



# Tutorial

## Overview

Welcome to MPicker! To install the software, please refer to the **Installation** and **Download** pages. For detailed information about the GUI, consult the **Manual** page. Explore the **Advanced Tutorial** for tasks such as flattening complex surfaces in mesh, automatic particle picking, and Class2D. Before exploring other tutorial sections, make sure you finish the **basic usage** part.

## Basic usage

This tutorial guides you through the process of flattening membranes in the cyanobacterium Anabaena. Download the tutorial file `MPicker_tutorial_v1.0.0.tar.bz2` to a directory of your choice and extract it:

```
tar -jxvf MPicker_tutorial_v1.0.0.tar.bz2
cd tutorial
```

The tutorial includes three files:

- `emd13771_isonet.mrc` : CryoET data from EMD-13771, processed by isonet and cropped to a small size.
- `seg_post_id0_thres_0.60_gauss_0.50_voxel_150.mrc` : Membrane segmentation of `emd13771_isonet.mrc`. You can generate it yourself using MPicker.
- `manual_surf.txt` : Manually picked points used to label a membrane, demonstrating how to flatten a membrane without segmentation.

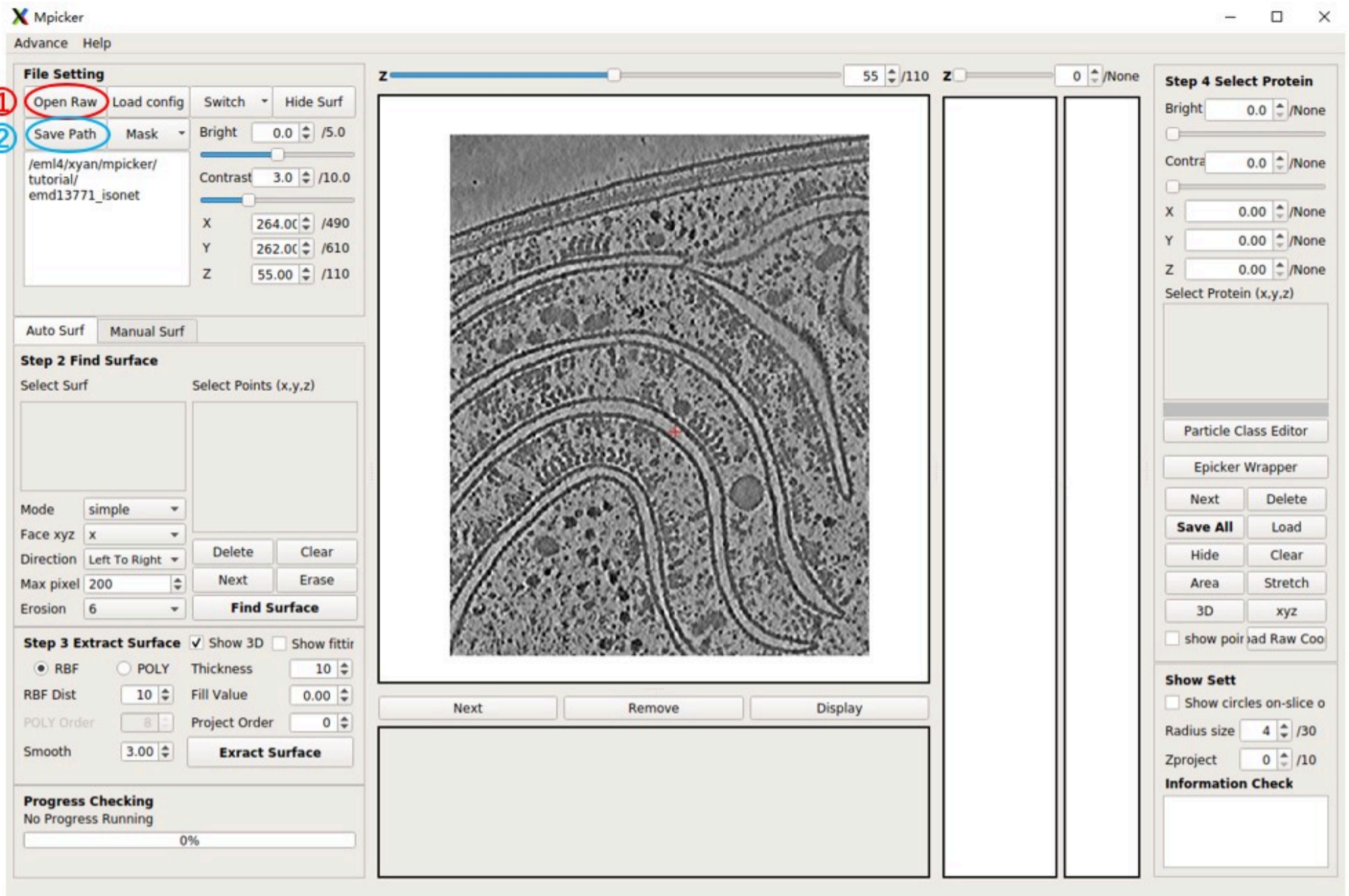
### • Load the data

Activate the conda environment and ensure that `mpicker/mpicker_gui` is in your `$PATH`. Open MPicker:

```
Mpicker_gui.py &
```

Press `Open Raw` and select `emd13771_isonet.mrc` to load it. Use mouse actions to navigate (drag to move, mouse wheel to zoom), and adjust image contrast via the spinbox of Bright and Contrast. Display xy slices at different z positions by scrolling the mouse wheel over the spinbox of z or dragging the bar.

Press `Save Path` and choose a folder to save the result (e.g., the current path `xxx/tutorial`). MPicker will create a folder `xxx/tutorial/emd13771_isonet` to save the result for this tomogram, and a file `xxx/tutorial/emd13771_isonet.config` for reloading your project later.



