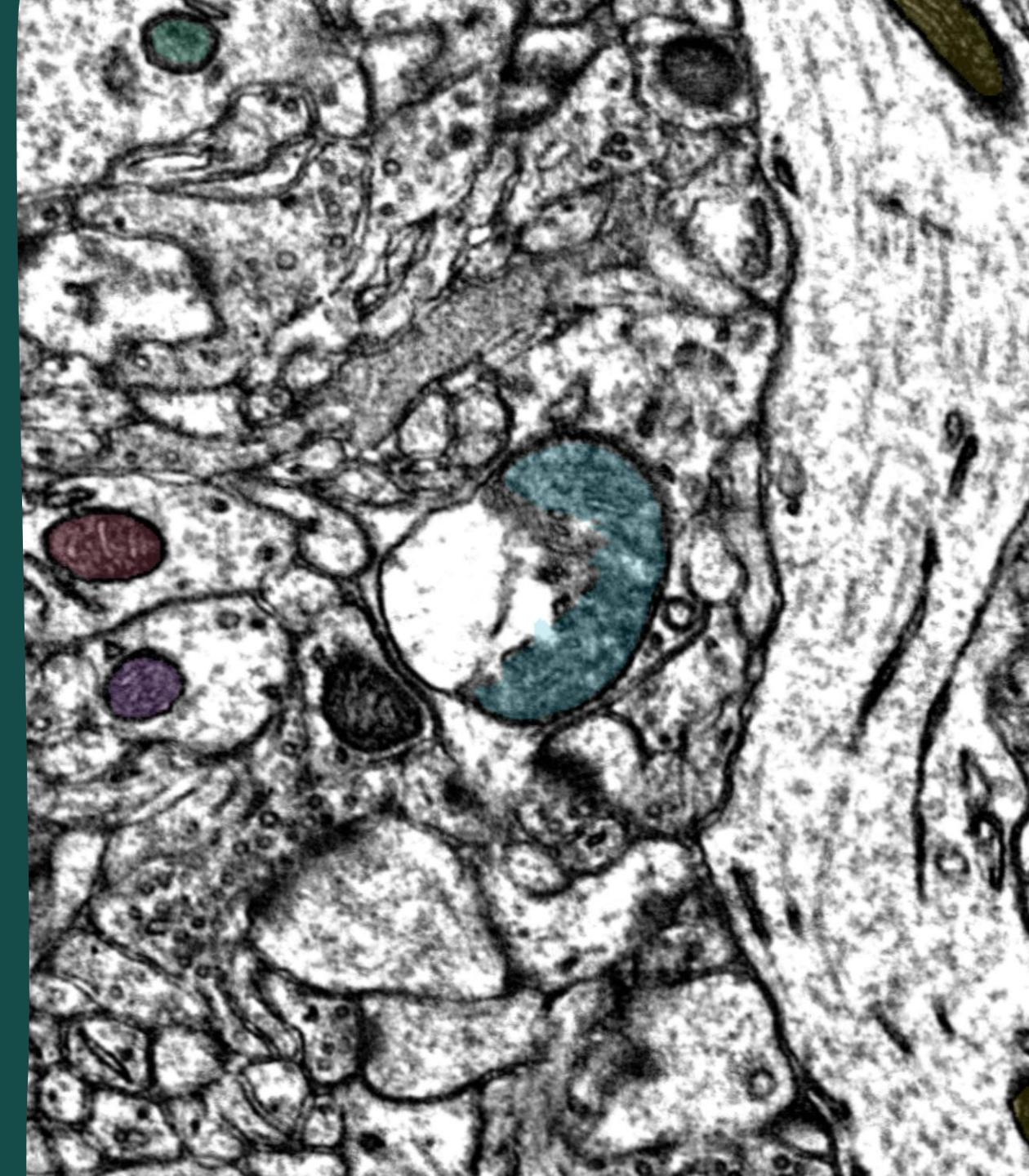
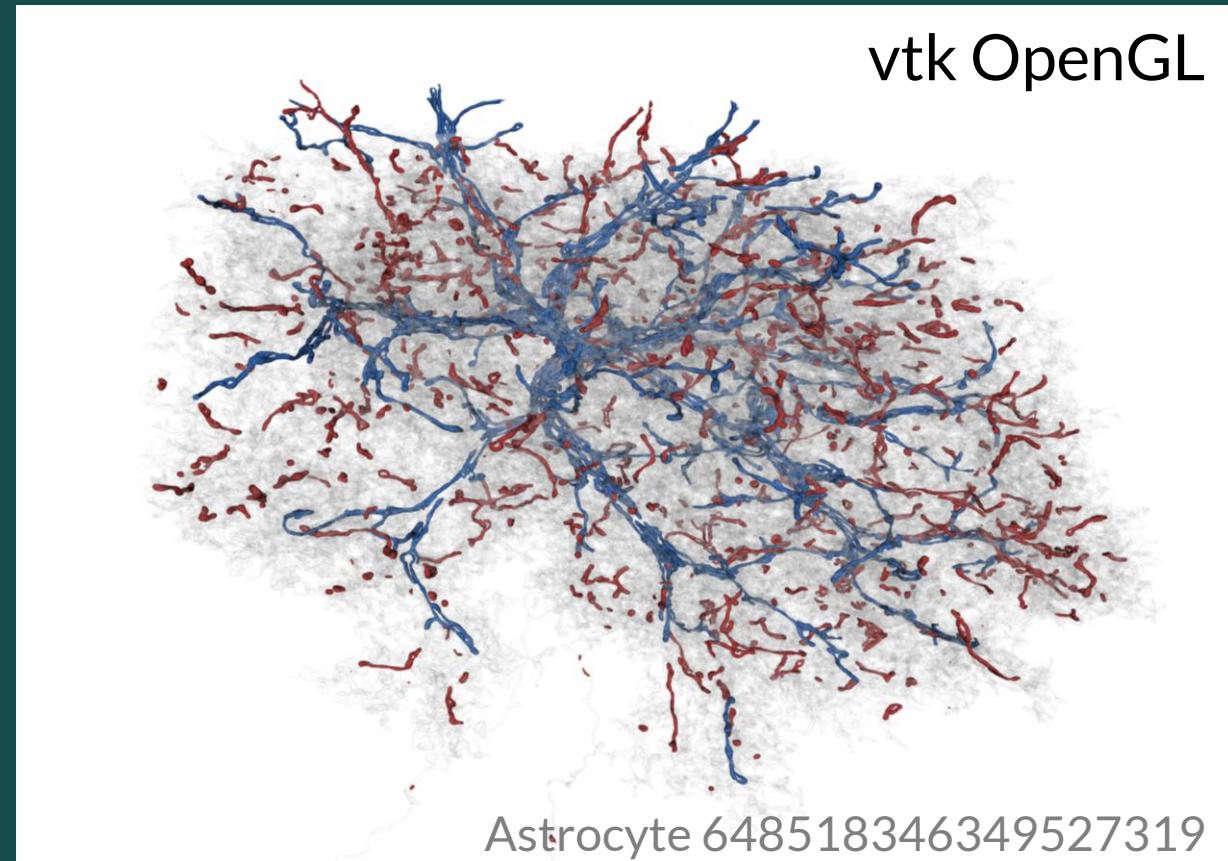
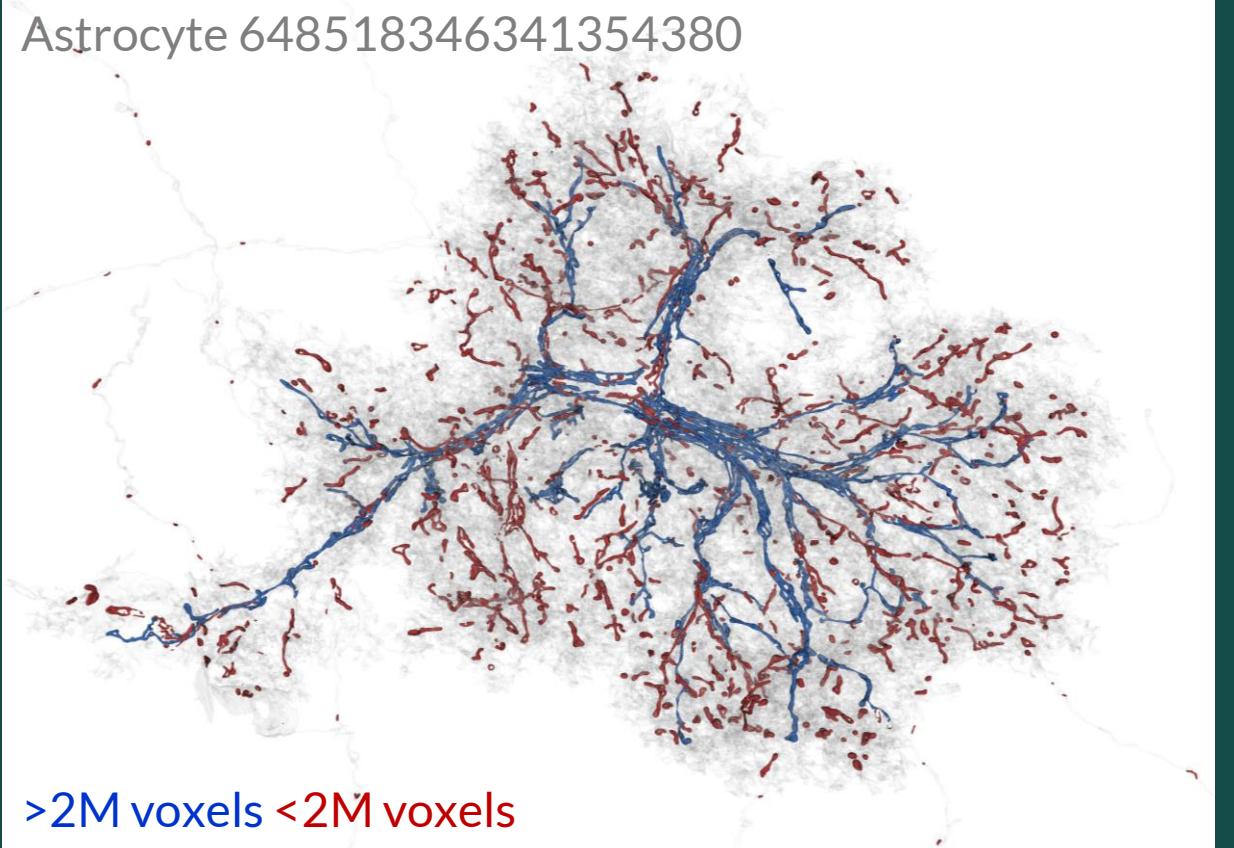


MITOCHONDRIA INCLUSIONS IN ASTROCYTES OF THE LAYER 2/3 VOLUME



Mitochondria in astrocytes are dense and extensively branched



View 3D mitochondria
visualizations of all 44 astrocytes
in the Layer 2/3 volume

[https://github.com/shandran/layer23-
volume/tree/main/astrocyte_mitochondria](https://github.com/shandran/layer23-volume/tree/main/astrocyte_mitochondria)

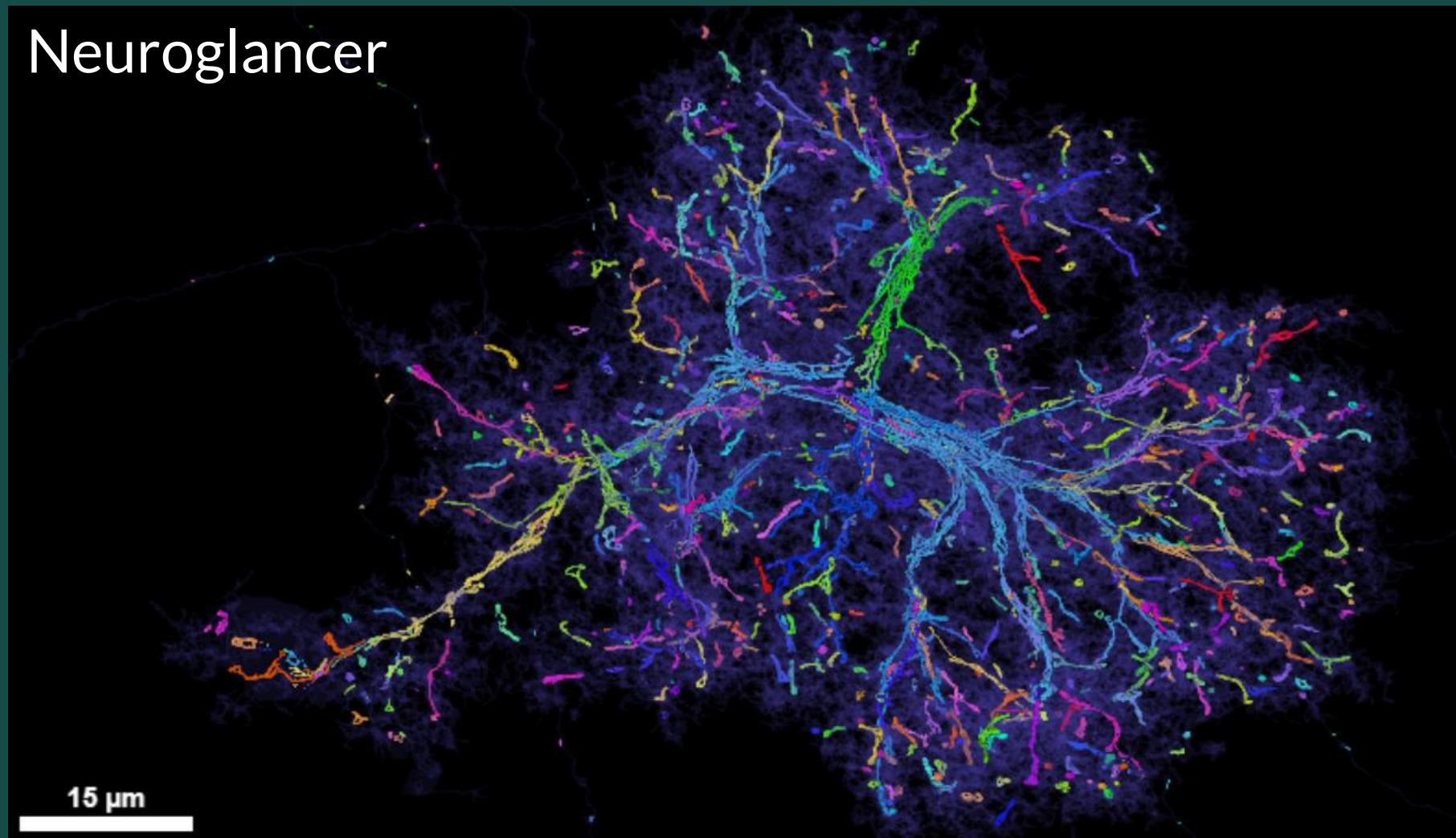
Mitochondrial inclusions are relatively easy to find in Neuroglancer

