and overview of the procedure

In this document we present MCcalc a plugin of MIB[1] for automatic detection of contacts between organelles of interest and calculation of distance distributions between them. The application of MCcalc is allowing to express the difference between phenotypes in numbers ready for statistical analysis. Importantly, to eliminate the errors due to human perception of the expected results, it is recommended to use the plugin together with the random sampling strategy during image acquisition and blind segmentation. The blind segmentation can be achieved using Rename and Shuffle tool of MIB, which anonymize collected images. The anonymized images are then segmented with a goal to segment the main organelle of interest (*eg.* mitochondria, lipid droplets) and the secondary organelle (*eg.* ER) between which the contacts and the distance distribution will be calculated. After the segmentation, the created models are de-anonymized back to match the imaged phenotypes and MCcalc plugin is applied. MCcalc uses the ray-tracing technique, when a beam, perpendicular to the surface of the main organelle, is generated at each point of the main organelle's perimeter and is followed until it hits the secondary organelle. When the beam hits the secondary organelle the travelled distance of the beam is recorded and used to extract:

1. Distances and their distribution between each two pairs of organelles of interest (*eg.* mitochondria and ER) across the main organelle's (mitochondria) perimeter
2. Perimeter of the main organelle (mitochondria) and absolute and relative number of hits
3. Detect number of contacts between the objects for each main organelle, calculate the length and mean distance of each contact, the ratio between the main organelle perimeter and the total length of the contact area

The results can be exported in the Matlab or Excel formats and can be supplemented with snapshots for each main organelle for presentation and visual examination.

The workflow is suitable for a broad scientific audience and in general does not require significant training. Details of the computational steps are described below, supplemented with video tutorials and can be easily repeated. The TEM imaging and analysis of two sets of specimens with about 20-30 images per specimen using this protocol can be done in one or two days depending on user's experience.

Expertise needed to implement the protocol

The protocol requires basic knowledge of TEM, image processing, image segmentation and approaches for data analysis. Access to TEM, standard sample preparation protocols and training are normally provided by a specialized core facility. Necessary image processing steps are described inside of the protocol and done using a freely available, open-source tool[1]. Here the image segmentation step is described briefly, due to existence of multiple tutorials (http://mib.helsinki.fi/tutorials_segmentation.html & http://mib.helsinki.fi/features_all.html). The results can be exported in the Microsoft Excel format as tables, ready for various statistical analysis.

Requirements

Equipment:

- Any modern transmission electron
- Basic computer for image processing, we recommend 16 or more Gb of RAM

Software:

- Microscopy Image Browser (<http://mib.helsinki.fi>)
- Microsoft Excel (Microsoft Corporation, Redmond, WA) or/and Matlab (Mathworks, Natick, MA)

Limitations

The method is based on the ray-tracing technique, when a single perpendicular beam is generated at each point of the main object's perimeter. When the beams are generated at the highly curved areas, the beams may have a large spread and thus may not be able to detect small remote secondary objects. This limitation, however, can be minimized when the "Extended rays" checkbox of the MCcalc plugin is checked. When checked, the MCcalc is using linear interpolation to fill the gaps between the remote ends of the rays vectors (Fig 1). This way the number of generated rays at the corners of the square areas on Fig.1 is increased. See also *Extend rays/precision factor* section of the Calculation of contacts and distance distributions chapter below.

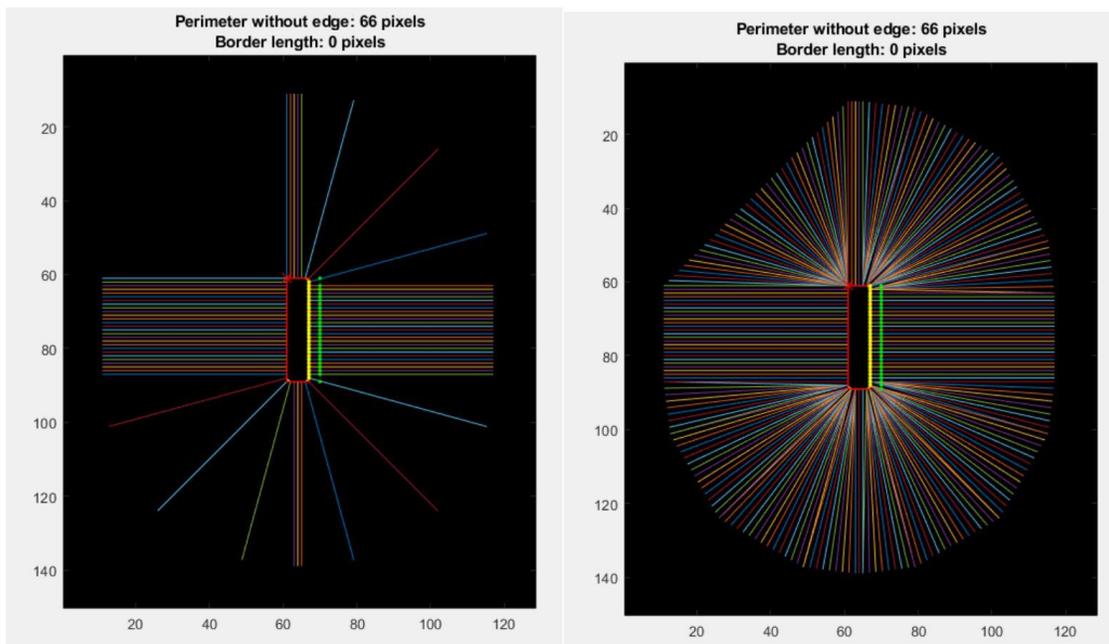


Figure 1. Demonstration of extended rays to minimize the gaps between the rays due to high curvature of the main object surface

Precision of contact detection and calculation of distance distributions between the organelles is based on the segmented areas, thus the quality of the modelling is essential (see more in the Blind segmentation section below).

Details of the procedure

Sample size. The number of collected images depends on natural abundance of the organelle of interest in cells. For abundant organelles (*eg.* mitochondria) it may be enough to obtain about 20 images (2 images / cell), while for rare ones (such as endogenous lipid droplets) one needs to collect much more. To obtain meaningful data, we recommend to obtain a minimum of 50 main organelles from at least 10 different cells. The best strategy is to spend at least 10 minutes before starting the data collection to get familiar with the sample and its variability. These will also help to select of the suitable magnification and to define a strategy for the random sampling.

Magnification. The suitable magnification should be selected before data collection. All images should be taken at the same magnification to minimize inconsistency between the images and to ease the image processing and analysis steps. In general, the magnification value should be high enough to enlarge potential contact sites between the organelles, while as low as possible to enlarge the imaging area. From our experience, the recommended magnification values for mitochondria or lipid droplets are about 5000x-6000x times (resulting in resolution of about 1.1-1.3 nm/pixel). As an option, the high magnification images can be extended with low magnification overviews including complete cells. This can be useful later during presentation of results, as it allows to place the high resolution images into the context of the corresponding cell.

Strategies for random sampling. It is highly recommended to utilize random sampling during the data collection. The random sampling strategy minimizes bias due to the human perception of results and is extremely important. Depending on the object of interest various strategies can be utilized:

- *Abundant organelles (eg. mitochondria):*

- a) At low magnification find a nucleus of a cell
- b) Take two (or any required amount) of high resolution images at 3 and 9 o'clock. At each image include the part of the nuclear envelope for the reference
- c) Continue to another cell

- *Rare organelles (eg. endogenous lipid droplets):* when the organelle of interest is rare the described above pattern may cause significant number of images without the organelles. To deal with that another strategy can be suggested:

- a) At low magnification find a nucleus of a cell

- b) Starting from 12 o'clock position follow the cell in the clock-wise direction until an organelle of interest is found
- c) Take an image
- d) Continue following the cell until the next object is found. When the complete circle is finished continue with another cell

Saving images. Normally, the microscope software provides various options for saving the images and several recommendations may be considered. It is the best to save images in formats that are compatible with the image processing software used during analysis; this minimizes number of data conversion steps. At the same time, the researchers should make sure that none of the corresponding imaging metadata is lost during the saving. In many situations, the latter requirement does not favor saving of images in the TIF format that does not have any common way to store the metadata, which may result in loss of valuable information – such as the pixel size and thus make the images unusable. We recommend to save the images in a native data format of the microscope, preferably using the 16-bit mode (this is not important for this particular workflow, but is a good practice in general). MIB used in the presented workflow is capable of handling various microscopy formats and in many situations additional data conversion is not needed.