## • Load membrane segmentation

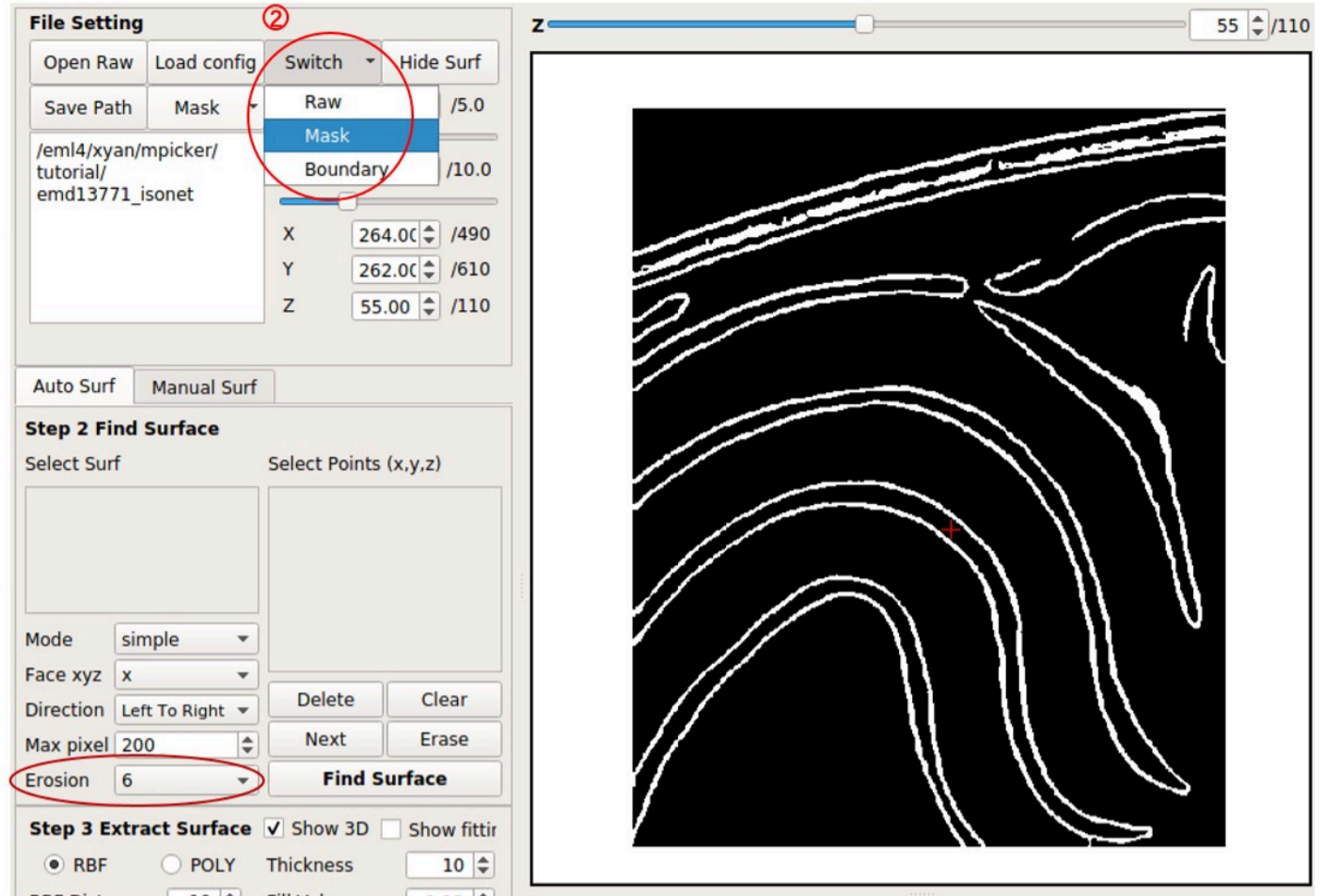
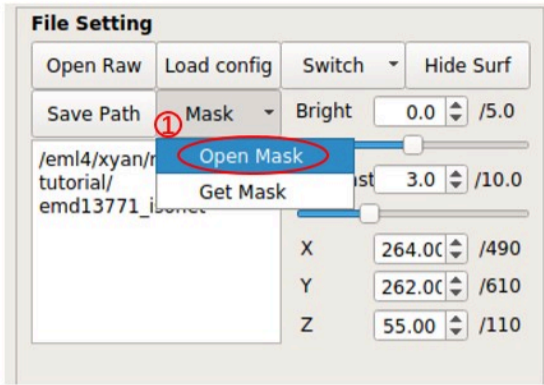
While we provide a segmentation, you can also generate it in MPicker. Refer to the **Generate membrane segmentation** section for details.

Press **Mask**, select **Open Mask**, and choose the segmentation file `seg_post_id0_thres_0.60_gauss_0.50_voxel_150.mrc`.

MPicker will load it and calculate its boundary, which may take some time. It will generate a new file `tutorial/emd13771_isonet/my_boundary_6.mrc`.

You can toggle between the raw tomogram, segmentation, or boundary by pressing **Switch** and choosing Raw, Mask, or Boundary. The boundary is the essential component for subsequent tasks, serving as the point cloud representation of the entire surface.

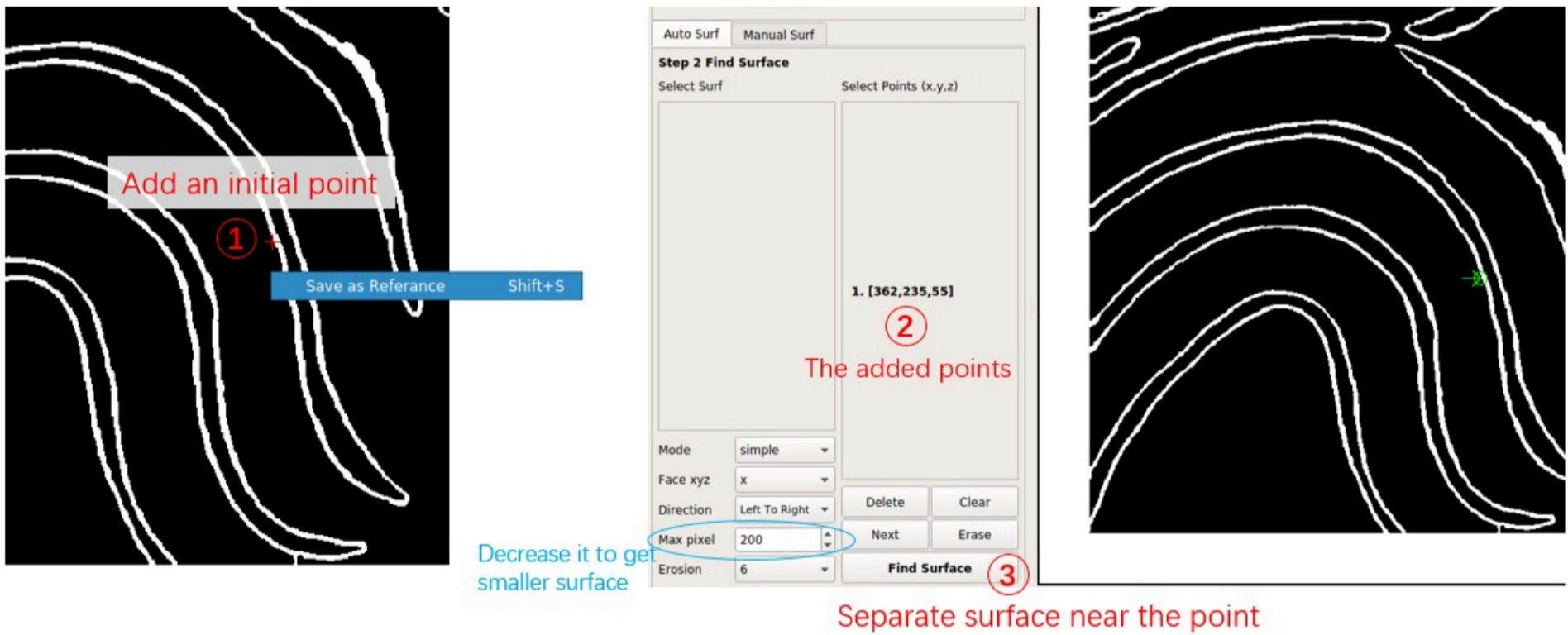
Raw: input tomogram  
 Mask: membrane segmentation  
 Boundary: boundary got from Mask



There are 4 types of boundary you can use. Hover the cursor on it for details.