Reveal all the mitochondria for an astrocyte in Neuroglancer using the [neuroglancer_link_generator_all_mitochondria.ipynb](#) notebook and then search manually for broken or fragmented mitochondria

Inclusion examples

Generate EM image stacks and montages using the
[EM_image_generator_master_astrocyte_inclusions.ipynb](#) notebook
and create 3D visualizations using the
[vtk_astrocyte_mitochondria_visualizer_using_threshold_with_neuroglancer_all_mitos.ipynb](#) notebook

Astrocyte 648518346341354380

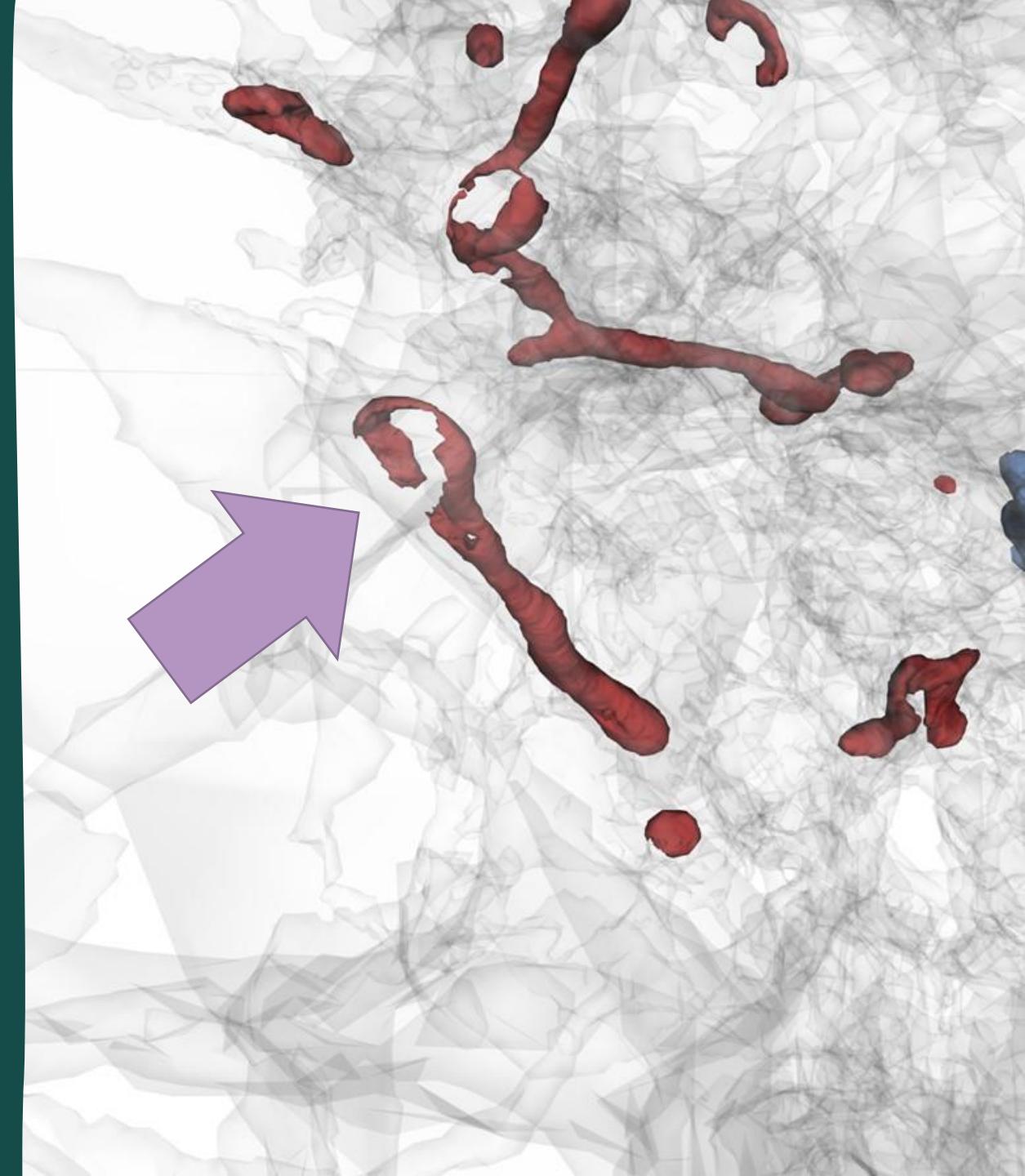