Basic image processing.

The whole presented workflow is possible without doing the basic image processing step, but we still recommend to allocate time to check the datasets and prepare them for the analysis (for details check video at <https://youtu.be/EORf13flxIY>).

We depicted 4 most important steps:

Image calibration. To obtain sensible results the source images should be calibrated. During the calibration process we ensure that each pixel has correct physical size. Even though the whole workflow can be done using pixels as units, it is more reasonable to use physical units (μm) to represent the results. MIB is capable to correctly acquire correct pixel size for many microscopy formats, but in all cases it is important to recheck the calibration. If MIB is not able to identify the correct pixel size automatically, the images should be calibrated manually by providing the correct pixel size (*MIB->Menu->Dataset->Parameters*) or using a calibration procedure from an existing scale bar, which may be fused into the image (*MIB->Menu->Dataset->Scale bar*).

Contrast normalization, this step is particularly useful when working with TEM images. In many situations, the images may be taken using automatic exposure and as result the contrast and brightness of the images may fluctuate. The contrast normalization procedure is matching the average intensity of each image and its standard deviation to that of the whole dataset (*MIB->Menu->Image->Contrast->Normalize layers* or *Normalize layers based on masked areas*) producing a new dataset with normalized intensities between the images.

Conversion to the 8-bit format, we recommend to store the processed TEM images in the 8-bit format, which takes twice less space on a disk and offers twice faster performance, when comparing with the 16-bit format. Prior the conversion, it is required to specify the black and the white points (*MIB->View settings panel->Display*

button). All intensities between these two values will be recalculated to match the 8-bit container, *i.e.* all intensities below the specified black point will become 0s; all intensities above the white point become 255s; and all intensities between these two values will be stretched between 0 and 255 (*MIB->Menu->Image->Mode->8 bit*).

Saving images ready for the analysis. The last step of the basic image processing is to save the processed images to the disk. Here the few tips:

1. use an image format that is capable to preserve the pixel size. There are multiple options, but in this workflow we were using the AM (AmiraMesh) or TIF formats. It is possible to encode the physical image size into the standard XResolution and YResolution fields within the TIF format file. MIB and some other image processing software programs (*eg.* [Imaris](#) (Bitplane, Oxford Instruments) are capable of utilizing this information when loading images in the TIF format. Another sensible option is to use AmiraMesh format that is capable of storing multiple color channels and compatible with Fiji[2] and [Amira](#) (Thermo Fisher Scientific).
2. Save images not as 3D stacks but as individual 2D sections. This is a requirement for the following rename and shuffle step used during the blind modelling procedure. It is recommended to do not mix the images coming from different conditions during saving and keep each condition in its own directory.
3. Please make sure that the original (16-bit) files are preserved and not overwritten during this step.

The recommended organization of the folders:

- Condition 1
 - Low magnification overviews (*optional*)
 - High magnification original 16-bit images
 - High magnification processed 8-bit images
- Condition 2
 - Low magnification overviews (*optional*)
 - High magnification original 16-bit images
 - High magnification processed 8-bit images

...

Blind segmentation

Blind segmentation is an important part of the workflow - when the researcher does not know the source of the analyzed image, which makes the analysis completely unbiased. To achieve this, we suggest to use Rename and Shuffle tool of MIB (*MIB->Menu->File->Rename and Shuffle*). This tool takes images from number of source folders, renames them and randomly shuffles to a number of output folders. In addition to the shuffled images, the tool also creates a Matlab-compatible file with *.mibShuffle extension that is needed for correct restoration after blind segmentation. Whenever is needed this file can be supplemented with a sheet in the Microsoft Excel format with human-readable description of the shuffled procedure. These randomized images can be analyzed, modeled or classified based on their features without knowing their source phenotype.

Following the segmentation, the models or data annotations are restored back and loaded together with the original dataset for analysis. We also would like to note that the Rename and Shuffle tool is quite universal and can be used also for other workflows that require unbiased analysis of images (for details check video at <https://youtu.be/mqKGxq8ilYc>).

After renaming and shuffling, the images are loaded in MIB for segmentation. During the segmentation process, we ask a researcher to draw the main organelle of interest (*eg.* mitochondria or lipid droplets) and edges of the secondary organelles (*eg.* the ER) that are facing towards the main organelle. It is required to segment all of the main organelles of interest seen on the images; these objects can be completely within the image or be partially clipped. In the latter case, MCcalc will take into account only the unclipped part of the main organelle. Secondary organelles may not be segmented completely: it is required to segment only parts of those organelles that are within the probe distance (length of a generated ray) that will be used for analysis. In addition, it is not needed to segment the sides of the secondary organelles that are facing away from the main object. Both of these factors, decrease the segmentation time significantly without compromising the results. The only other requirement is careful segmentation at the areas of potential contacts. Whenever is possible, the manual segmentation can be replaced with the semi-automatic segmentation but for TEM images used in this example that was not possible (for details, check video at <https://youtu.be/N1EtxMciIQY>).

After the segmentation of the randomized sets, the models are restored back, so that the folders with original data are getting new model files with the combined segmentation for all original images (for details, check video at <https://youtu.be/lqpk-gVemlk>). The segmentation file can be loaded over the original dataset and further analyzed using MCcalc plugin.

Calculation of contacts and distance distributions.

Before starting the calculations images and the models of both main and secondary organelles of interest should be loaded into MIB. It is recommended to check the area sizes of the main organelles of interest to make sure that there are no holes and no accidental noise present in the model (please follow the description of the procedure in the **Full procedure as a list, section 6.1 (iii and iv)**, below, and video at https://youtu.be/Vf6_DgshGGk)

MCcalc can be started from *MIB->Menu->Plugins->Organelle analysis->MCcalc* (Fig. 2).

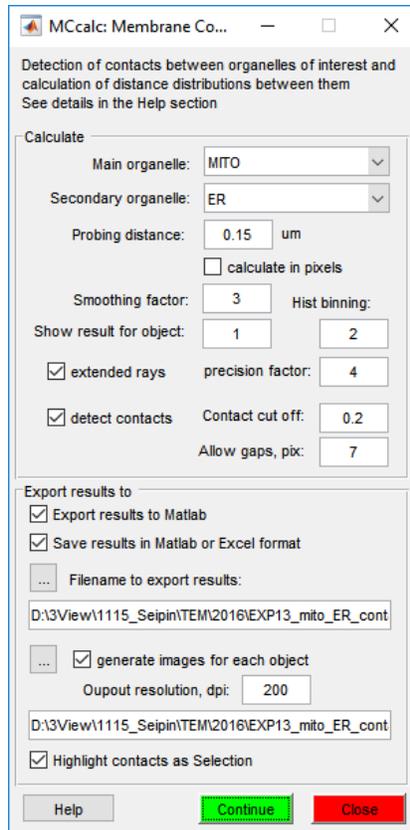


Figure 2. Graphical user interface of MCcalc plugin

MCcalc has a graphical user interface that offers the following configuration tweaks:

Main and secondary organelles. These combo boxes can be used to choose the main and the secondary organelle of interest. During analysis, each of the main organelles are identified and a set of rays is generated from each point of the organelle perimeter. The generated rays are followed until they hit the secondary organelle or reach the *Probing distance*. The detected hits are visualized on the resulting snapshots with green dots overlaid over the secondary organelle model (Fig. 3).