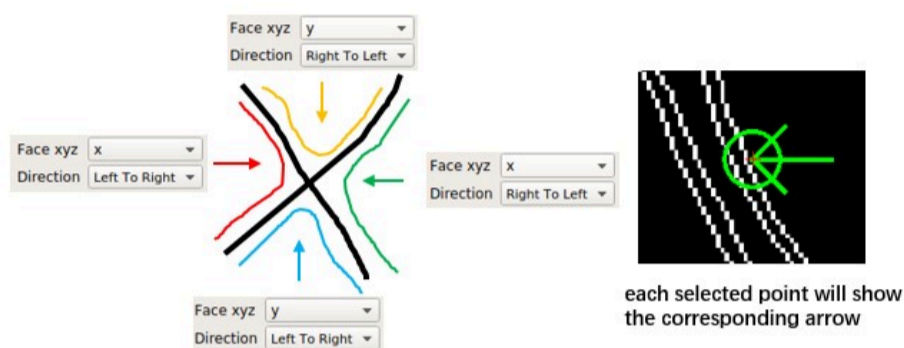
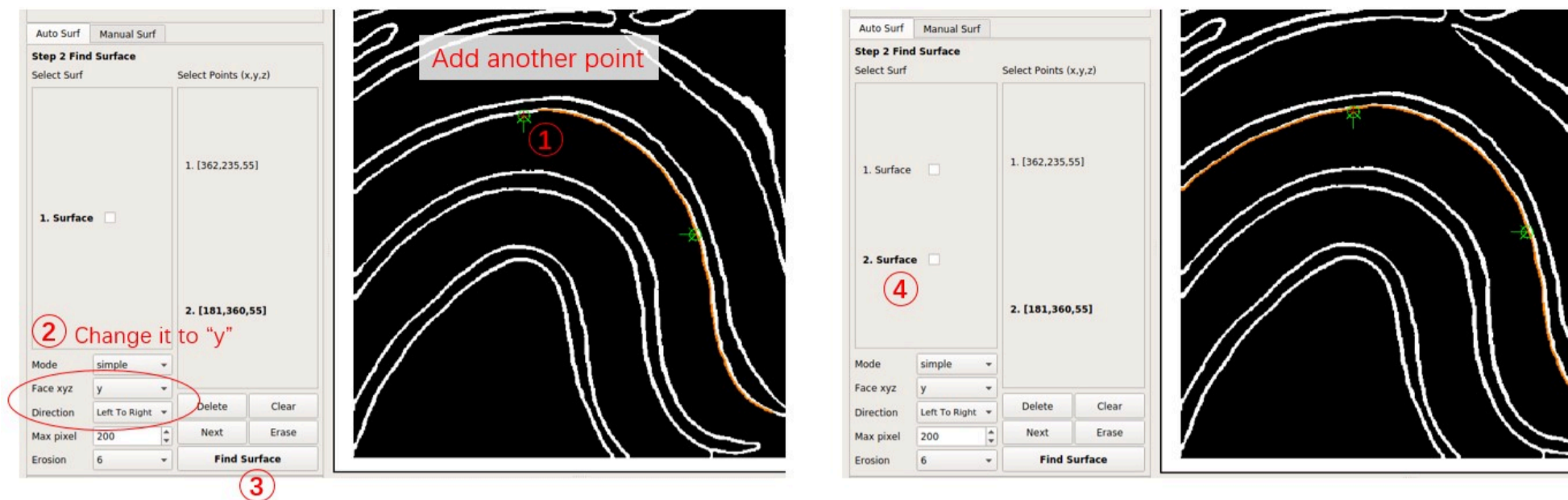
## • Separate a Surface

Move the red cross to a point near the membrane surface by left-clicking it. Then press **Shift + S** or right-click and choose **Save as Reference**. The point will be added to the **Select Points** list, marked by green arrows. Press **Find Surface** to separate the surface near the point.



Now you have the first separated surface, shown in orange. The orange line ends at the top because the **Face xyz** of the point is **x**, indicating that it separates the surface perpendicular to the x-axis, or the surface can be represented as  $x = f(y, z)$ .

To add the top region to the surface, add another point based on the first surface. Click near the membrane surface on the top region, press **Shift + S** as before, and change **Face xyz** to **y**. Each point can have different **Face xyz** and **Direction**, determining where the line extends when facing forks. Press **Find Surface** to separate the second surface.



Note that the `Select Points` list of the surface in the `Select Surf` list will be fixed. You can add new points based on the selected surface and press `Find Surface` to get a new surface. **Remember to** press `Clear` before adding points for a completely new surface. To delete the selected point, press `Delete`.

Switch between different surfaces by clicking them in the `Select Surf` list, and delete them by right-clicking and choosing `Delete Selected`.

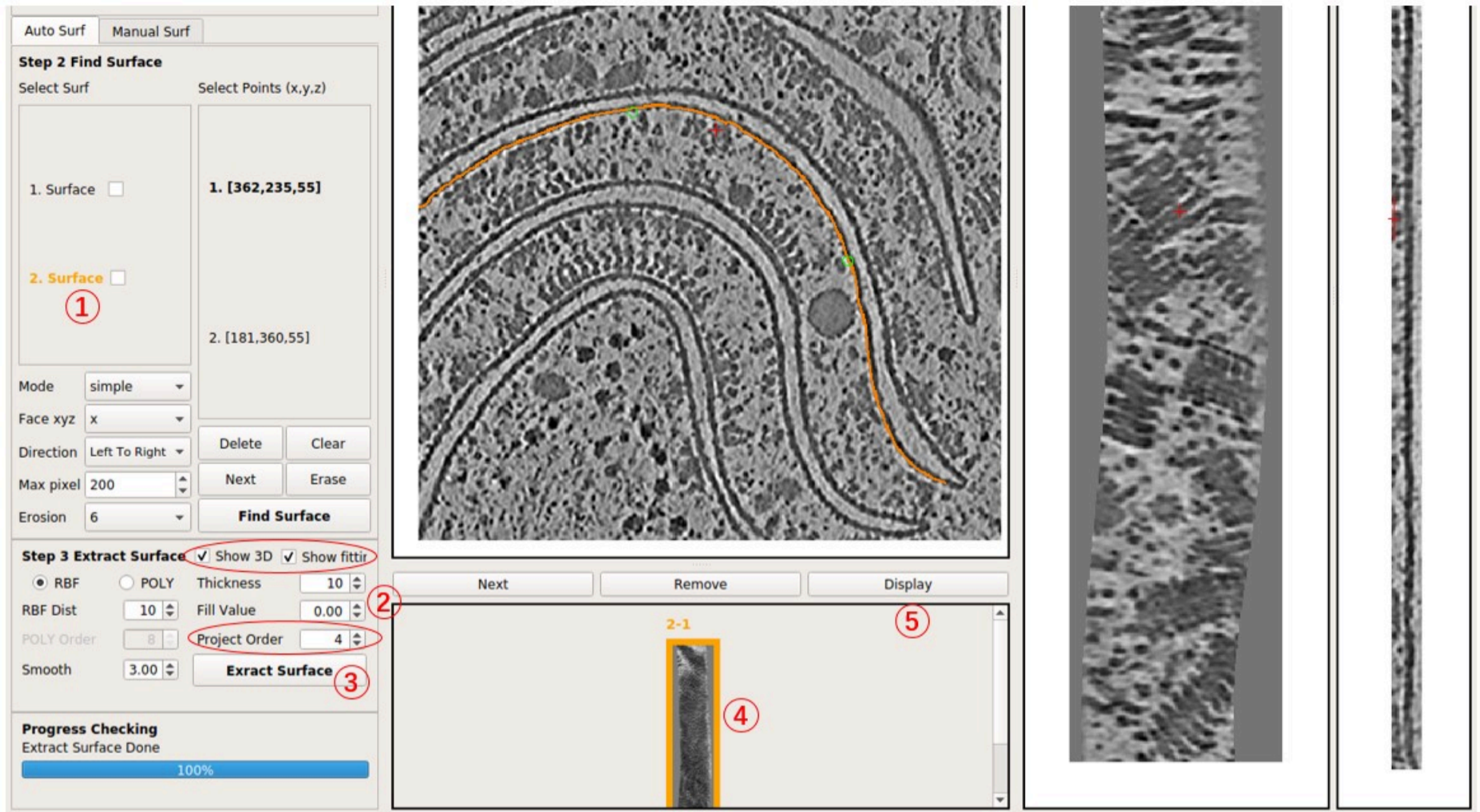
For a boundary with low quality, indicating many forks and discontinuities, it's advisable to decrease `Max pixel` and choose more points at different places on the membrane surface.

## Flatten a Surface

Select the second surface by clicking `2. Surface`. Change `Project Order` to `4`, indicating MPicker will project it onto a fourth-order polynomial cylinder. Increase `thick` for a thicker flattened tomogram.