Inclusion example 1

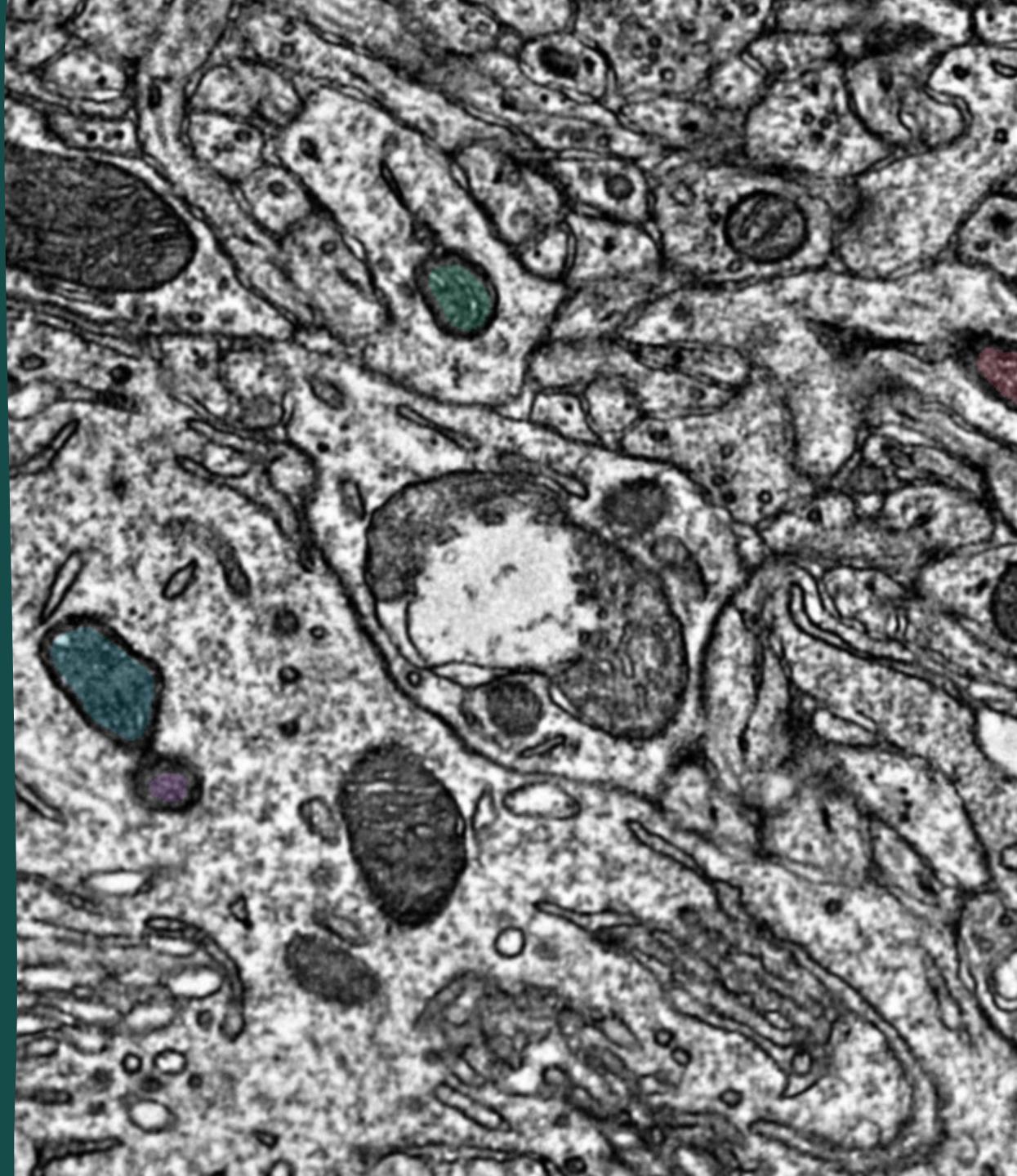
Cellid 648518346341354380

Mitoid 2643657

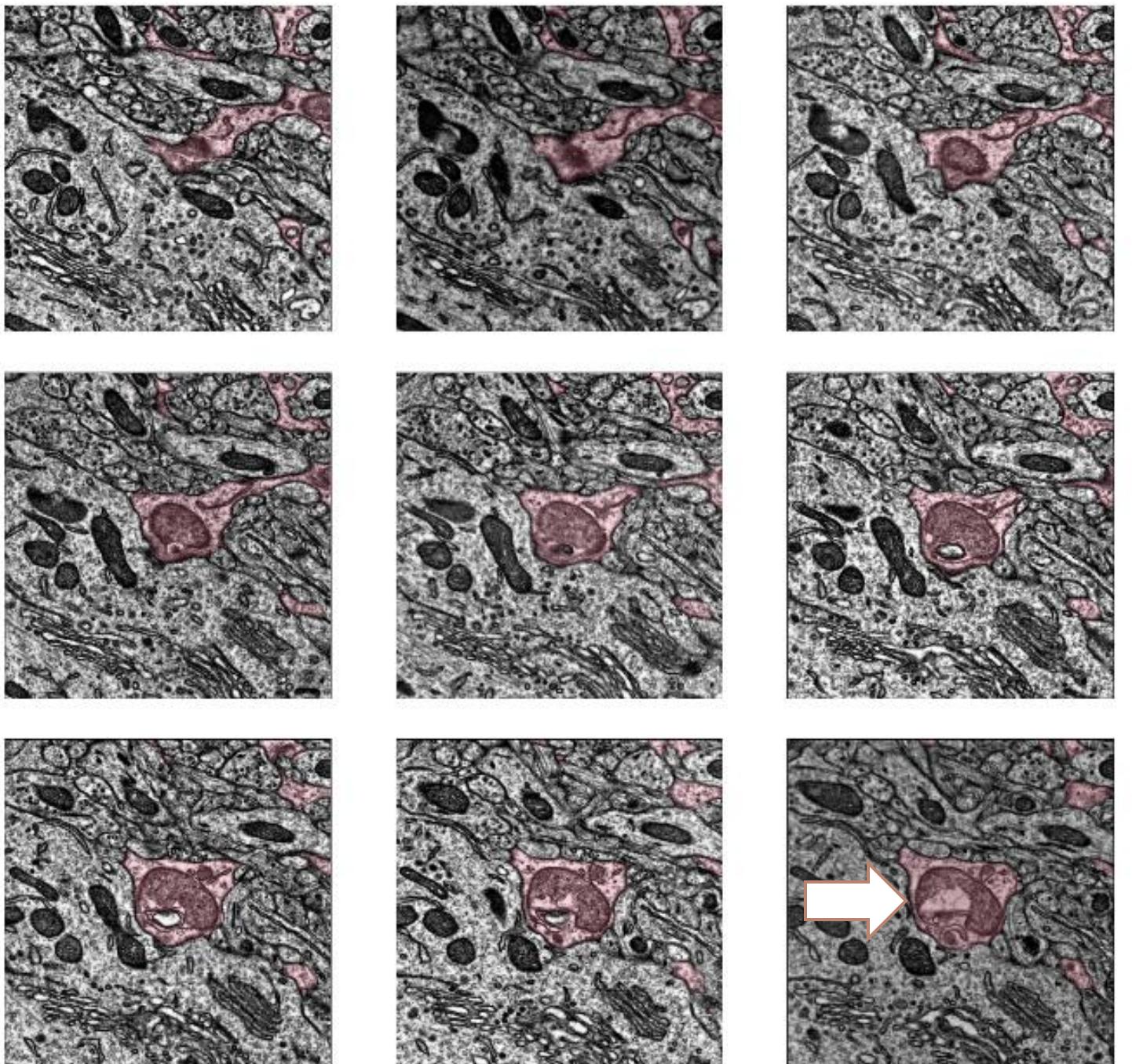
Mito 2643657



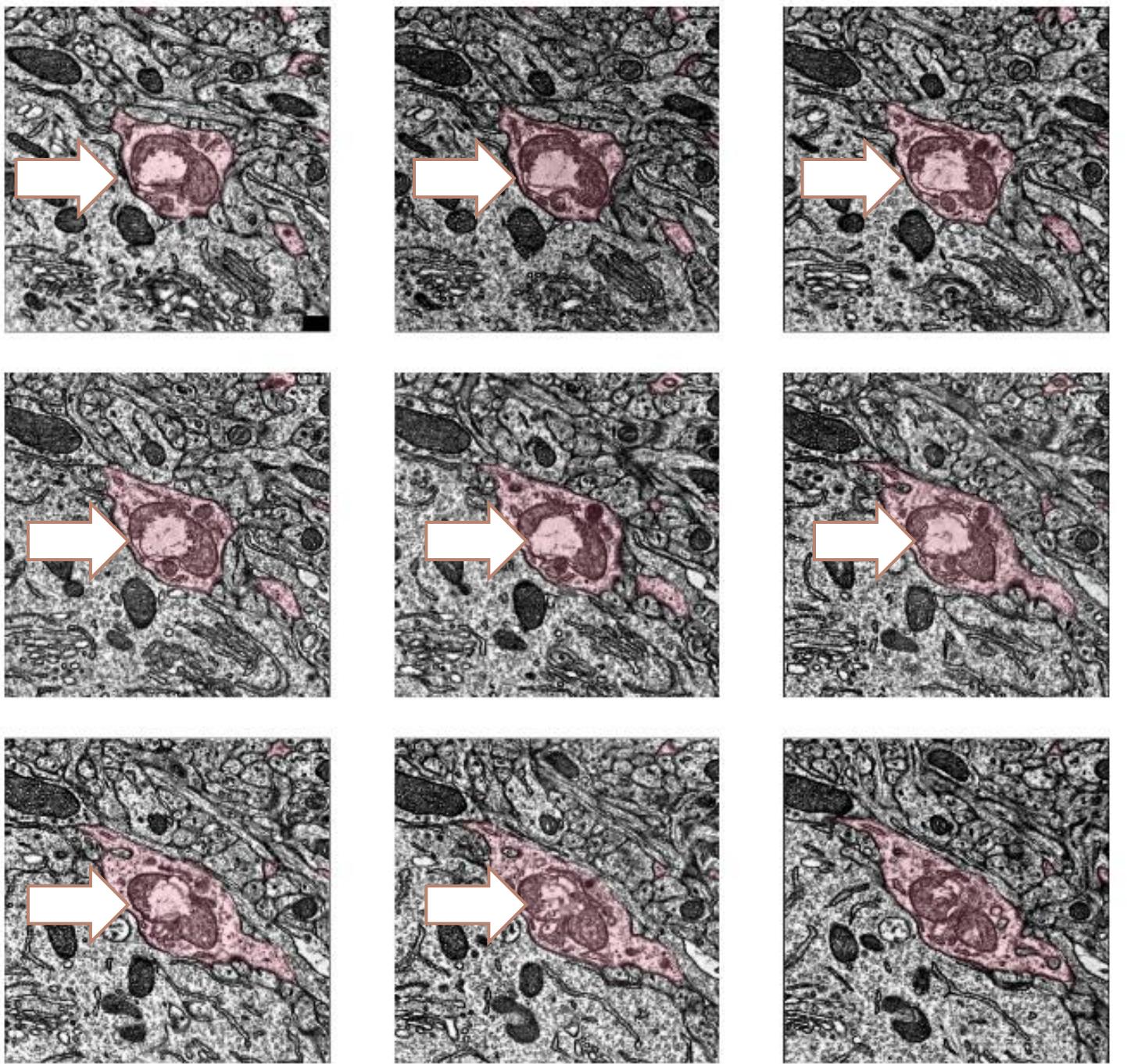
Mito 2643657



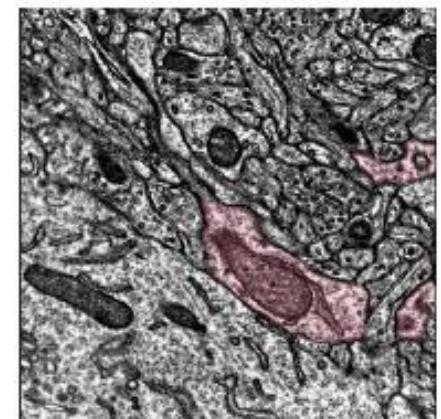
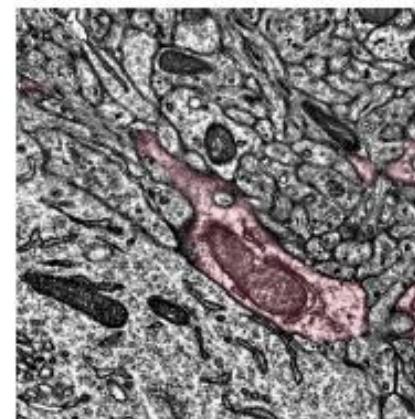
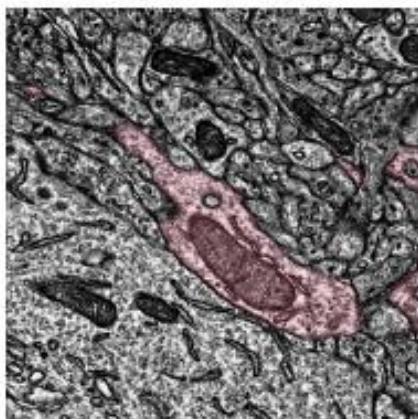
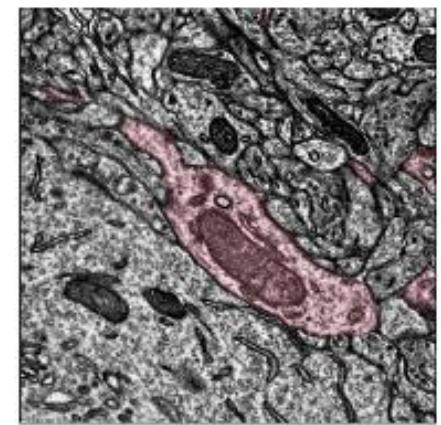
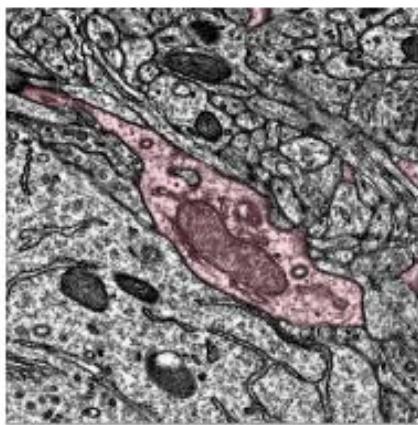
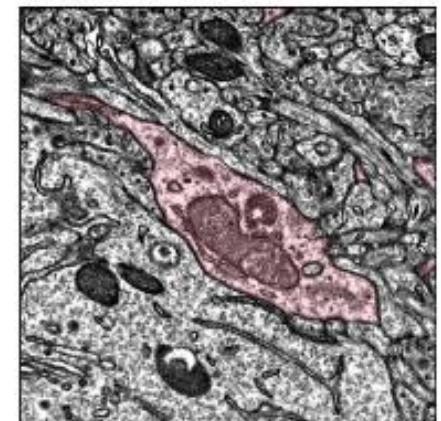
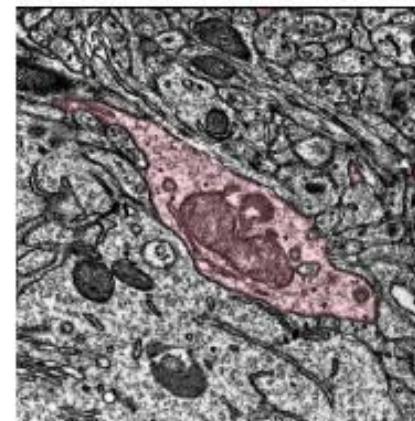
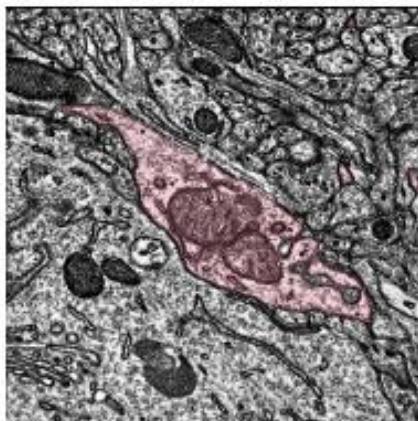
Mito 2643657