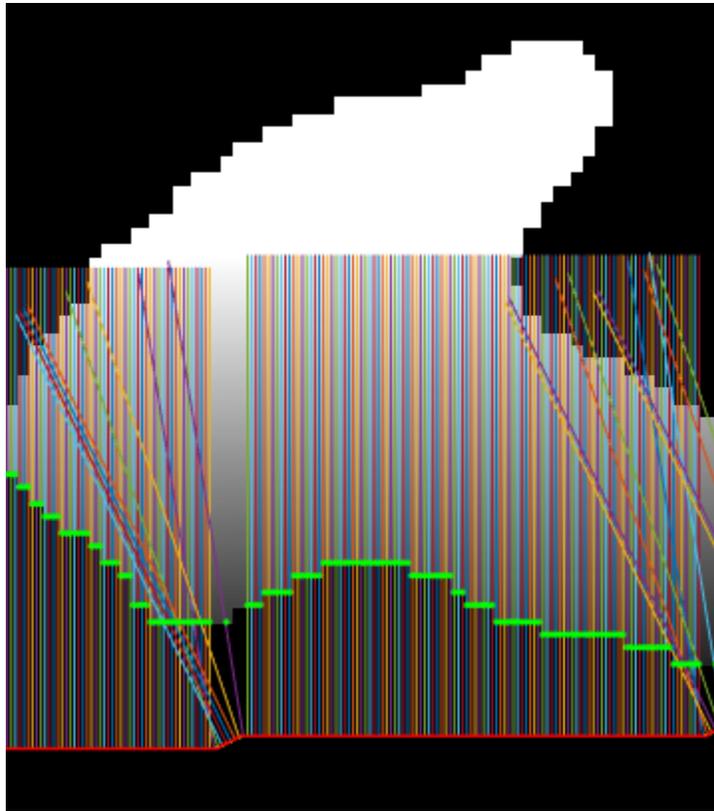
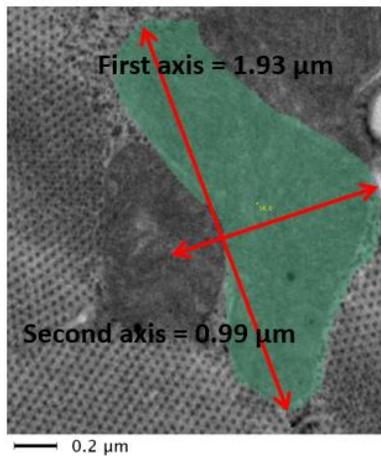


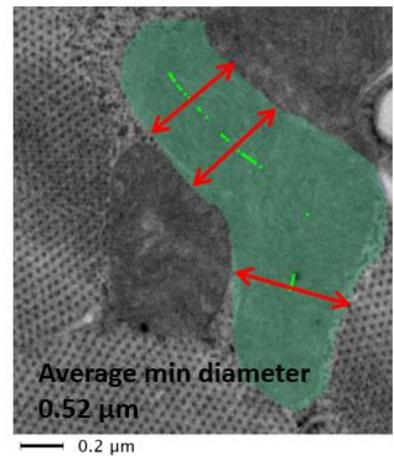
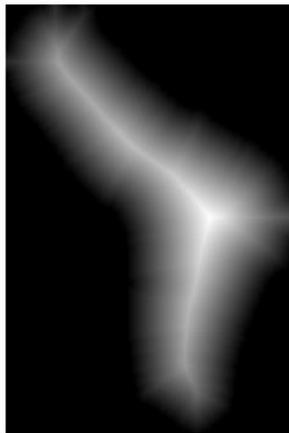
Figure 3. Visualization of rays generated at each point of the main organelle perimeter (red line). The detected hit points are shown in green. The outlines of the secondary organelle are shown in white.

For each main organelle MCcalc also calculates its area, eccentricity and its average minimal thickness:

Find regional maxima for the distance map and obtain distance to the edge at those points



Calculate distance map

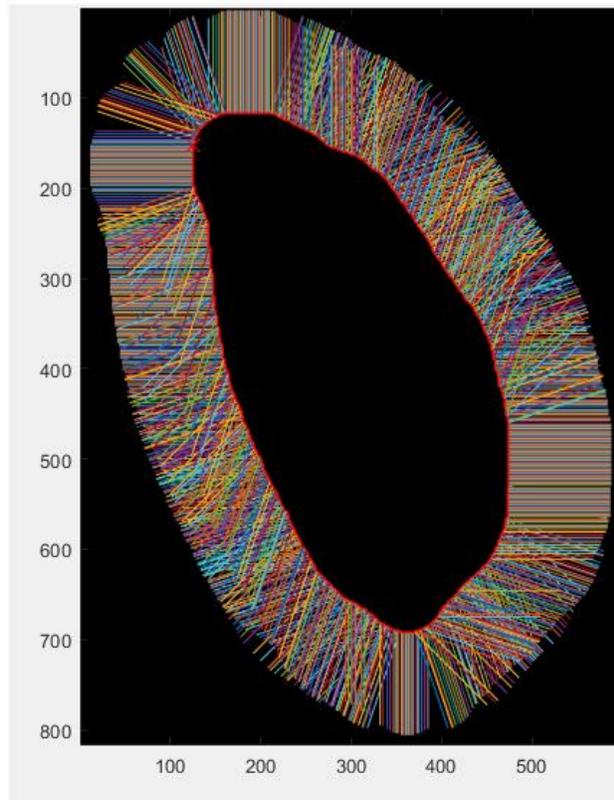


Probing distance. Since the contacts are the areas, where two organelles are coming close to each other, in many cases it is enough to follow the rays only for a short distance from the main organelle. The shorter probing distance improves performance of the plugin, however it should be longer than the *Contact cut-off* distance (see below).

Calculate in pixels. When this check box is ticked, MCcalc uses pixels as the units for reporting results; otherwise physical units (μm) are used. See the *Image calibration* section of the *Basic Image Processing* chapter above for details of the image calibration procedure.

Smoothing factor. The rays that are generated at each point of the main organelle's perimeter are the normal vectors. The normal vectors are perpendicular to the perimeter and their spread is defined by the local smoothness of the perimeter. To make spread of the rays more even, the perimeter smoothing factor can be used. The larger smoothing values give more even distribution of the rays (Fig. 4).

Smoothing factor: 0



Smoothing factor: 7

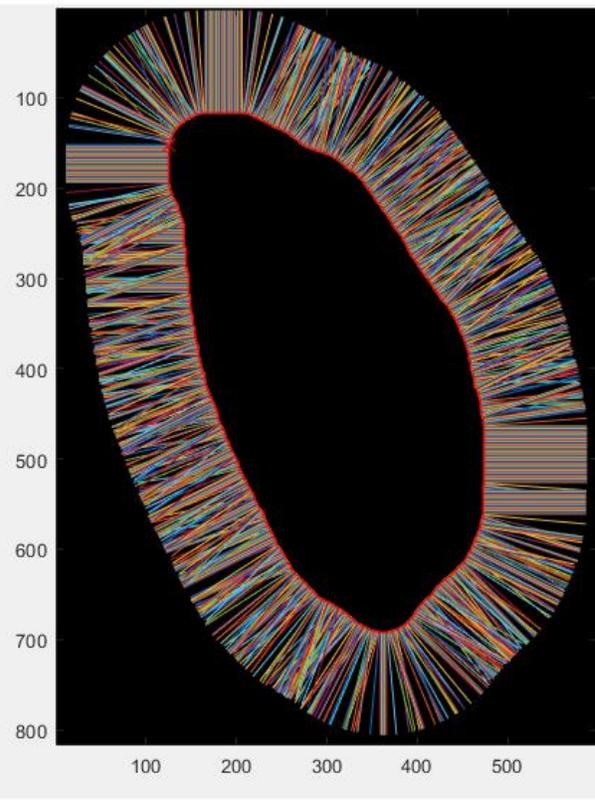


Figure 4. Increase of the smooth factor makes distribution of the generated rays more even.

Show result for object. This option was mainly designed for checking of the plugin settings. Here, one can specify an index of an object that belongs to the main organelle material and the results for the object with that index will be displayed in a separate window (Fig. 5). The figure has two panels: the upper panel is introducing the selected object, while the lower panel plots the normalized to the number of the generated rays distance distribution histogram between the selected organelle and all surrounding secondary organelles.