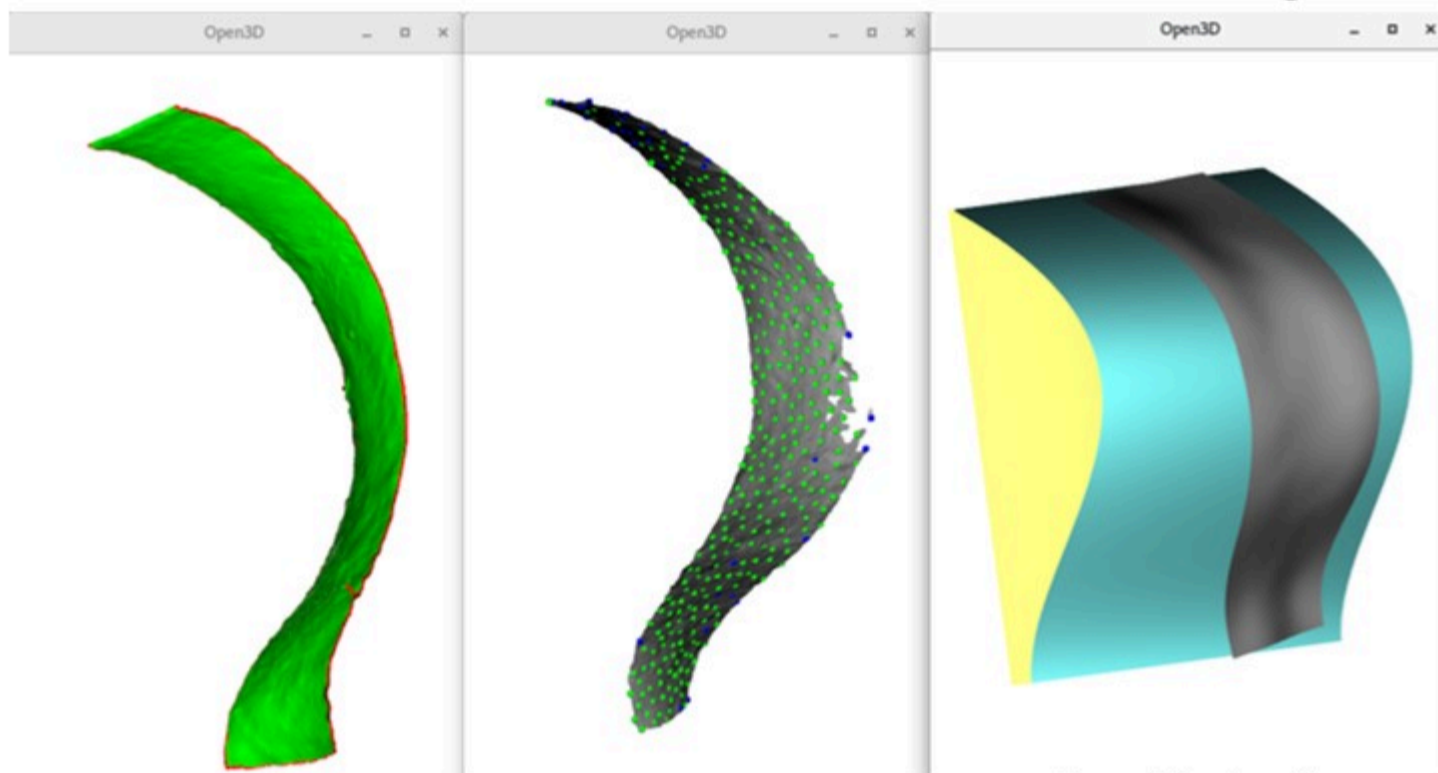
Press `Extract Surface` to flatten the membrane; this process takes about 5 seconds. You will obtain a flattened tomogram labeled `2-1`. Select it by clicking, then press `Display` to view the flattened tomogram on the right side.

Check `Show 3D` and `Show fitting` to observe the intermediate process in 3D. Note that viewing it remotely, for example, through an SSH connection, may limit visibility due to OpenGL reasons, but it won't affect the result.



Show 3d

Show fitting



separated surface  
red: removed outlier point

Selected points for RBF interpolation  
blue: removed outlier point

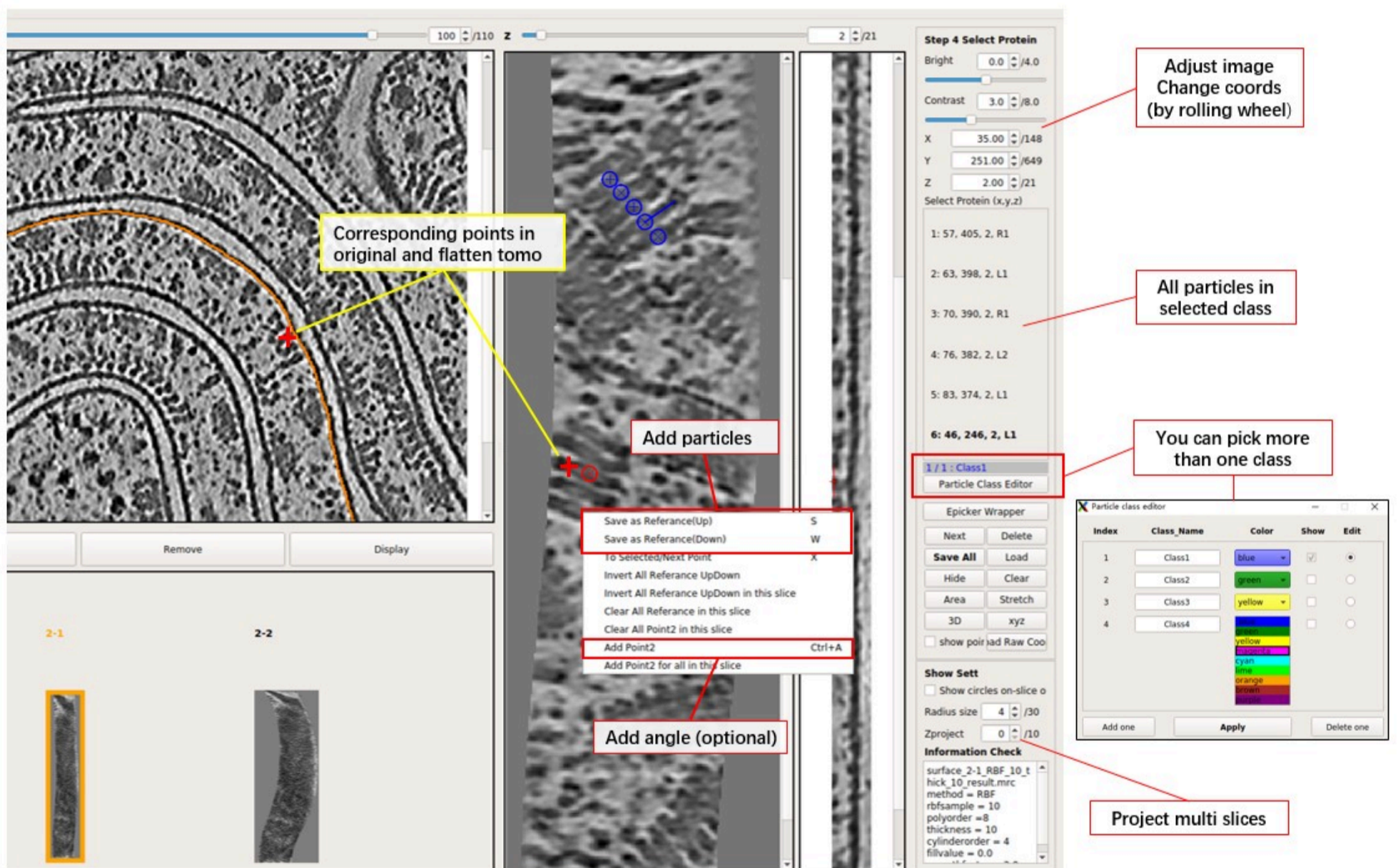
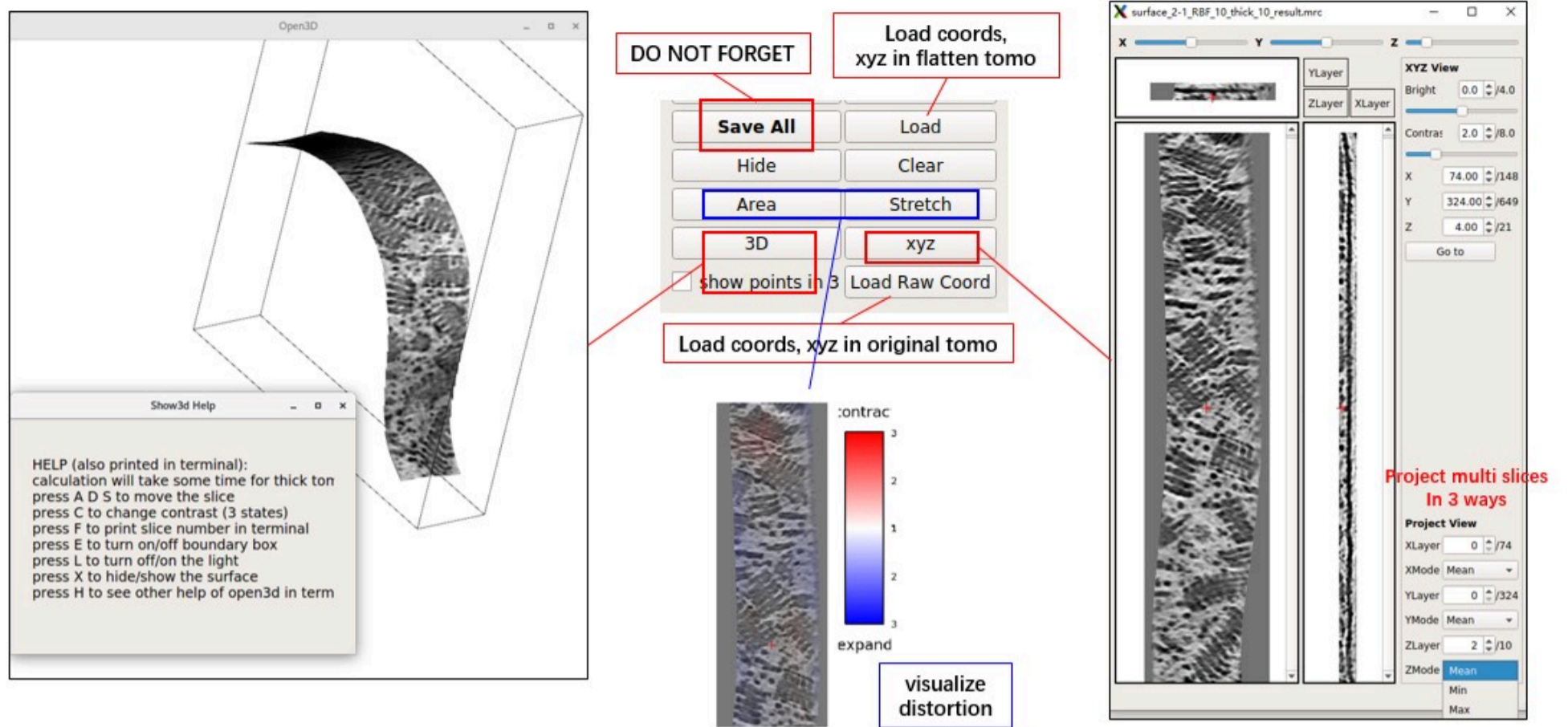
Gray: fitted surface  
cyan: fitted cylinder  
Yellow: fitted plane