Mito 2643657



Mito 2643657



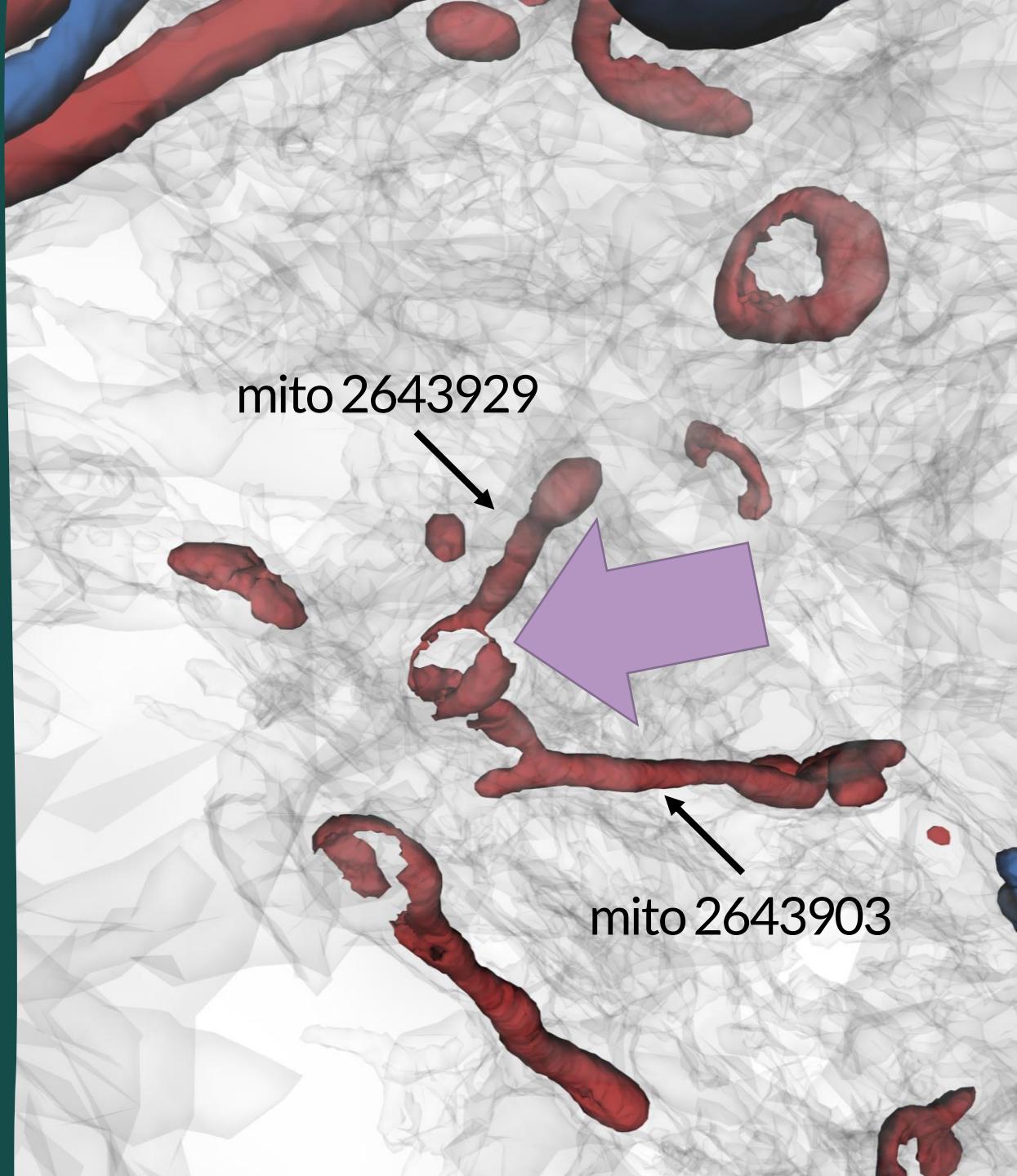
Inclusion example 2

Cellid 648518346341354380

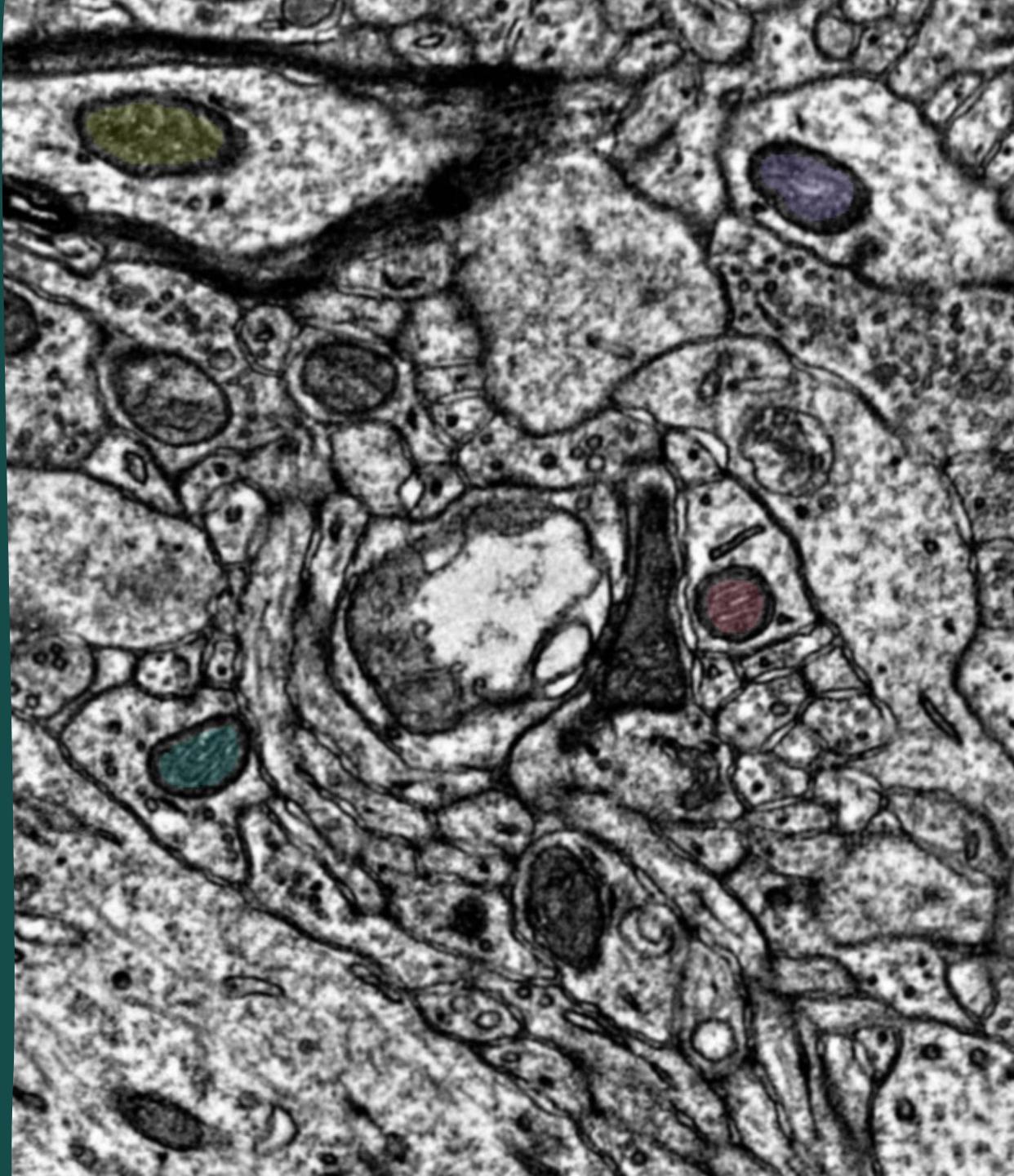
Mitoid 2643929 and 2643903

Mito 2643929

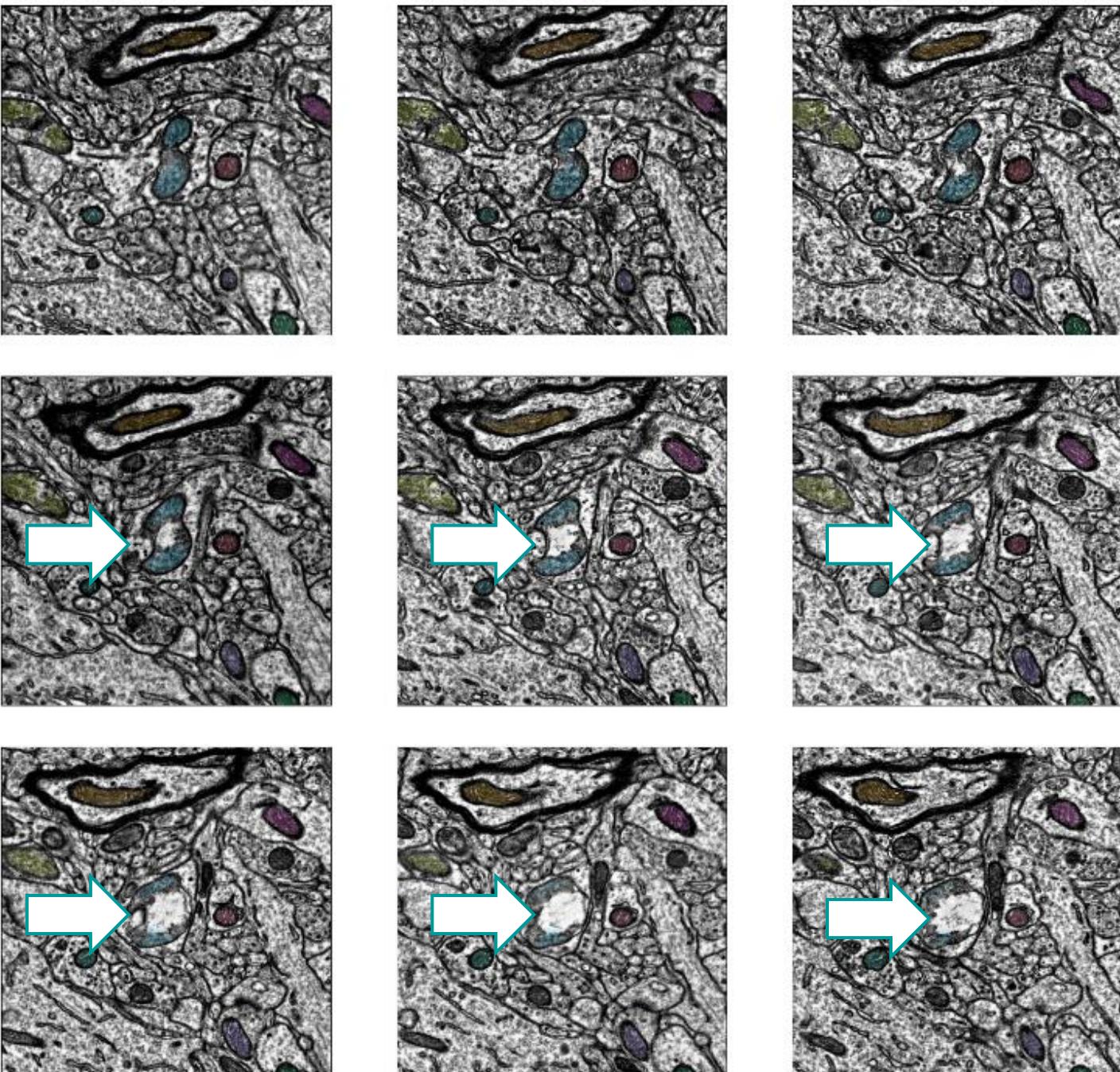
continuous with mito 2643903