The upper panel reports:

- Title has information about the total perimeter of the organelle and the length of the clipped by the image frame area;
- The background of the figure is formed with an image that shows the secondary organelles in the proximity of the main organelle;
- The traced outline of the main organelle of interest is shown in red, a red cross demonstrate the point from where the tracing has started;
- The generated rays are displayed as straight lines originating from the traced outline of the main organelle and displayed with various colors;
- Positions, where the rays hit the secondary objects are highlighted in green;
- The detected contact areas (closer than the specified *Contact cut-off* distance) are shown in yellow;

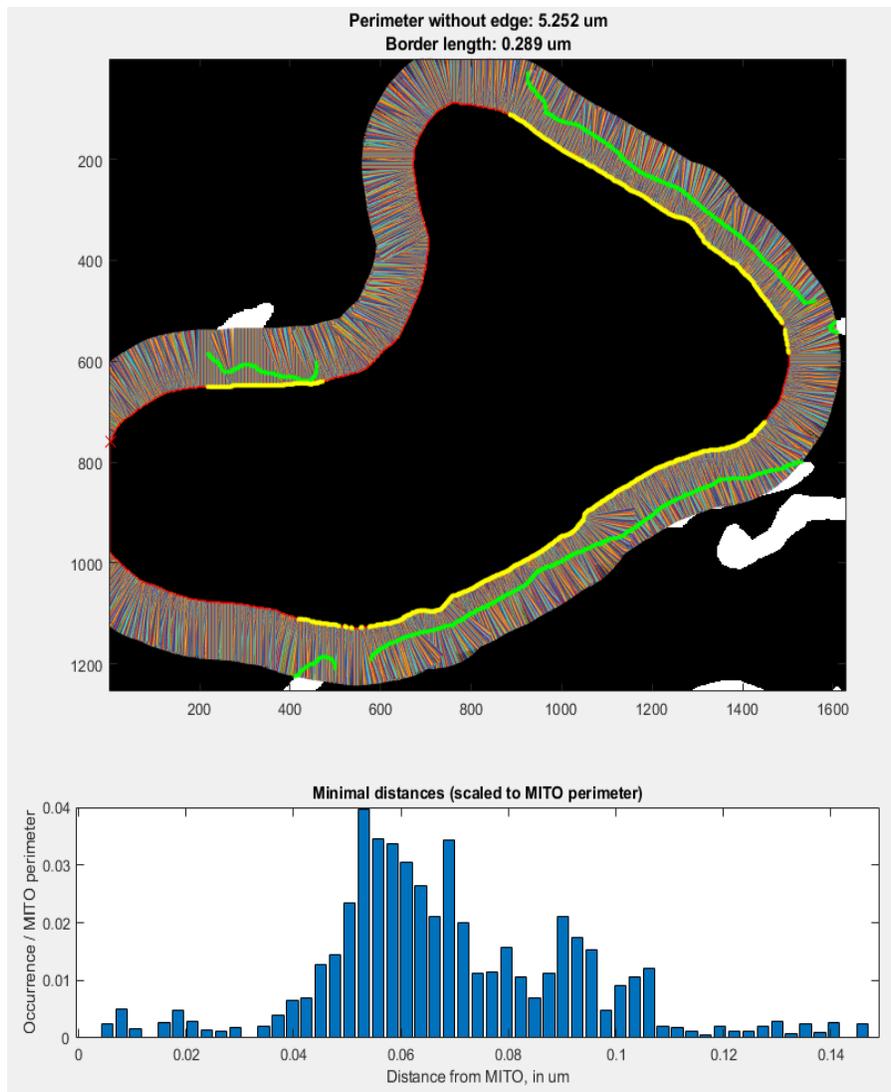
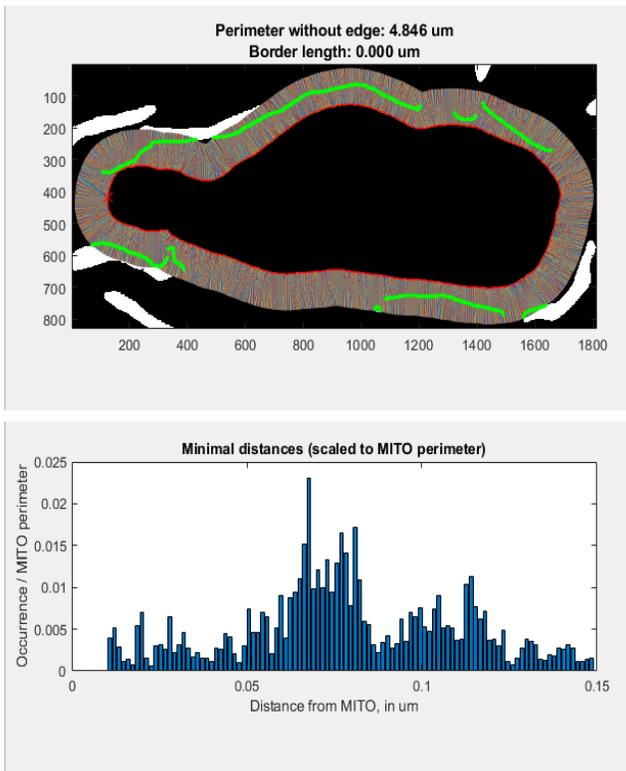


Figure 5. MCcalc results presented as an output image

Histogram and Hist binning. For each main organelle, MCcalc calculates the distance distribution between this particular organelle and all the secondary organelles within the probing distance. The distribution of the detected distances is plotted as a histogram graph. The precision of the histogram is defined using the *Hist binning* parameter. When this parameter is 1 the precision of the histogram is 1 pixel (Fig. 6).

Hist binning: 1



Hist binning: 4

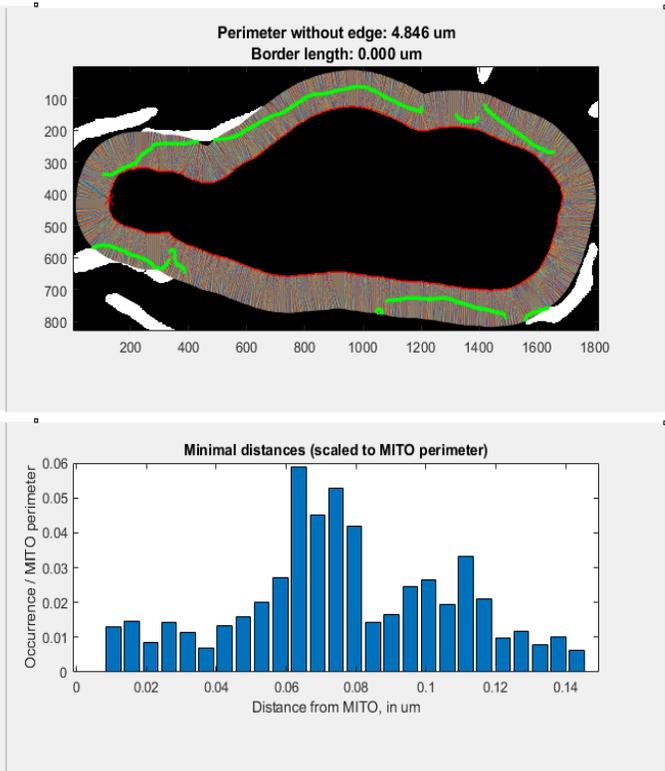


Figure 6. Representation of results using different histogram binning parameters.