You can modify parameters and press **Extract Surface** again to obtain another flattened tomogram of this surface, for example, **2-2**. Experiment by changing **Project Order** to **1** to observe the result of projecting it onto a plane rather than a cylinder. In general, increasing **Project Order** may solve issues if you find the result strange.

## • Check the Result

Examine the flattened tomogram on the right side. Adjust the image contrast using the Bright and Contrast spinbox. Move the red cross by left-clicking on the image, adjusting the spinbox of xyz coordinates, or pressing arrow keys. Note that the red cross on the left side (raw tomogram) will move correspondingly. These two crosses indicate the same location.

The main GUI will display XY and YZ slices. You can view both XY, YZ, and XZ slices and project multiple slices in a new window by pressing **xyz**. To assess the distortion caused by flattening, press **Area** or **Stretch**. To view XY slices in 3D space (without distortion), press **3D**.



## • Output Files of MPicker

MPicker should generate some files and folders as shown below. The most important ones are `emd13771_isonet.config`, `emd13771_isonet/*/*_result.mrc`, and `emd13771_isonet/*/*_InterpCoords.txt`. You can ignore others for now if you are not an advanced user.

```

--tutorial
-- emd13771_isonet.config
-- emd13771_isonet
-- my_boundary6.mrc
-- surface_1_emd13771_isonet
...
-- surface_2_emd13771_isonet
-- surface_2.config
-- surface_2_surf.mrc.npz
-- surface_2-1.config
-- surface_2-1_RBF_10_thick_10_convert_coord.npy
-- surface_2-1_RBF_10_thick_10_result.mrc
-- surface_2-1_RBF_10_thick_10_SelectPoints.txt
-- surface_2-1_RBF_InterpCoords.txt
...
-- manual_1_emd13771_isonet
-- manual_1.config
-- manual_1_surf.txt
-- manual_1-1.config
-- manual_1-1_RBF_10_thick_10_convert_coord.npy
-- manual_1-1_RBF_10_thick_10_result.mrc
-- manual_1-1_RBF_10_thick_10_SelectPoints.txt
-- manual_1-1_RBF_InterpCoords.txt
-- memseg
-- seg_raw_id0.mrc
-- seg_post_id0_thres_0.60_gauss_0.50_voxel_150.mrc
...

```

- `emd13771_isonet` is the folder that contains all results of MPicker.
- `emd13771_isonet.config` is used to reload the result. Continue working in MPicker next time with:

```
Mpicker_gui.py --config path_to/emd13771_isonet.config &
```

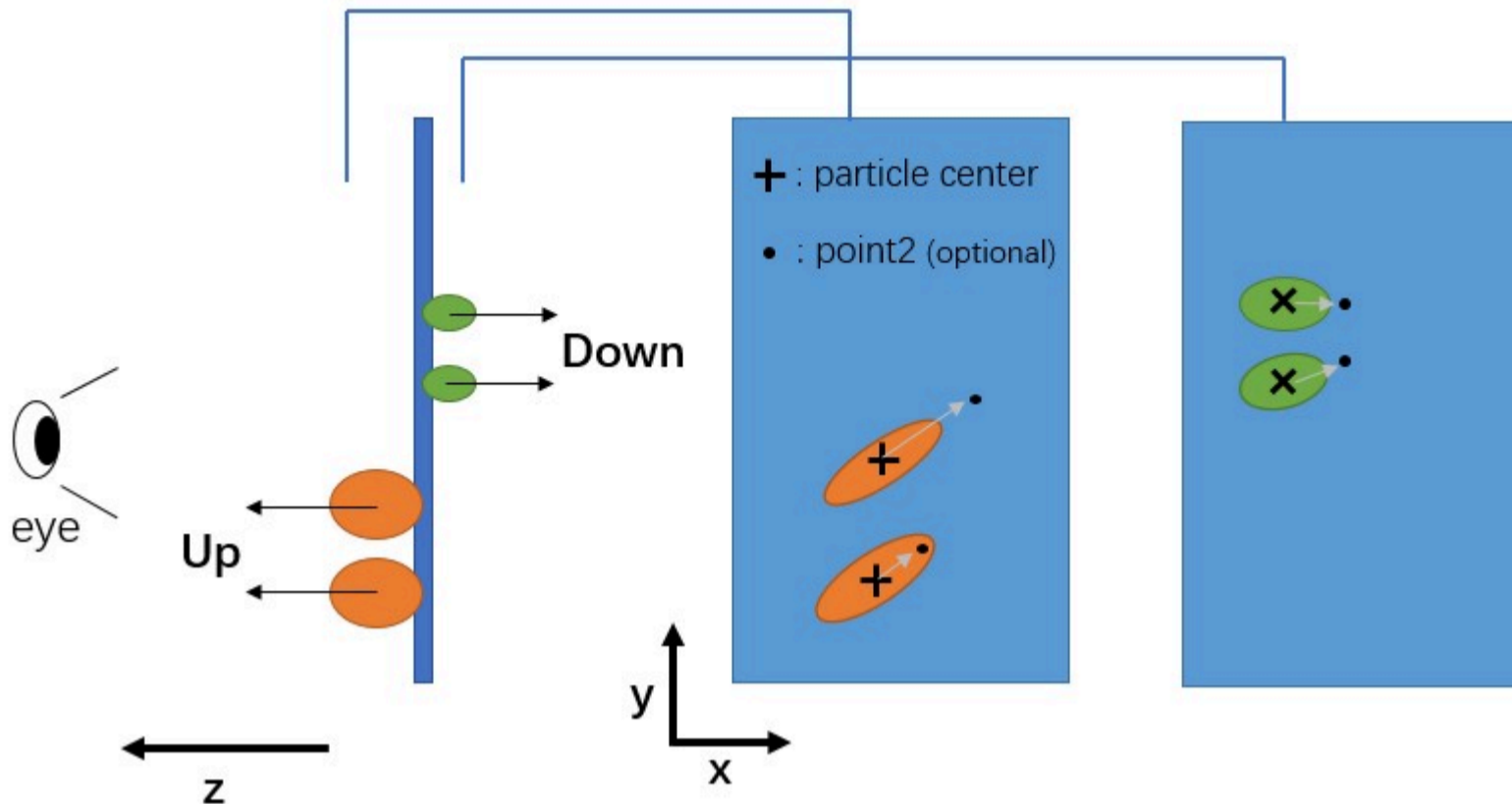
You can also load the config file by pressing `Load Config` after opening the GUI. The config file saves the absolute path of the input tomogram, membrane segmentation, and boundary file. If you relocate the result folder and config file, ensure that they remain in the same directory. Also, remember to update the file paths in the config file, including the path for `my_boundary6.mrc`.