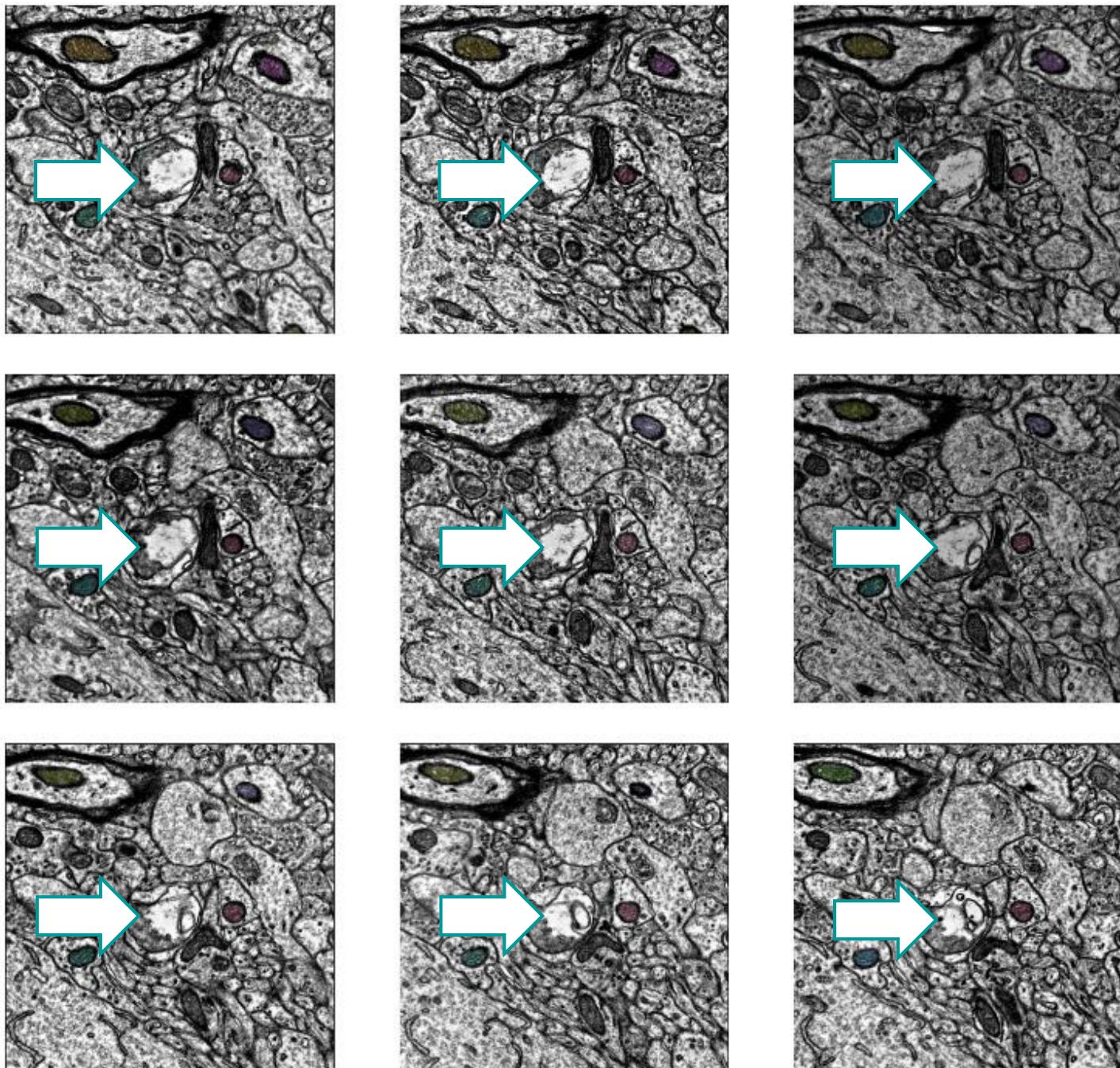
Mito 2643929

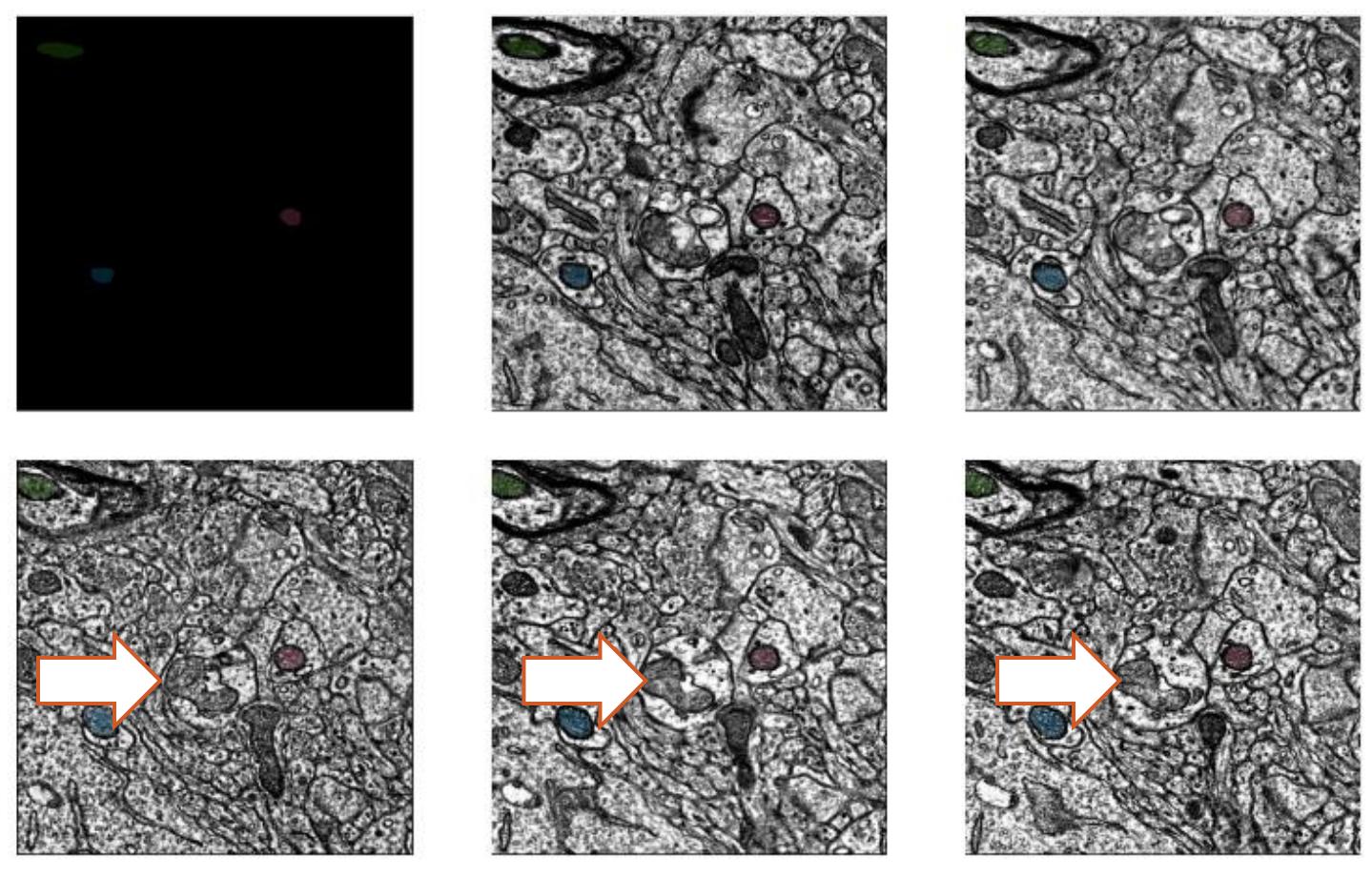


Mito 2643929



Mito 2643929





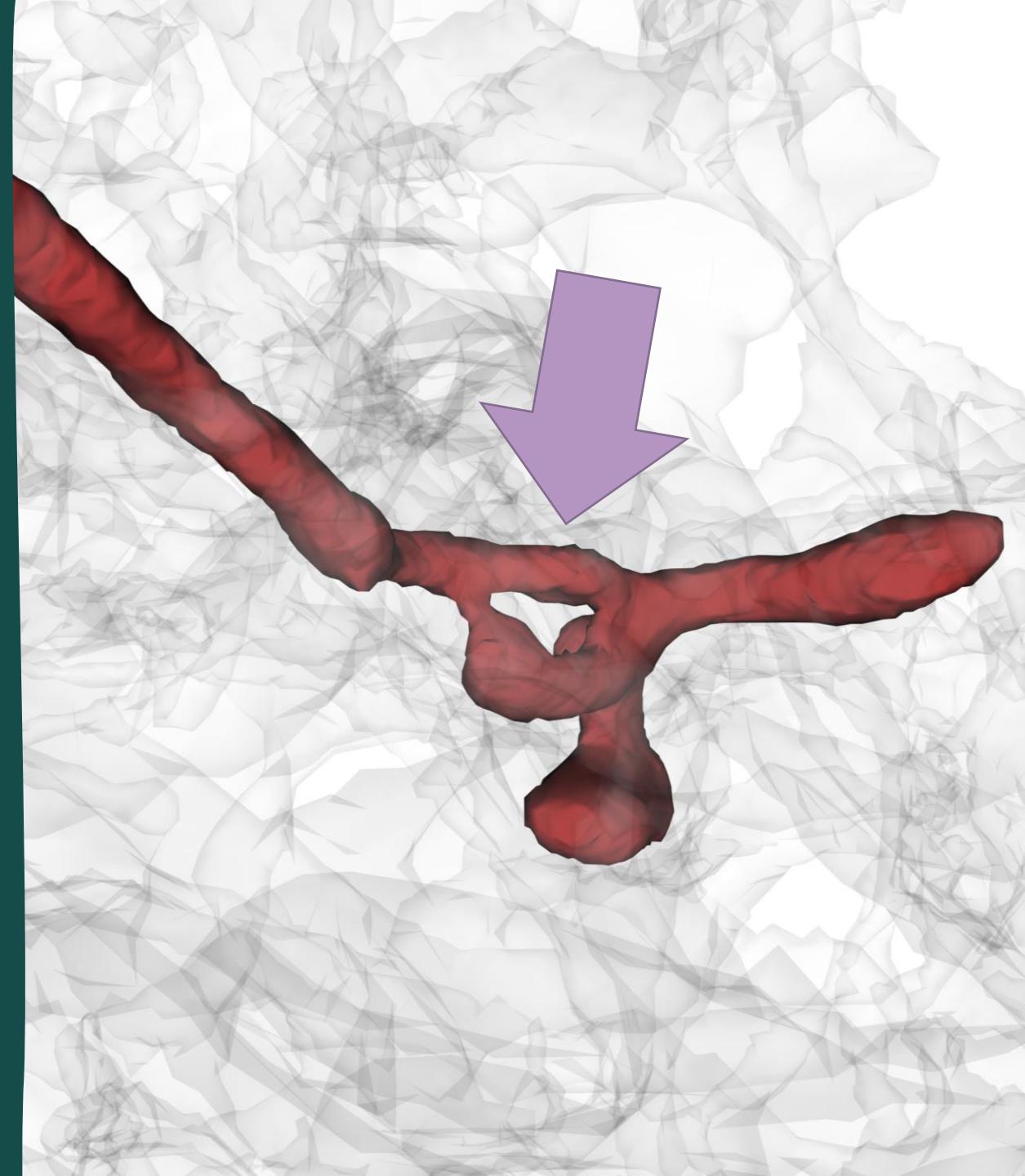
Mito 2643929 is continuous with Mito 2643903
(the inclusion interrupted the segmentation)