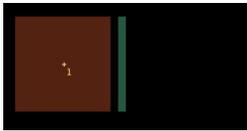
For each object MCcalc calculates two histograms:

- **distribution histogram of raw data**, showing number of detected hits at each distance. The distribution values are stored in `MCcalc(objId).DistributionMinDist` (or “Dist. of distances” sheet in the generated Excel file). The corresponding X-vector is stored in `MCcalc(1).DistributionCenters`. The result for object 2, can be plotted with Matlab command:

```
figure(2)
objId = 2;
bar(MCcalc(1).DistributionCenters, MCcalc(objId).DistributionMinDist);
```

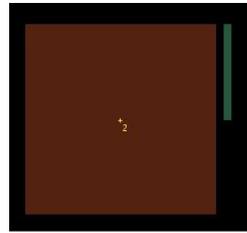
- **distribution histogram normalized to number of generated rays**, showing probability of the hit at each distance. The distribution values are stored in `MCcalc(objId).DistributionMinDistNorm` (or “Dist. of distances (norm)” sheet in the generated Excel file). The corresponding X-vector is stored in `MCcalc(1).DistributionCenters`. The result for object 2, can be plotted with Matlab command:

```
figure(2)
objId = 2;
bar(MCcalc(1).DistributionCenters, MCcalc(objId).DistributionMinDistNorm);
```



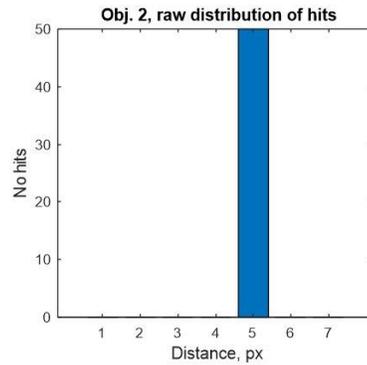
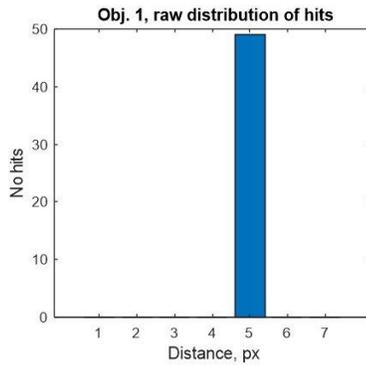
Object 1, 50x50 pixels

Secondary object:
4 pixels away with length 50 pixels

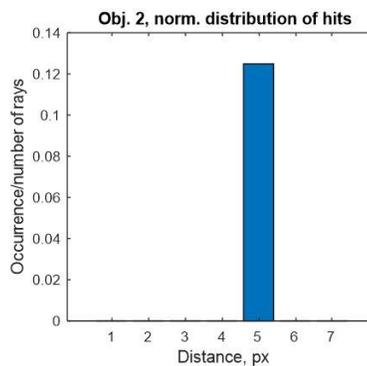
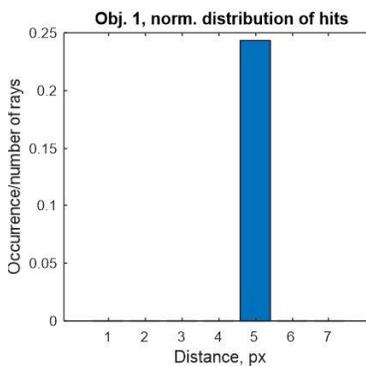


Object 2, 100x100 pixels

Secondary object:
4 pixels away with length 50 pixels



In the **raw** distribution both objects have the same number of hits (50), which is defined by the length of the secondary object (50 pixels)



In the normalized distribution the objects have different occurrence value:
25% for object 1, which matches the fact that the secondary object is detected completely across one of the sides
12% for object 2, which matches the fact that the secondary object is detected only for 1/2 of the side, giving in total 12%

Averaged distribution histogram for all main objects

In addition, at the end of procedure, MCcalc calculates averaged distribution histogram of hits for all detected objects of the main material. This average distribution is calculated by summarizing for each object **distribution histogram normalized to number of generated rays** at each distance and dividing it by number of main objects. The resulting plot is shown at the end of the run and combined in MCcalcExport(1).DistributionMinDistNormAv and in "Dist. of distances (norm)" sheet of the generated Excel file ("Averaged" column). In Matlab the resulting plot can be generated as

```
figure;
plot(MCcalcExport(1).DistributionCenters, MCcalcExport(1).DistributionMinDistNormAv);
xlabel(sprintf('Distance from %, in %, step=%f', ...
    cell2mat(MCcalcExport(1).mainMaterial), units, pixSize*histBinningEdit));
ylabel(sprintf('Occurrence/number of rays, averaged for all %s', ...
    cell2mat(MCcalcExport(1).mainMaterial)));
title(sprintf('Averaged results for all points (N=%d)', numel(MCcalcExport)));
```

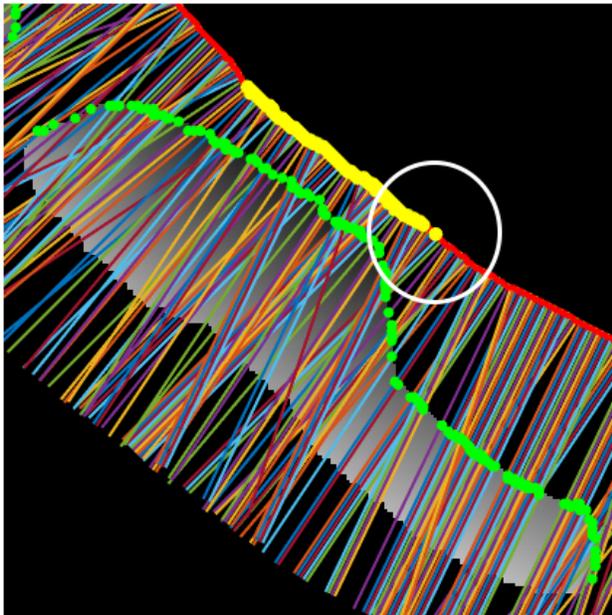
Extend rays/precision factor. When enabled, this option allows to extend number of generated rays at each point of the main organelle's perimeter (Fig. 1). Number of additional rays can be tweaked by the *precision factor* parameter. The smaller the value, more rays are generated but it takes longer to perform the calculations. See also the Limitation section.

Detect contacts. MCcalc is analyzing the distance distribution between the organelles of interest and using it for contact detection. The detected contacts are highlighted with the yellow dots on the snapshots.

Contact cut-off. This parameter specifies the maximal distance between two organelles that should be considered as a contact. When the distance between two organelles is smaller than this number, a point on the main organelle's perimeter is assigned to a contact and highlighted in yellow.

Allow gaps, px. Due to spread of the rays at the curved areas the main organelle (Fig. 1), a single contact may be broken to several smaller. The *Allow gaps, px* parameter allows to take care about this situation and when two contact areas are closer than this value to each other, they will be fused together and form a single contact (Fig. 7). The minimal value for this field should be above 1.42 (square root of 2), otherwise the contacts will have a size of a single pixel.

Allow gaps, px: 1.46



Allow gaps, px: 7

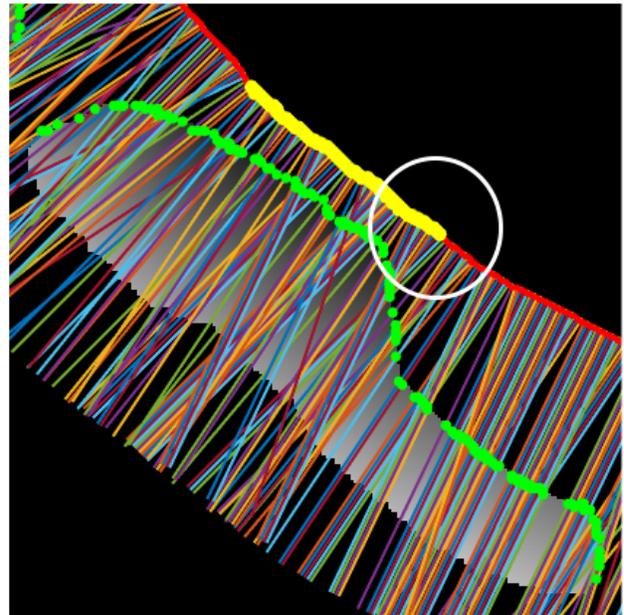


Figure 7. Fusing gaps in contacts due to spreading of the rays. A gap on the left panel has width less than 7 pixels and because of that the contact the two contacts are fused into a single contact, when allow gaps factor is set to 7 (right panel)

Export results to Matlab. When MIB is used under Matlab environment it is possible to export results as a structure directly to Matlab main workspace. The structure contains all detected values and offers possibility to continue with the analysis of results using other Matlab tools and function. Upon the press of this check box,

MCcalc asks a user to provide a name for the structure. The detailed description of the exported fields is available in the **Supplementary Table 1** (see below).