- `my_boundaryxxx.mrc` is the boundary file calculated from segmentation. MPicker calculates it because a membrane is thick. By using only one of the two boundary surfaces of a membrane, MPicker can flatten the membrane with higher accuracy.
- Each separated surface will have its folder, such as `surface_2_emd13771_isonet`, which is the folder of **2.Surface**. The file `surface_2.config` records parameters MPicker used to separate it from the boundary. `surface_2_surf.mrc.npz` is the separated surface, and you can convert between .npz file and .mrc file by the script `Mpicker_convert_mrc.py`. MPicker converts .mrc to .npz to save space.
- Each separated surface can be flattened into more than one flattened tomogram, such as `2-1`, `2-2`. They are all saved in one folder. The `.config` file records the parameters used to flatten. The `_result.mrc` file is the flattened tomogram itself. The `_convert_coord.npy` file records the relationship of coordinates between the flattened tomogram and the original tomogram. If  $x, y, z$  in the flattened tomogram correspond to  $R_x, R_y, R_z$  in the original tomogram, and they start from 0, then you can get  $R_x, R_y, R_z$  by `Rz, Ry, Rx = numpy.load("xxx.npy")[:, z, y, x]`. The `_InterpCoords.txt` file records the points MPicker finally used for RBF interpolation, and you can let it use more points by decreasing `RBF Dist` and vice versa. The `_SelectPoints.txt` file records all information of particles you picked in the flattened tomogram. See **Particle Picking by Hand** for details.
- If you separated surfaces by manually labeled points, see **Flatten Membrane Without Segmentation** for details, then each surface will also have its folder, such as `manual_1_emd13771_isonet`. It doesn't have a `.mrc.npz` file because it is not from segmentation, but it has a file to record manually labeled points such as `manual_1_surf.txt`. Other files are the same as that in the surface separated from segmentation.
- If you do membrane segmentation in MPicker, see **Generate Membrane Segmentation** for details, then you will get a folder `memseg`, which saves all results of segmentation. Notice you cannot use `seg_raw_id0.mrc` as the `Mask` because it is not a binary segmentation (with only 0 or 1 as its value).

# Particle Picking by Hand

Now you can manually pick particles in the flattened tomogram. Position the red cross by left-clicking and press **S** to save the position (or right-click and select **Save as Reference**). Note that there is a distinction between Up (**S**) and Down (**w**), but you can ignore it (just press **S**) if particle orientation is not a concern. Refer to the image below for a more intuitive understanding.

**Remember** to press **Save All** before quitting or displaying another flattened tomogram.



MPicker will record the corresponding membrane normal vector for each picked particle, but the normal vector can have two options, Up or Down, depending on which side of the membrane the particle is on.

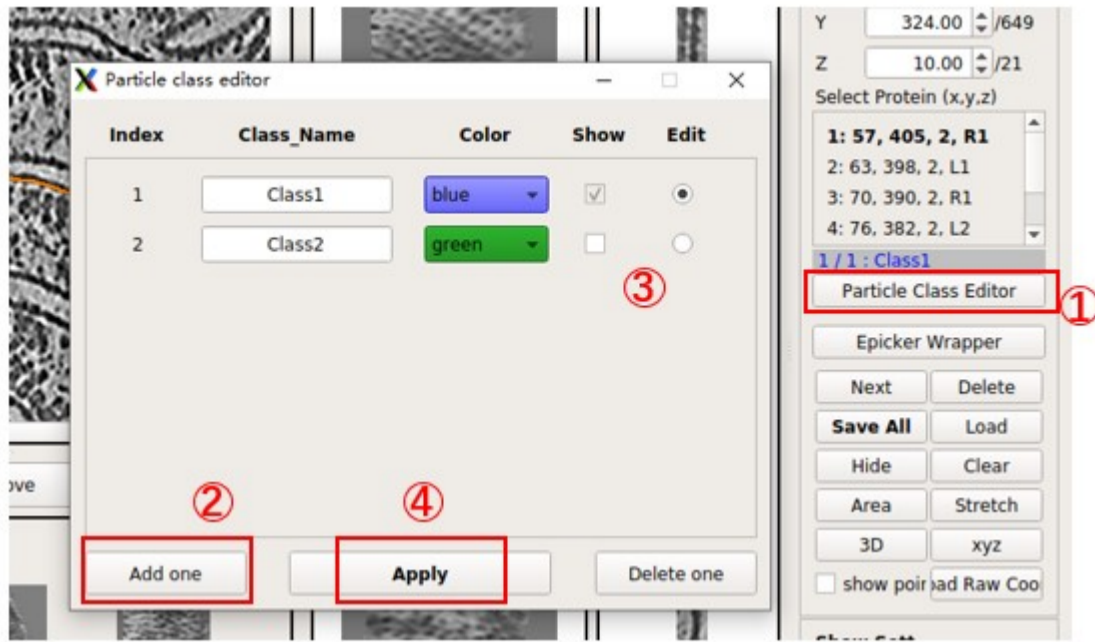
You can also add a position as "point2" for a selected particle, enabling you to completely determine the orientation of the particle (grey arrow as the x-axis and the norm vector as the z-axis). To add point2 for a particle, move to the same xy slice as the selected particle, position the red cross, and press **Ctrl+A** (or right-click and press **Add Point2**).

To select a picked particle, you can double-click on a particle in the xy slice, and its circle will turn red. Ensure that the center of the circle is on this xy slice (cross is displayed). You can also select a particle by clicking on the ones listed on the right side.

You can also pick particles in the flattened tomogram (a file ending with **\_result.mrc**) using other software if you prefer. To load the result into MPicker, simply load the coordinate file (a txt file with 3 columns as x y z, starting from 1) by pressing **Load**. If you already picked particles in the original tomogram, you can load the coordinates by pressing **Load Raw Coord** and check them in the flattened tomogram. Particles outside of the flattened tomogram will be ignored. You can also perform coordinate conversion between the original and flattened tomogram using the script **Mpicker\_convert\_coord.py**.

You can also pick more than one class of particle together. Press **Particle Class Editor**, and then press **Add one** to add a new class. Select which class to edit (pick particles) and which classes to display. Then press **Apply** to use it. You can change the color and name for each class, but MPicker will only save the **index** into **\_SelectPoints.txt**. Note that pressing **Delete one** will not delete the data of particles; it only controls the display. For example, if you picked some particles in Class2 and then delete it, the gui will display **1/2: Class1, 2** means the particle in the saved data with the highest index is 2. You can see them again if you add class2 back.





The information of picked particles for each flattened tomogram is saved in files ending with `_SelectPoints.txt`. For surface 2-1, it is `tutorial/emd13771_isonet/surface_2_emd13771_isonet/surface_2-1_RBF_10_thick_10_SelectPoints.txt`. Below is what it looks like.