Inclusion example 3

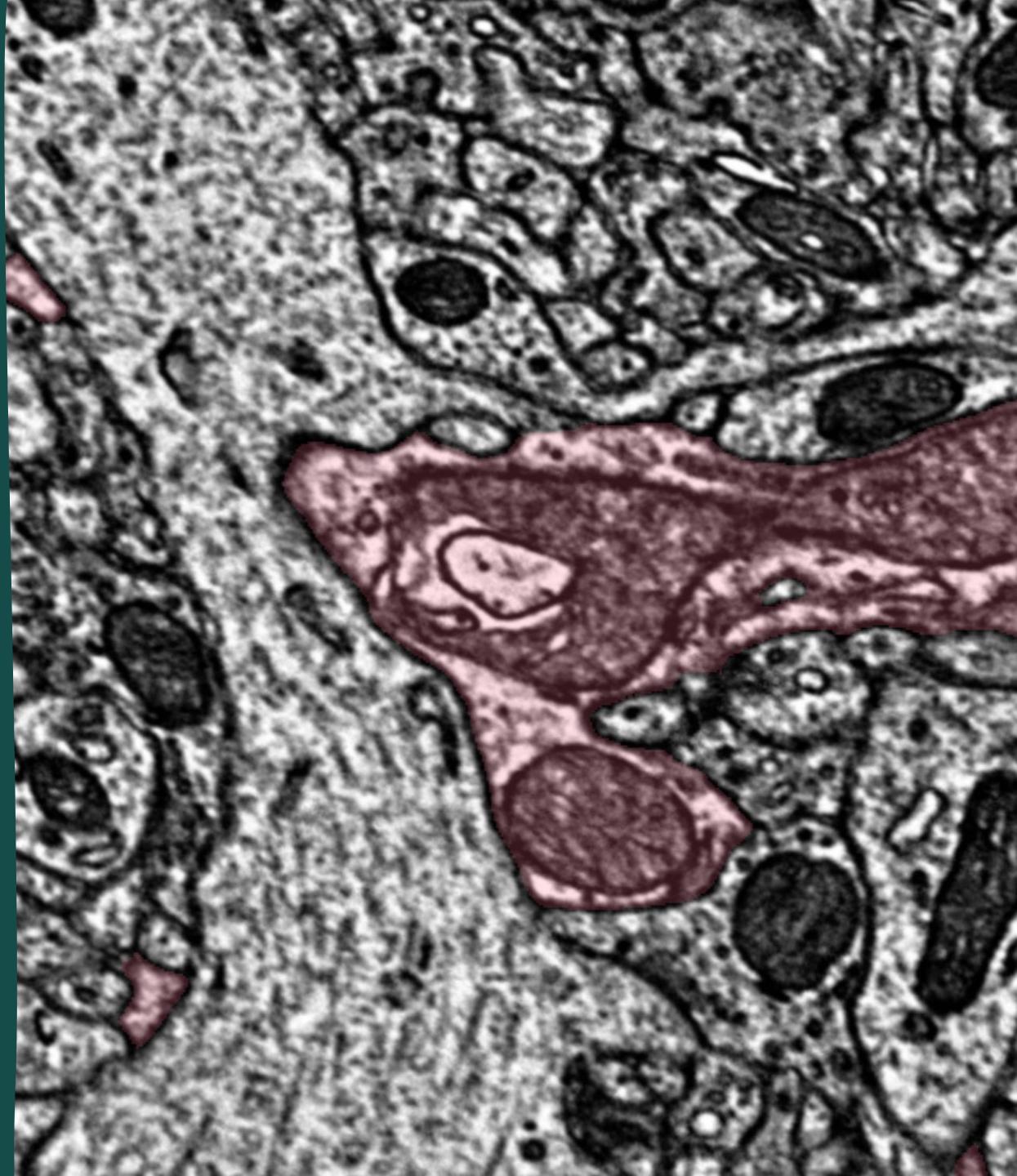
Cellid 648518346341354380

Mitoid 3425723

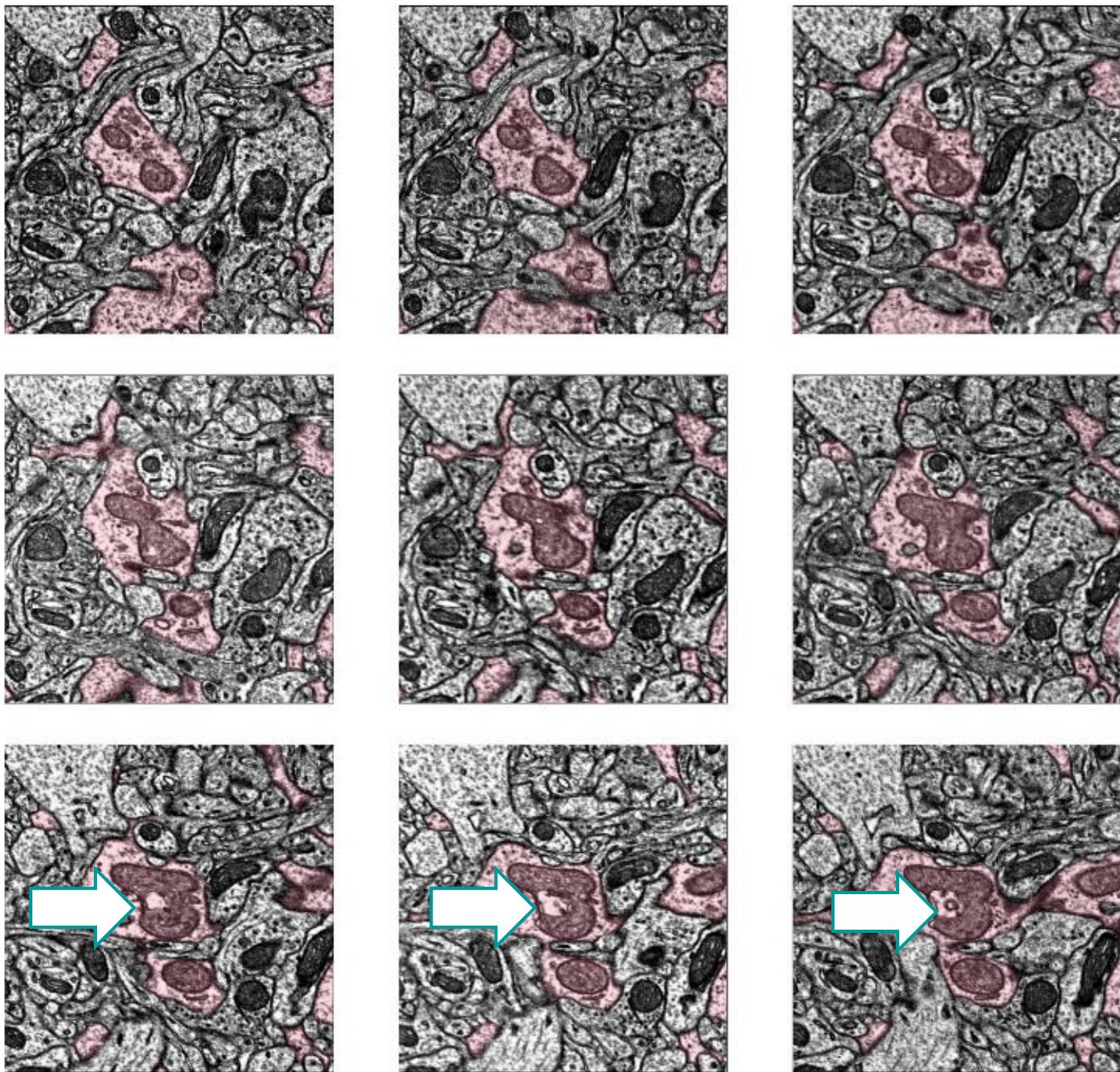
Mito 3425723



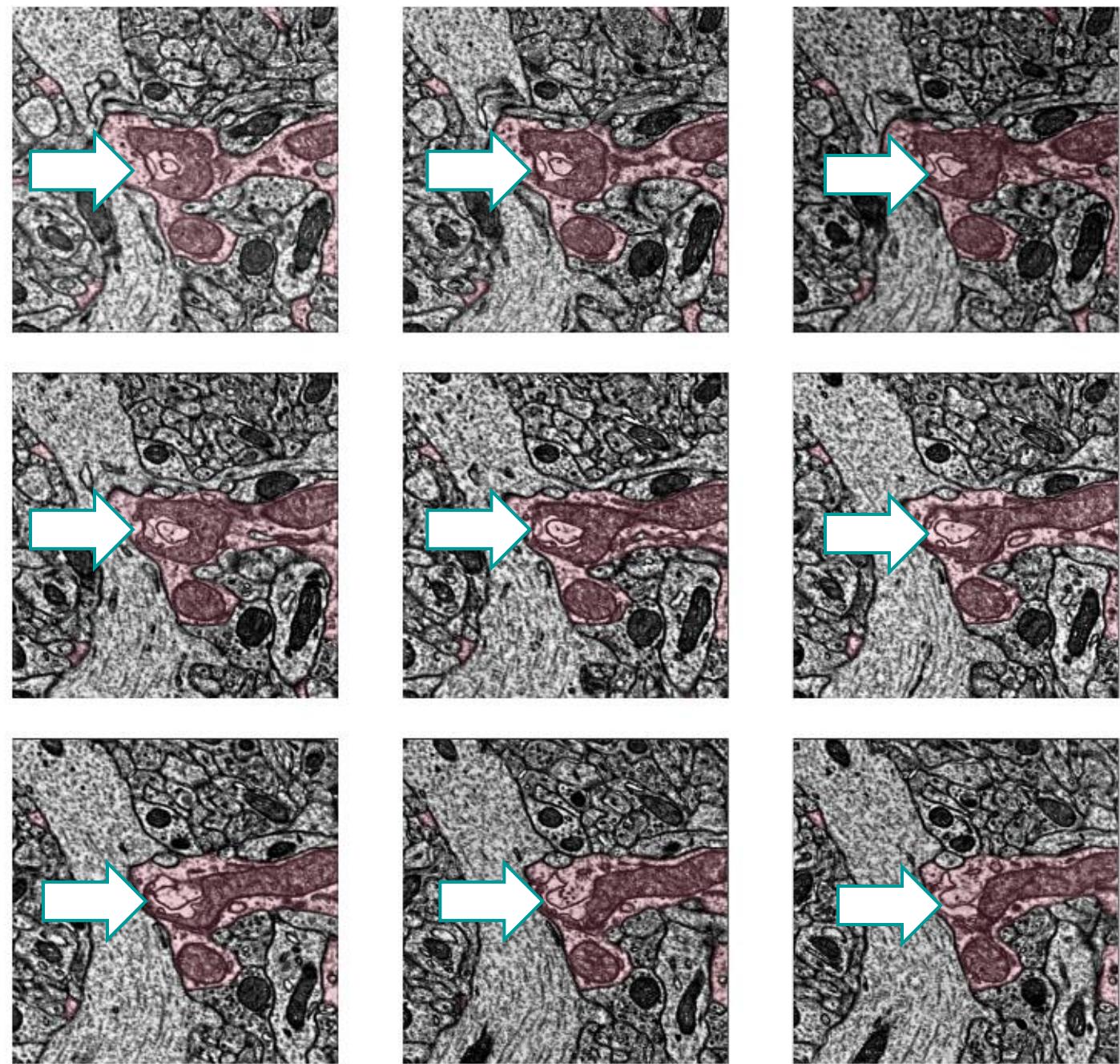
Mito 3425723



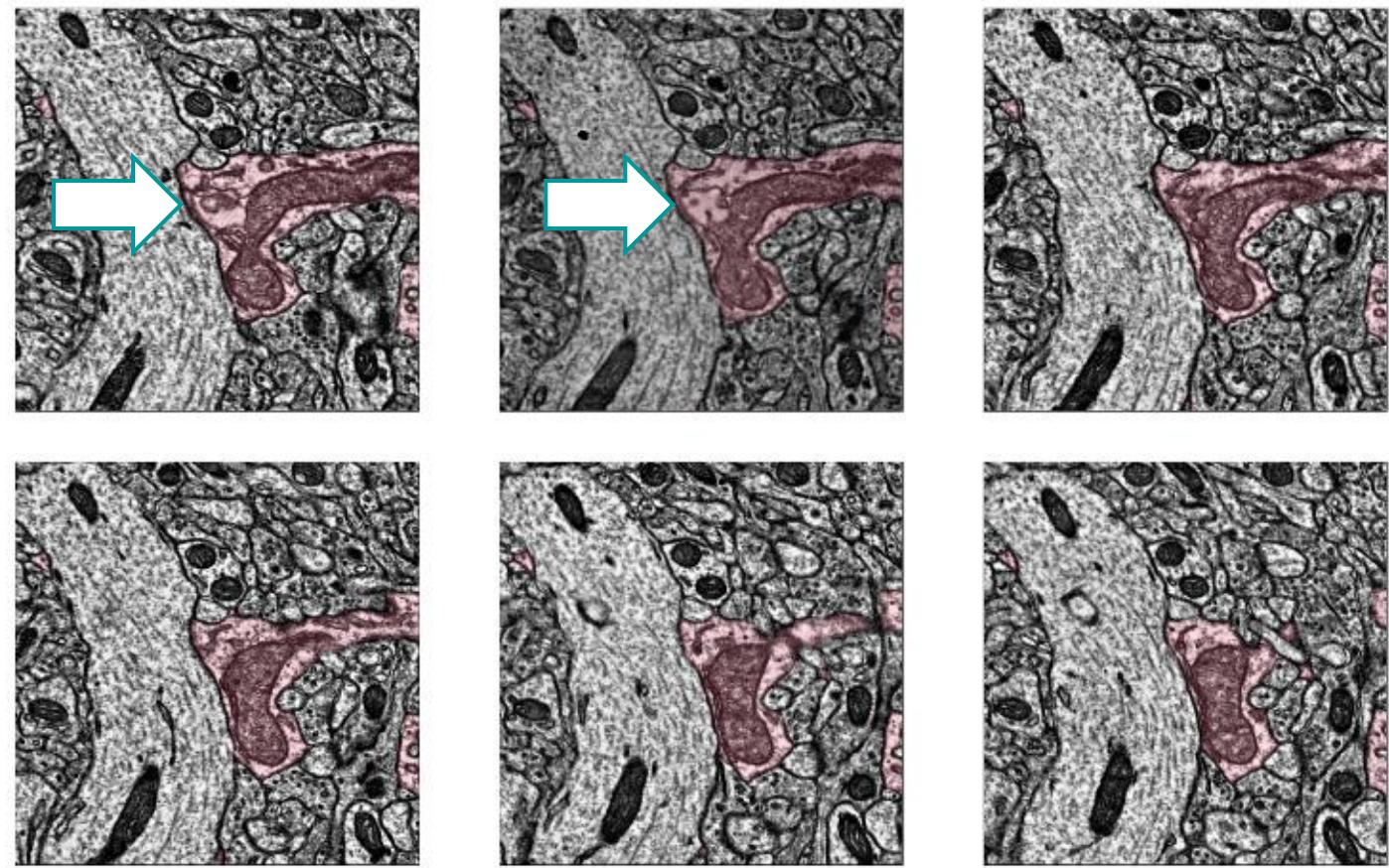
Mito 3425723



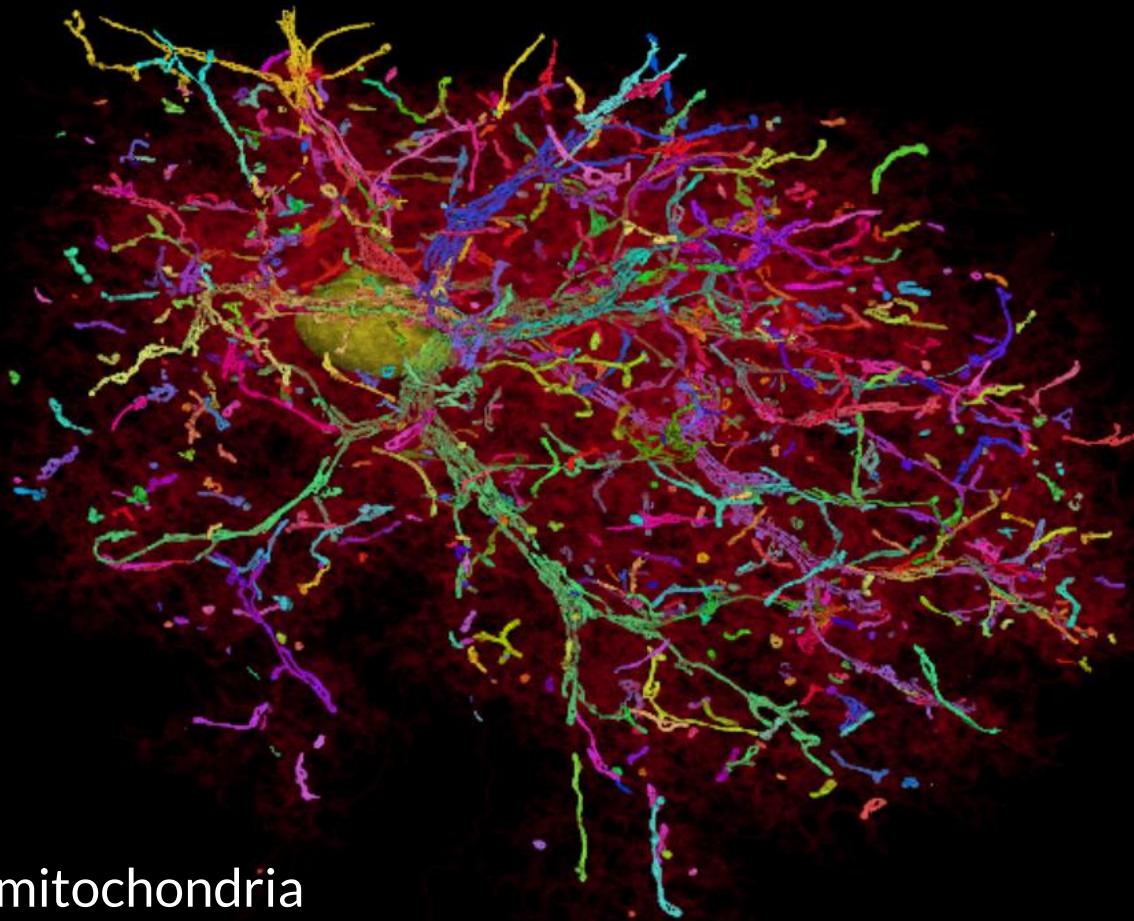