Save results in Matlab or Excel format. The results can also be saved in Matlab format (as a structure explained in the **Supplementary Table 1**) or as a spreadsheet in Microsoft Excel format. The latter case allows to use the MIB and MCcalc as a standalone program without installation of Matlab. The generated Excel-compatible file contains multiple sheets (**Supplementary Table 2**)

Generate images for each object. The results of the plugin can also be visualized and exported as images in PNG format. The output directory and the resolution of the images can be specified. The colors are the same as in for the snapshot described in the *Show result for object* section above.

Highlights contacts using the Selection layer. The detected contacts can be highlighted using the Selection layer of MIB. When highlighted, the contacts can be seen in the Image View panel of MIB as green lines at the contact areas (Fig. 8). **Attention!** The green lines belong to the selection layer of MIB and can accidentally be assigned to any other material of the model. It is recommended to use the Shift+C key shortcut of MIB to remove those highlighted areas after their examination.

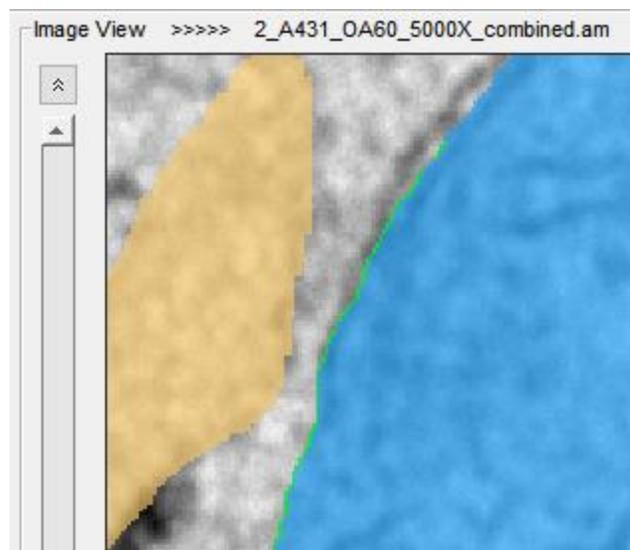


Figure 8. A snapshot from MIB user interface showing contact area highlighted in green using the Selection layer

Continue. Press the Continue button to start MCcalc. Performance of the plugin depends on the selected settings, for example use of the extended rays and generation of images for each object may slow down the performance significantly. Indices of the detected main organelles of interest will be labeled using the annotations in MIB (Fig. 9). The list of the annotations is available from *MIB->Menu->Models->Annotations->List of Annotations...*

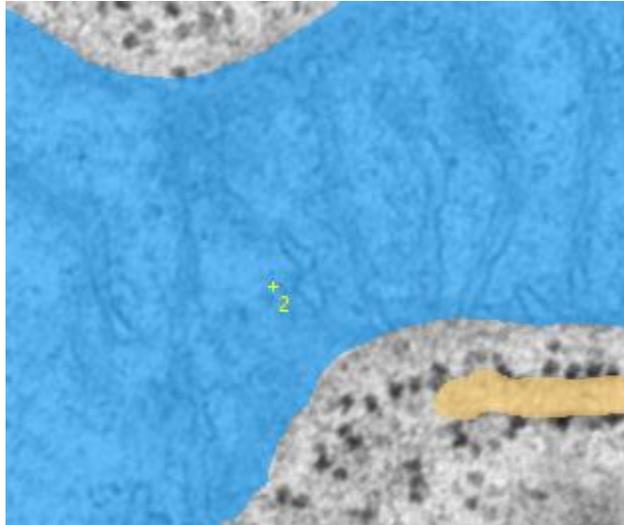


Figure 9. A snapshot from MIB user interface showing index of the detected object stored as the Annotation layer

Full procedure as a list

1. Sample preparation:

1.1 Prepare the specimens using a suitable TEM staining protocol

1.2 Trim the specimens from different conditions in the similar way. For example, take 3rd, 4th and 5th sections for cell monolayers

2. Definition of parameters for random sampling (timing: 15-20 min):

2.1 Start TEM session and get familiar with the specimens. Look at least several cells, preferably from each condition and identify the main organelle of interest. Try to estimate abundance of the main organelle of interest within the cells.

2.2 Define a pattern how to image the specimens. Evaluate and set the following factors:

(i) The low magnification for finding individual cells

(ii) The high magnification required for imaging. Try to find a balance between the magnifications to resolve the contact areas, while preserving as large as possible field of view. The recommended magnification for mitochondrial/ER contacts is x5000-x6000 (resolution of about 1.1-1.3 nm/pixel)

(iii) Required number of areas per cell

(iv) Required number of cells

(v) Decide on take or not the overview images of each cell

(vi) Imaging pattern of imaging within each single cell:

(A) *There are many main organelles of interest.* Use a nucleus as a landmark and obtain images at the certain orientation relative to the nucleus, *eg.* at 12 and 6 o'clock.

(B) *There are few main organelle of interest.* Use the high magnification and search for the main organelle of interest following the cell area in the predefined manner (*eg.* clock-wise, starting from 12 o'clock). When the main organelle of interest is found take an image using the pre-defined high magnification value.

2.3 Define the file format for saving the images. Consult with the core facility staff on the formats used by the microscope; make sure that all required meta data (*eg.* pixel size, beam power, microscope name, *etc.*) are also stored. It is preferable to use 16-bit format for saving the images.

3. Imaging of specimens:

3.1 Set the microscope to the low magnification value defined at **step 2.2(i)**

3.2 Find a cell

3.3 [*Optional*] take an overview image, save the overview image to disk

3.4 Zoom-in to the starting point of the imaging pattern (**step 2.2(vi)**)

3.5 Obtain an image using the specified high magnification value (**step 2.2(ii)**)

3.6 Save the image to disk

3.7 Proceed across all positions specified in the imaging pattern and take images at the high magnification

3.8 Repeat **steps 3.1-3.7** for as many cells as needed

3.9 Change the specimen and repeat the imaging (**steps 3.1-3.8**)

4. Basic image processing (timing: 10-60 min):

4.1 Arrange the images in folders; keep the structure of the folders as described in the section Basic image processing/*Saving images ready for the analysis* above

4.2 Download and install MIB (<http://mib.helsinki.fi/>)

4.3 Start MIB

4.4 Load into MIB high resolution images belonging to the same condition (see [youtube-demo, part 1](#)). To load images stored in the proprietary formats please use the Bio-Formats[3] reader (check the *Bio* checkbox of the Directory Contents panel of MIB)

4.5 Calibrate the pixel size (*MIB->Menu->Dataset->Preferences*, [youtube-demo, part 1](#))

4.6 Perform contrast normalization (*MIB->Menu->Image->Contrast->Normalize layers->Z stack*, [youtube-demo, part 1](#))

4.7 Convert images to the 8-bit format (*MIB->Menu->Image->Mode->8 bit*, [youtube-demo, part 1](#))

4.8 Save images as 2D slices to a separate folder (*MIB->File->Save image as...*). Use the AM (*AmiraMesh*) or TIF format (*TIF format LZW compressed*) and the Sequence of 2D files with the original naming ([youtube-demo, part 1](#))

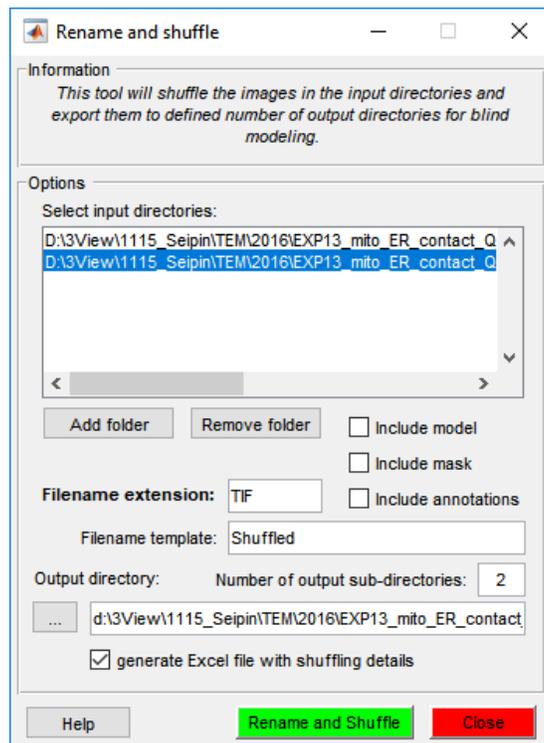
5. Blind modeling:

5.1 Renaming and shuffling of images ([youtube-demo 2](#))

(i) start MIB

(ii) navigate to the top folder containing all subfolders with the imaged conditions (*MIB->Path panel*)

(iii) Start Rename and Shuffle tool (*MIB->Menu->File->Rename and Shuffle->Rename and Shuffle...*) (Fig. 10)



MCcalc Figure 10. User interface of the Rename and Shuffle tool of MIB

(iv) Using the *Add folder* button, add folders with the 8-bit processed TIF images. If a wrong folder was added, it can be removed using the *Remove folder* button.