#	coords in flatten tomo						coords in original tomo						normal vector			class id	# xyz start from 1
	x	y	z	x2	y2	z2	Raw_x	Raw_y	Raw_z	Raw_x2	Raw_y2	Raw_z2	UpDown	nx	ny	nz	
57	405	2	-1	-1	-1	228.9	347.9	78.6	-1.0	-1.0	-1.0	1	0.303	0.944	0.128	1	
63	398	2	-1	-1	-1	235.9	346.7	72.3	-1.0	-1.0	-1.0	-1	0.322	0.926	0.196	1	
70	390	2	-1	-1	-1	244.0	345.5	65.2	-1.0	-1.0	-1.0	1	0.345	0.901	0.263	1	
76	382	2	92	393	2	252.1	344.1	59.4	246.3	352.4	43.1	-1	0.375	0.876	0.302	1	
83	374	2	-1	-1	-1	260.2	342.8	52.7	-1.0	-1.0	-1.0	-1	0.423	0.848	0.319	1	
46	246	2	-1	-1	-1	336.8	254.5	88.4	-1.0	-1.0	-1.0	-1	0.864	0.466	0.188	1	

coords of point2 (-1 if no point2)

You can merge all `_SelectPoints.txt` of one tomogram and convert them into Euler angles using the script `Mpicker_particles.py`. For example:

```
Mpicker_particles.py --input emd13771_isonet --out_mer merge.txt --out_ang angle.txt
```

It will merge all `_SelectPoints.txt` files in the result folder `emd13771_isonet` into `merge.txt` and then convert it to `angle.txt`. Below is what `angle.txt` looks like. `X Y Z` are the coordinates of particles in the original tomogram, and the definition of `rot tilt psi` is the same as `rlnAngleRot rlnAngleTilt rlnAnglePsi` in Relion.

#	coords in original tomo			Euler angles			class id	# xyz start from 1
	X	Y	Z	rot	tilt	psi	class	
228.9	347.9	78.6	118.0	82.6	107.8	1		
235.9	346.7	72.3	31.4	101.3	-70.8	1		
244.0	345.5	65.2	166.8	74.8	111.0	1		
252.1	344.1	59.4	-153.4	107.6	-66.8	1		
260.2	342.8	52.7	104.3	108.6	-63.5	1		
336.8	254.5	88.4	251.7	100.8	-28.3	1		

**Note: Normal vectors can only decide tilt and psi**, so the `rot` of particles without point2 will be filled with a random angle. You can also set the `rot` of all particles without point2 to `0` by adding `--fill_rot 0`. Type `Mpicker_particles.py -h` for details. Note that `UpDown` can be `1` (Up) or `-1` (Down), so the real normal vectors should be `UpDown * (nx, ny, nz)`.

## Generate Membrane Segmentation

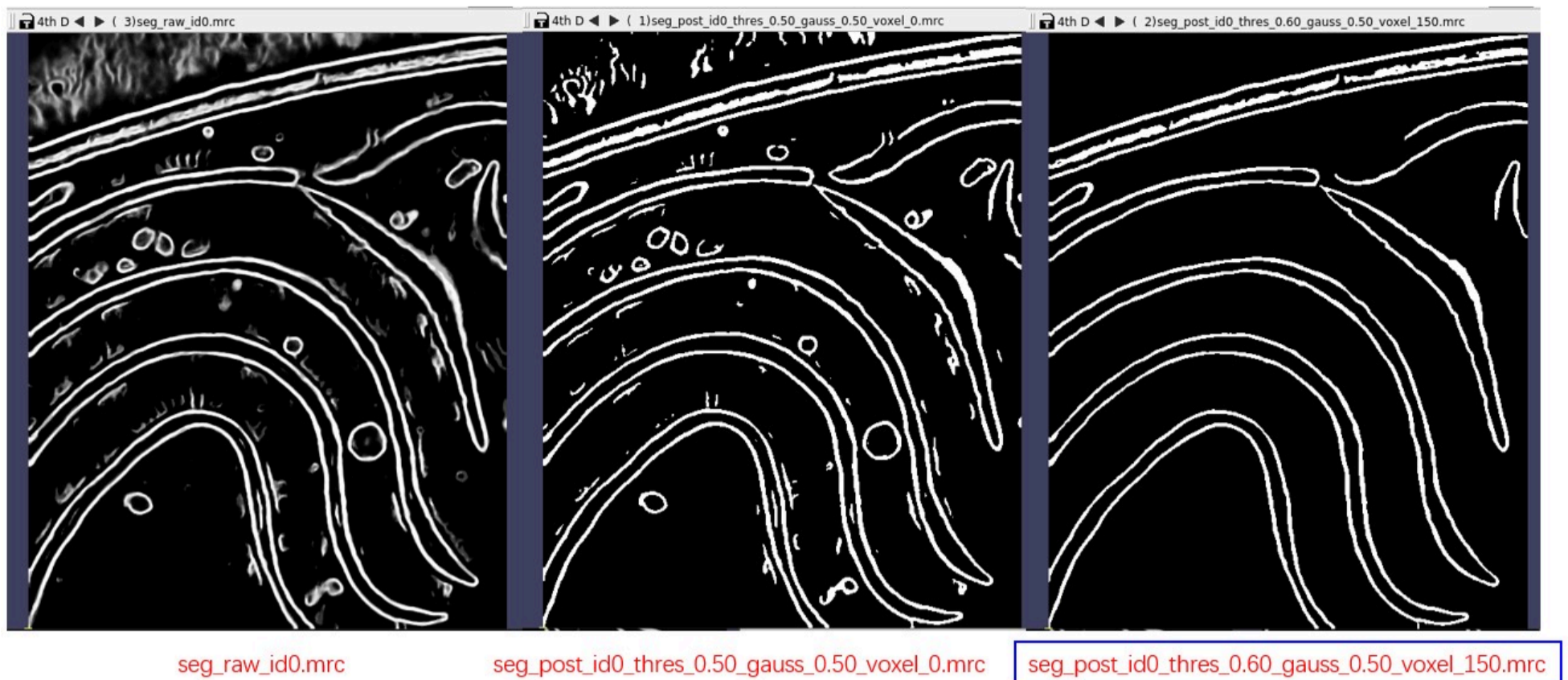
You can skip this step if you already have a segmentation. To run membrane segmentation, make sure the computer has at least one GPU, and you install the full version of MPicker (see installation instructions for details).

First, press `Mask` and choose `Get Mask`, and MPicker will open a new window.

Next, press `Run` in the left half of the window. (You can change GPU id to `0,1` and adjust batch size and threads to `4` if you have 2 GPUs.)



file `seg_post_id0_thres_0.60_gauss_0.50_voxel_150.mrc` , which should be similar to the segmentation contained in the tutorial folder.