Mito 3425723



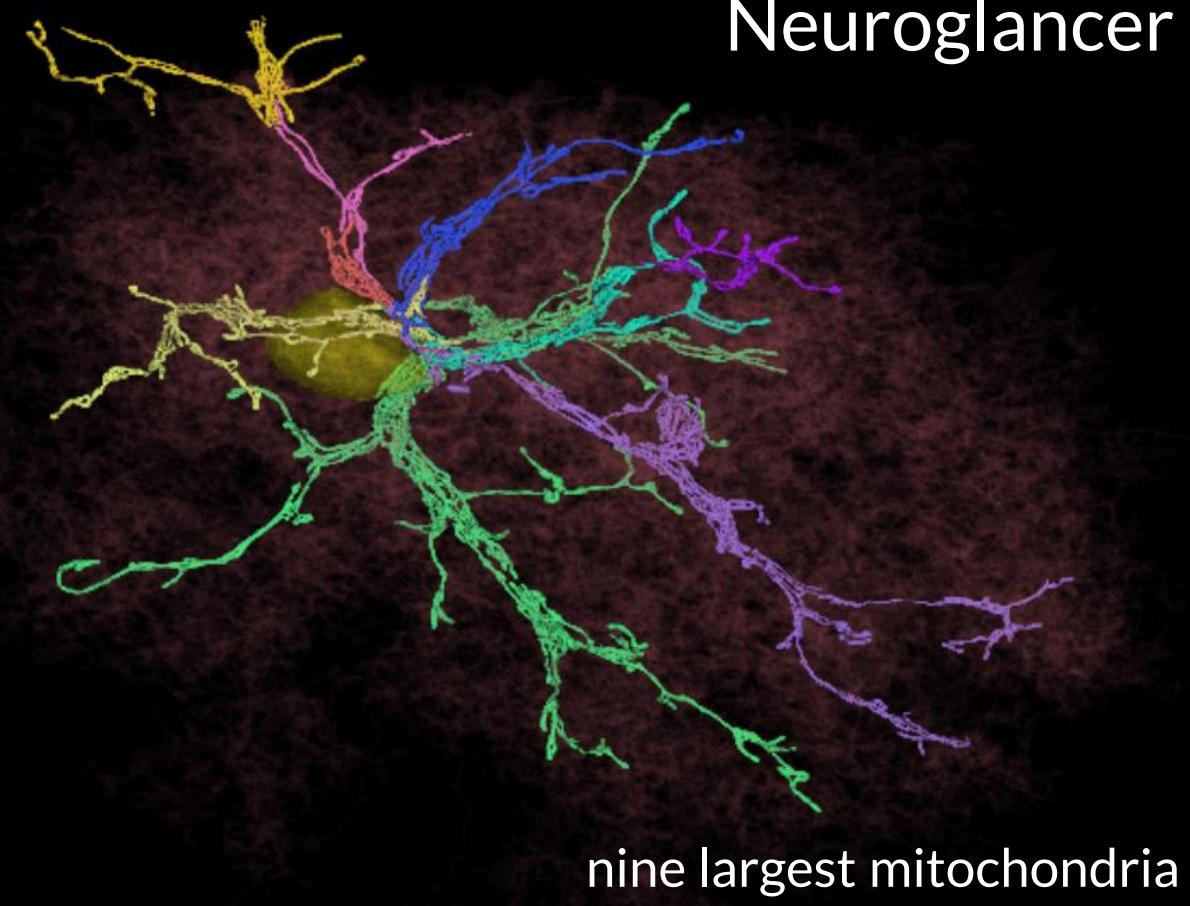
Mito 3425723



Astrocyte 648518346349527319



all mitochondria



nine largest mitochondria

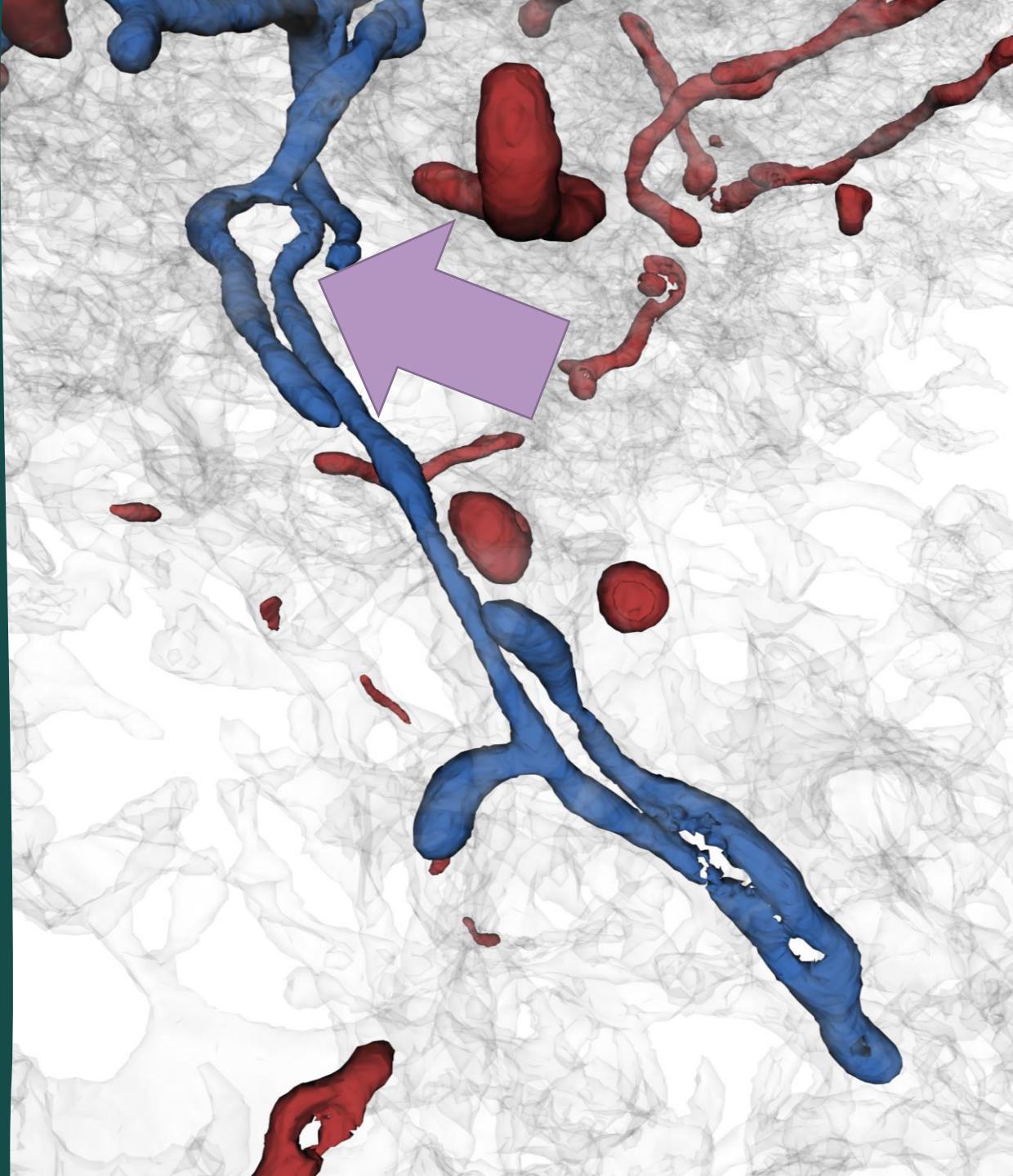
Neuroglancer

Inclusion example 4

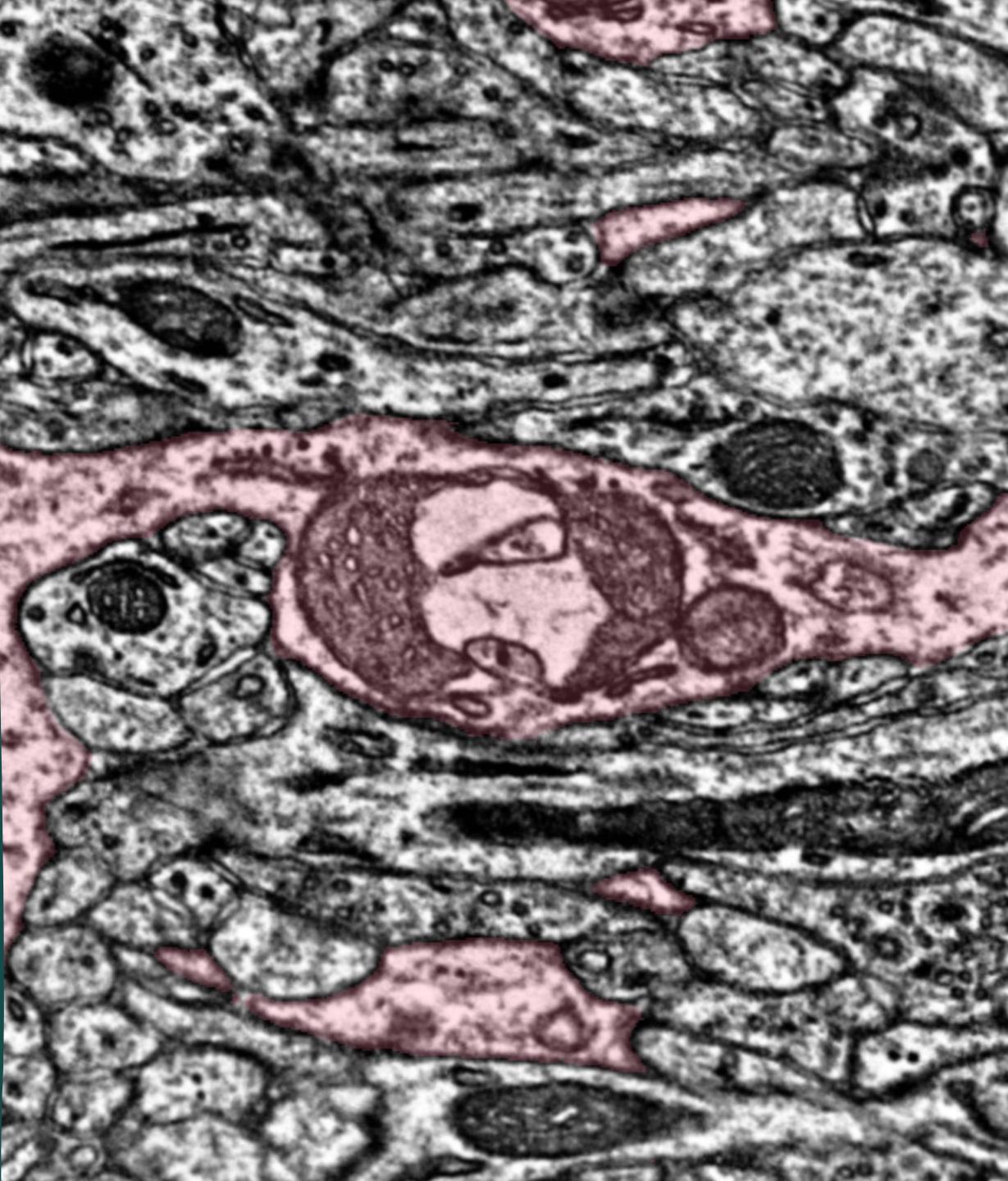