(v) Make sure the *Filename extension* is TIF

(vi) Set number of output sub-directories to 1. This parameter allows to shuffle input directories to a number of output directories. In this example, we use only a single output directory

(vii) Specify the Output directory

(viii) Make sure that the generate Excel file with shuffling details is enabled

(ix) Press the *Rename and Shuffle* button. The tool will rename and shuffle the images from the input directories and copies them to the destination directory (the images will be placed into Subset_001, Subset_002, etc. directories). In addition, an Excel sheet with the performed conversion and a Matlab-compatible file (*.mibShuffle) required for the reverse operation will also be created.

5.2 Modeling of objects of interest. There are number of tutorials available on MIB website (<http://mib.helsinki.fi>), but here we provide a short description of the segmentation process using the brush tool ([youtube-demo 3](#))

(i) Load the randomized images

(ii) Create a new model (*MIB->Segmentation Panel->Create->63 material model*)

(iii) Press the “+” button to add two materials one for the main (mitochondria in this example) and secondary (ER) organelle of interest (*MIB->Segmentation Panel->+*) and name them accordingly. When working with multiple output directories (**step 5.1(vi)**) it is important to have the sequence of materials the same for all models, *i.e.* the first material should always be the main organelle and the second - the secondary organelle of interest

(iv) Select the brush tool (*MIB->Segmentation Panel*)

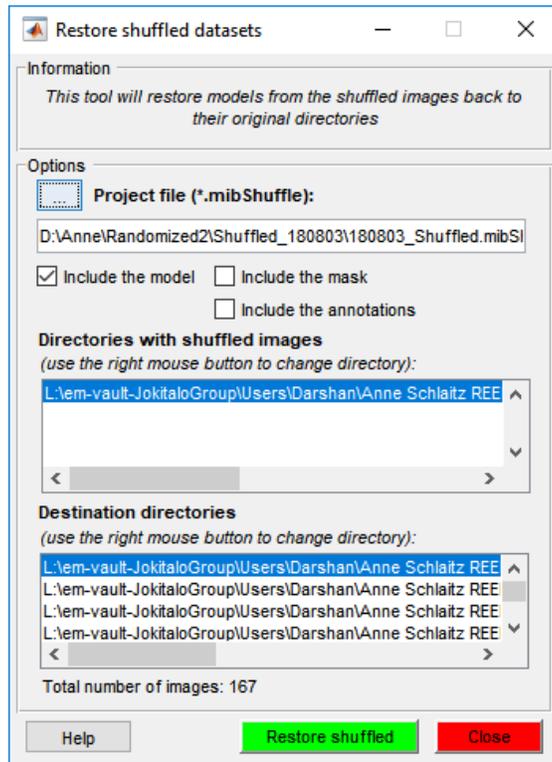
(v) Using the brush tool draw the shapes (press and hold the left mouse button) of all main organelles of interest and the sides of the secondary organelles that are facing them. List of most useful key shortcuts:

- ‘**C**’ - clear the selected areas
- ‘**A**’ - add selected area to the highlighted in the Segmentation table material of the model
- ‘**S**’ – subtract selected area from the materials of the model
- ‘**F**’ – fill holes in the selected areas
- ‘**Ctrl+Z**’ – undo the last action
- ‘**Ctrl+left mouse button**’ – starts the eraser tool

(vi) Save the model to disk (Menu->Models->Save model as...->Matlab format (*.model))

5.3 Re-shuffling the models for analysis. The aim of the re-shuffling step is to re-shuffle the generated models so that they match the images of the original dataset ([youtube-demo 4](#)).

(i) Start the Restore mode of the Rename and Shuffle tool (MIB->Menu->File->Rename and Shuffle->Restore)



MCcalc Figure 11. User interface of the Restore shuffled datasets tool of MIB

(ii) Select a project file in *.mibShuffle (Options->...)

(iii) Make sure that the Include the model check box is selected

(iv) Press the “Restore shuffled” button to generate a new model files in each of the destination directories. The model file will be named as “Labels_RestoreRand_XXXXXX.model”, where encode the date when the restoration procedure was done.

TROUBLESHOOTING

Sometimes the directories with the shuffled images and destination can no longer be the same. In this case it is possible to manually update the paths of the each directory. This can be done by selecting the directory in the list and use the right mouse button to call a popup menu. Select the ‘Update directory’ option to specify a new path to each of the imaging conditions.

6. Detection of contacts and analysis:

6.1 Loading of the model and the quality check ([youtube-demo 5](#))

(i) Load with MIB images taken at certain conditions

(ii) Load the corresponding model (*MIB->Segmentation panel->Load*) generated after **step 5.3**

(iii) Check the main material for presence of small irrelevant objects.

- Highlight the main material in the Segmentation table
- Right click using the mouse button to start a popup menu
- Start the *Get Statistics...* tool from the popup menu
- Press the *Run* button of the *Get Statistics* tool to calculate areas of the main objects
- Check the *Statistics* table, if small objects are found highlight by providing their largest value into the second edit box of the *Highlight range* panel. Press the *Do* button to highlight the objects smaller than the provided value in the *Image View* panel of MIB
- Make sure that the main window of MIB is in focus and press the Shift+S key shortcut to subtract the highlighted small objects from the model
- Save the model (*MIB->Menu->Models->Save model as...*)

(iv) Make sure that objects of the main material do not have holes

- Highlight the main material in the Segmentation table
- Press Alt+A key shortcut (or right click, Material to Selection->New 3D stack)
- Press Shift+F key shortcut to fill holes for all slices
- Press Shift+A key shortcut to add the filled area to the main material of the model

6.2 Calculation of contacts and distance distributions between the organelles ([youtube-demo 5](#))

(i) Start MCcalc plugin (*MIB->Menu->Plugins->Organelle Analysis->MCcalc*)

(ii) Specify options for the detection and filenames for results. Please refer to the *Calculation of contacts and distance distributions* section above for details.

(iii) Press the *Continue* button to start the plugin

Supplementary information

Supplementary Table 1

Section 1: MCcalc settings and cumulative results

Field name	Data type	Units / Contents	Description
<i>MCcalc(1).mainMaterial</i>	cell	text	name of the main organelle of interest
<i>MCcalc(1).secondaryMaterial</i>	cell	text	name of the secondary organelle of interest
<i>MCcalc(1).probingRangeInUnits</i>	numeric	<i>MCcalc(1).units</i>	maximal probing distance for the rays in physical units or pixels
<i>MCcalc(1).units</i>	string	µm / pixels	definition of the selected units: µm or pixels
<i>MCcalc(1).smoothing</i>	numeric	pixels	smoothing factor for the edge of the perimeter of the main organelles
<i>MCcalc(1).extendRays</i>	logical	0 / 1	a logical switch, whether the extended mode for rays was used (1) or not (0)
<i>MCcalc(1).extendRaysFactor</i>	numeric	pixels	precision of the extended rays
<i>MCcalc(1).contactCutOff</i>	numeric	<i>MCcalc(1).units</i>	maximal distance in the selected units for detection of contacts
<i>MCcalc(1).contactGapWidth</i>	numeric	pixels	maximal allowed distance between two detected contacts that defines whether the contacts will be fused or kept separate
<i>MCcalc(1).pixSize</i>	numeric	<i>MCcalc(1).units</i>	width of pixels in the selected units
<i>MCcalc(1).histBinningEdit</i>	numeric	<i>pixels</i>	step used for generation of the histograms, defines positions of <i>MCcalc(1).DistributionCenters</i>
<i>MCcalc(1).DistributionCenters</i>	numeric array	<i>MCcalc(1).units</i>	x-axis for the distributions, i.e. <i>DistributionMinDist</i> and <i>DistributionMinDistNorm</i> fields in the section 2
<i>MCcalc(1).DistributionMinDistNormAv</i>	numeric array	<i>counts</i>	Averaged normalized to <i>MCcalc(objId).mainPerimeterNoPixels</i> distribution of minimal distances for each organelle pooled together for all organelles; the X-axis is stored in <i>MCcalcExport(1).DistributionCenters</i> field