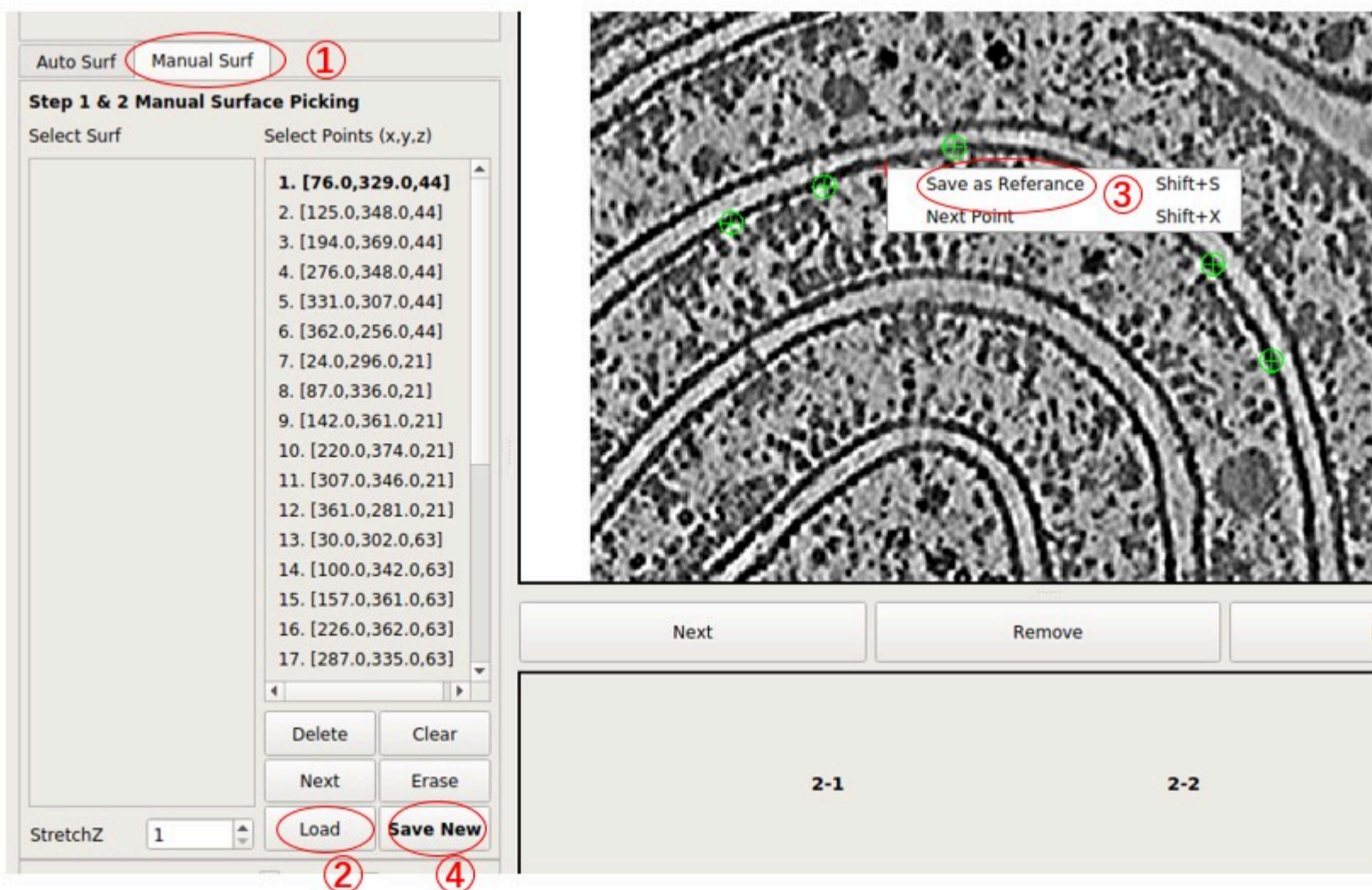


Now, you can close the segmentation window and return to the main window.

## Flatten Membrane Without Segmentation

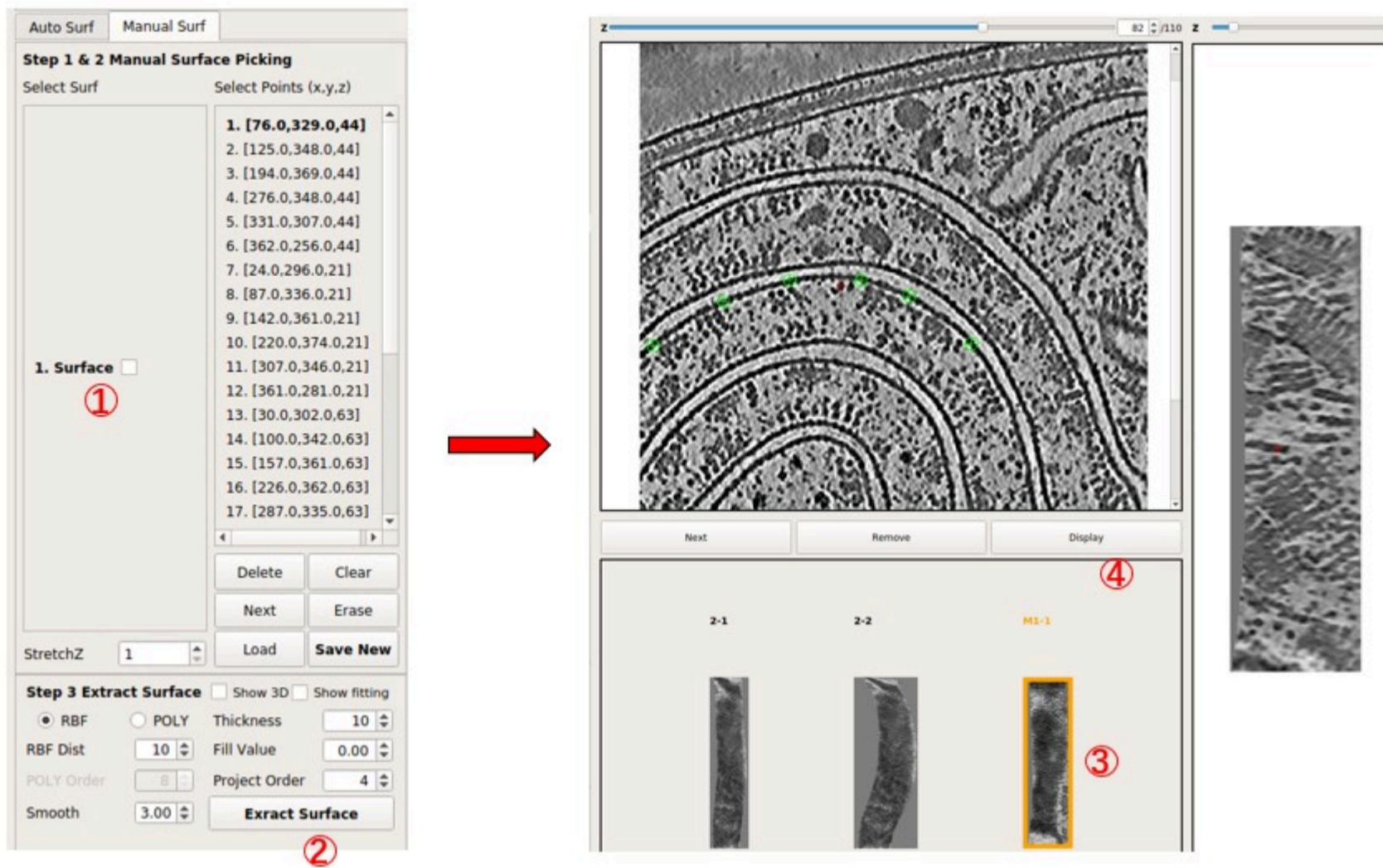
In case you don't have a suitable segmentation, you can skip **Load Membrane Segmentation** and **Separate a Surface** by labeling membranes by hand.

Select the tab `Manual Surf` instead of `Auto surf` , then press `Load` to load the coordinate file `manual_surf.txt` , which contains 29 manually labeled membrane positions. You can also add labels yourself. Similar to what we did in **Separate a Surface**, you can move the red cross to a point on the membrane by left-clicking it, then press `Shift + S` or right-click and choose `Save as Reference` to save it. Then, it will be added to the `Select Points` list. The added points are indicated by green circles. Press `Save New` to save these points as a new surface.



Next, press `Extract Surface` to flatten the membrane, and you will get a flattened tomogram called `M1-1` . Select it by clicking it. Then, press `Display` to view the flattened tomogram on the right side. You can see that the membrane is not as flattened as in surface `2-1` because the number of points is much less than what we used in surface `2-1` . However, it's sufficient to visualize large membrane

proteins. You can add more points to improve the result.



In general, you can decrease **RBF Dist** to **1** in this mode to preserve all points you added. It decides the minimal distance (in pixels) between points we used at last. Also, **Project Order** should not be high if the number of points is small.

To see which points were used for surface **2-1** (separated from segmentation), press **Clear** to clear current points, and then press **Load** to load file `emd13771_isonet/surface_2_emd13771_isonet/surface_2-1_RBF_InterpCoords.txt`, which should contain about 305 points.