Section 2: MCcalc results for each main organelle of interest

Field name	Data type	Units/Contents	Description
<i>MCcalc(objId).Centroid</i>	2-elements numeric	pixels	coordinates of centroid for the detected organelle
<i>MCcalc(objId).slice</i>	numeric	index	index of a file of the original dataset containing the detected organelle
<i>MCcalc(objId).time</i>	numeric	index	index of a time point containing the detected organelle
<i>MCcalc(objId).MinDist</i>	numeric array	<i>MCcalc(1).units</i>	an array with distances between main and secondary organelles for each ray with a hit
<i>MCcalc(objId).rayDestinationPosX</i>	numeric array	pixels	an array with X-coordinates of points where rays hit the secondary organelle
<i>MCcalc(objId).rayDestinationPosY</i>	numeric array	pixels	an array with Y-coordinates of points where rays hit the secondary organelle
<i>MCcalc(objId).raySourcePosX</i>	numeric array	pixels	an array with X-coordinates of the originating points from where the rays were generated
<i>MCcalc(objId).raySourcePosY</i>	numeric array	pixels	an array with Y-coordinates of the originating points from where the rays were generated
<i>MCcalc(objId).mainPerimeterWithoutBorder</i>	numeric	<i>MCcalc(1).units</i>	Length of the detected organelle perimeter, excluding clipped areas at edges of the image
<i>MCcalc(objId).mainPerimeterBorder</i>	numeric	<i>MCcalc(1).units</i>	Length of the clipped areas at edges of the image
<i>MCcalc(objId).mainPerimeterNoPixels</i>	numeric	counts	Total number of points from where the rays were generated, excluding clipped areas at edges of the image
<i>MCcalc(objId).mainMinThicknessUnits</i>	numeric	<i>MCcalc(1).units</i>	Averaged minimal thickness of the main object
<i>MCcalc(objId).mainAreaUnits</i>	numeric	<i>MCcalc(1).units</i>	Area of the main object
<i>MCcalc(objId).mainEccentricity</i>	numeric	value	Eccentricity of the main object
<i>MCcalc(objId).secondaryHitsNoPixels</i>	numeric	counts	Total number of points where the rays hit the secondary organelles
<i>MCcalc(objId).DistributionMinDist</i>	numeric array	counts	Distribution of minimal distances for each organelle; the X-axis is stored in <i>MCcalcExport(1).DistributionCenters</i> field
<i>MCcalc(objId).DistributionMinDistNorm</i>	numeric array	counts	Normalized to <i>MCcalc(objId).mainPerimeterNoPixels</i> distribution of minimal distances for each organelle; the X-axis is stored in <i>MCcalcExport(1).DistributionCenters</i> field
<i>MCcalc(objId).cutOffContactX</i>	numeric array	pixels	an array with X-coordinates of the detected contacts

<i>MCcalc(objId).cutOffContactY</i>	numeric array pixels	an array with Y-coordinates of the detected contacts
<i>MCcalc(objId).contactLength</i>	numeric array <i>MCcalc(1).units</i>	an array with lengths of the detected contacts
<i>MCcalc(objId).contactMeanDistance</i>	numeric array <i>MCcalc(1).units</i>	an array with average distance at the contact between the organelles of interest

where **objId** is a linear index of the detected main organelle of interest

Supplementary Table 2. Organization of results in the Excel format

Sheet name	Field	Description
General results	Image directory	path to the original dataset
	Main object material	name of the secondary organelle of interest
	Secondary object material	name of the secondary organelle of interest
	Model filename	full path to the model containing segmented main and secondary objects
	Pixel size/units	size of pixels and the unit used
	Probing range	maximal probing distance for the rays in physical units or pixels
	Smoothing of rays, px	smoothing factor for the edge of the perimeter of the main organelles
	Histogram bin factor	step used for generation of the histograms, defines positions of <i>Distance, um</i> in the <i>Dist. of distances</i> sheets
	Extended rays, logical	a logical switch, whether the extended mode for rays was used (1) or not (0)
	Extend rays precision, px	precision of the extended rays
	ObjId	Linear index of the object
	Time	index of a time point containing the detected organelle
	Z, index	index of a file of the original dataset containing the detected organelle
	X, px	X-coordinate of the centroid of the detected organelle
	Y, px	Y-coordinate of the centroid of the detected organelle
	Perimeter without border, units	Length of the main organelle perimeter, excluding clipped areas at edges of the image
	Border perimeter, units	Length of the clipped areas at edges of the image
	No pixels of perimeter without border	Total number of points from where the rays were generated, excluding clipped areas at edges of the image
	No of hit pixels	Total number of points where the rays hit the secondary organelles
	MinThickness, um	Averaged minimal thickness of the main object
Area, um	Area on the main object	
Eccentricity, 0-circle	Eccentricity of the main object 0-circle, 1-line	
Dist. of distances	Distance, units	x-axis for the distributions, i.e. distance from the main object with precision defined in the <i>Histogram bin factor</i> field
	Obj. 1, Obj. 2 etc	number of hits detected at the <i>Distance, units</i> for each main organelle
Dist. of distances (norm)	Distance, units	x-axis for the distributions, i.e. distance from the main object with precision defined in the <i>Histogram bin factor</i> field
	Obj. 1, Obj. 2 etc	normalized to <i>No pixels of perimeter without border</i> distribution of number of hits detected at the <i>Distance, units</i> for each main organelle

	Averaged, units	Averaged normalized distribution for all main organelles
Contacts	Cut off distance	maximal distance in the selected units for detection of contacts
	Fuse contacts with brakes smaller than	maximal allowed distance between two detected contacts that defines whether the contacts will be fused or kept separate
	Obj. Id	linear index of the object
	Perimeter without border, units	length of the main organelle perimeter, excluding clipped areas at edges of the image
	Number of contacts	total number of detected contacts for each organelle
	Sum length of contacts, units	total length of all detected contacts for each organelle
	Ratio Contacts/Obj	ratio of the contact length to the total length of the perimeter without the border
	Length of each contact, units	length of each detected contact
RAW dist to hits	ObjId: 1, 2, etc	distances between main and secondary organelles for each ray with a hit

Acknowledgments

For calculation of the normal vectors MCcalc is using LineNormals2D function from the “2D Line Curvature and Normals” package, written by Dirk-Jan Kroon, University of Twente / Focal Machine Vision en Optical Systems 2011 (<https://se.mathworks.com/matlabcentral/fileexchange/32696-2d-line-curvature-and-normals>)

References

1. Belevich I, Joensuu M, Kumar D, Vihinen H, Jokitalo E. Microscopy Image Browser: A Platform for Segmentation and Analysis of Multidimensional Datasets. Plos Biology. 2016;14(1):e1002340. doi: 10.1371/journal.pbio.1002340. PubMed PMID: WOS:000371882900009.
2. Schindelin J, Arganda-Carreras I, Frise E, Kaynig V, Longair M, Pietzsch T, et al. Fiji: an open-source platform for biological-image analysis. Nat Methods. 2012;9(7):676-82. Epub 2012/06/30. doi: 10.1038/nmeth.2019. PubMed PMID: 22743772; PubMed Central PMCID: PMC3855844.
3. Linkert M, Rueden CT, Allan C, Burel JM, Moore W, Patterson A, et al. Metadata matters: access to image data in the real world. J Cell Biol. 2010;189(5):777-82. Epub 2010/06/02. doi: 10.1083/jcb.201004104. PubMed PMID: 20513764; PubMed Central PMCID: PMC3855844.