Cellid 648518346349527319

Mitoid 980814

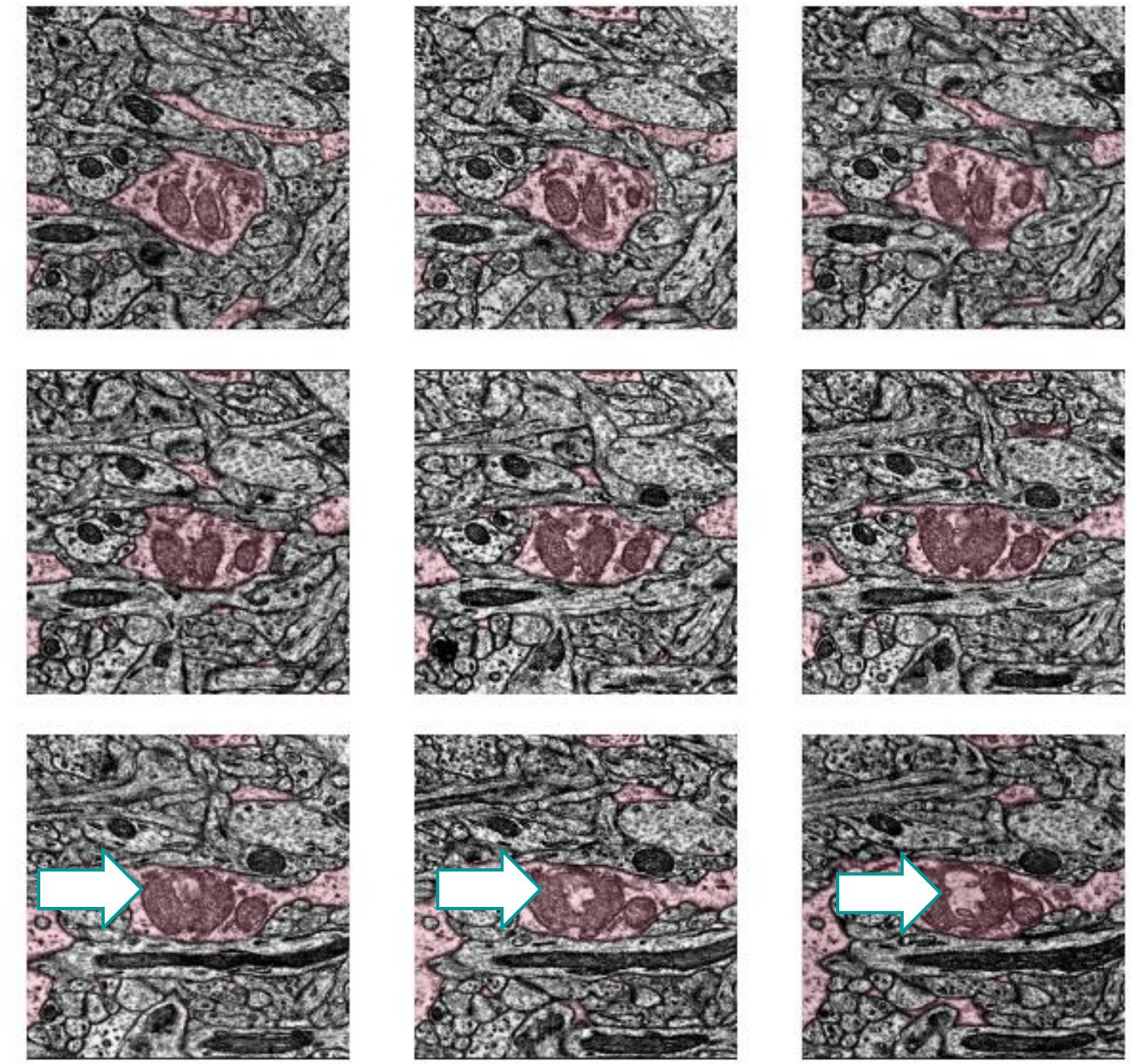
Mito 980814



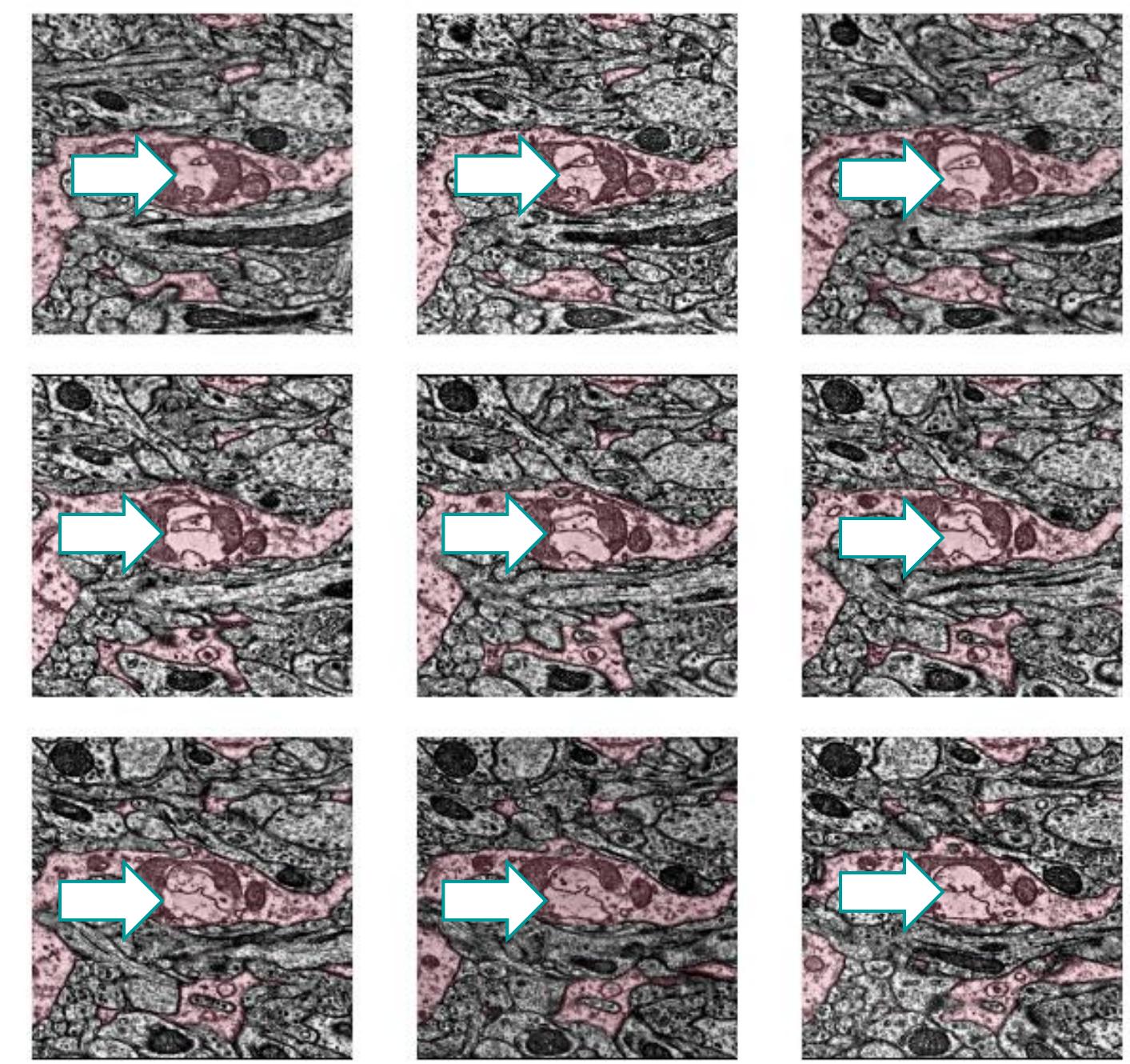
Mito 980814



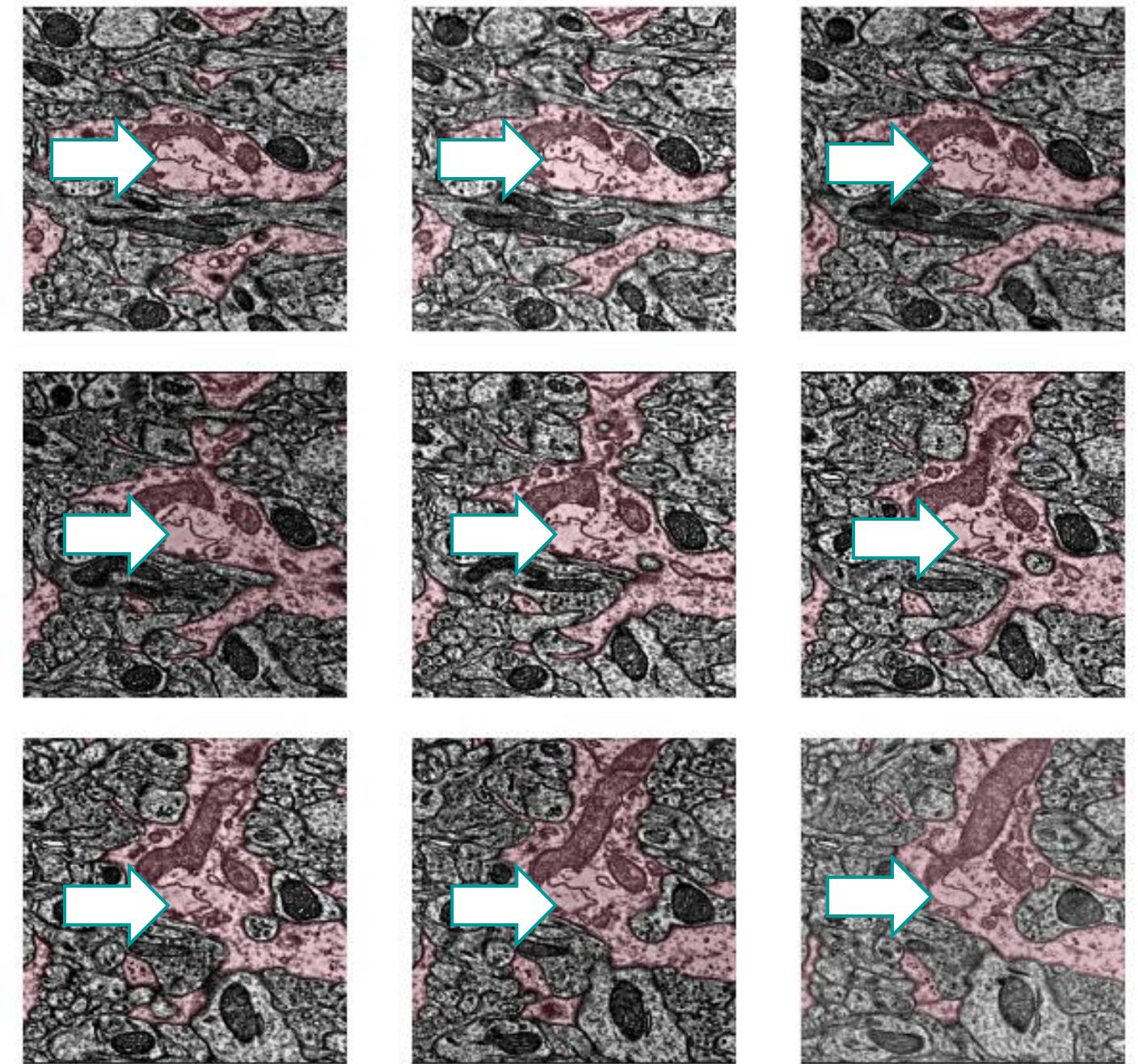
Mito 980814



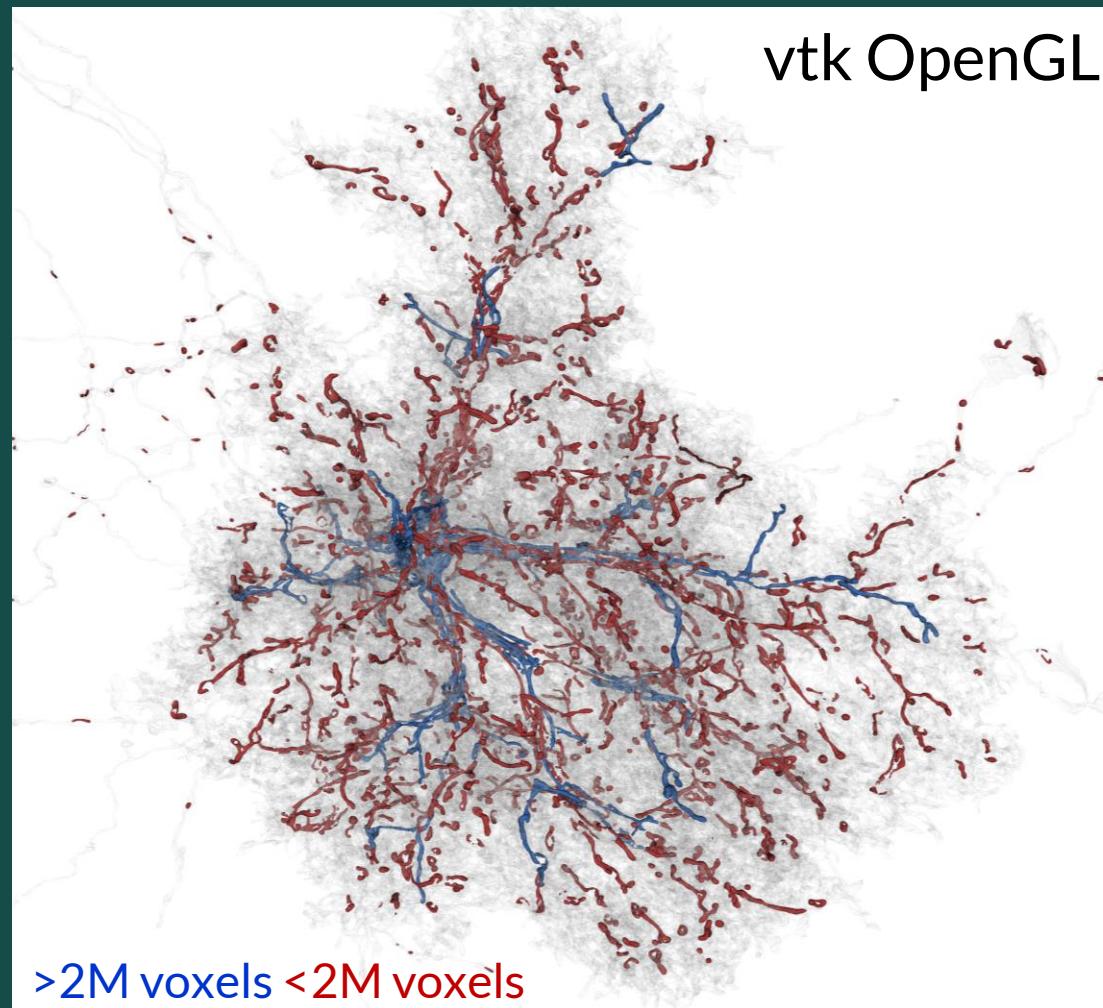
Mito 980814



Mito 980814



Astrocyte 648518346349538089

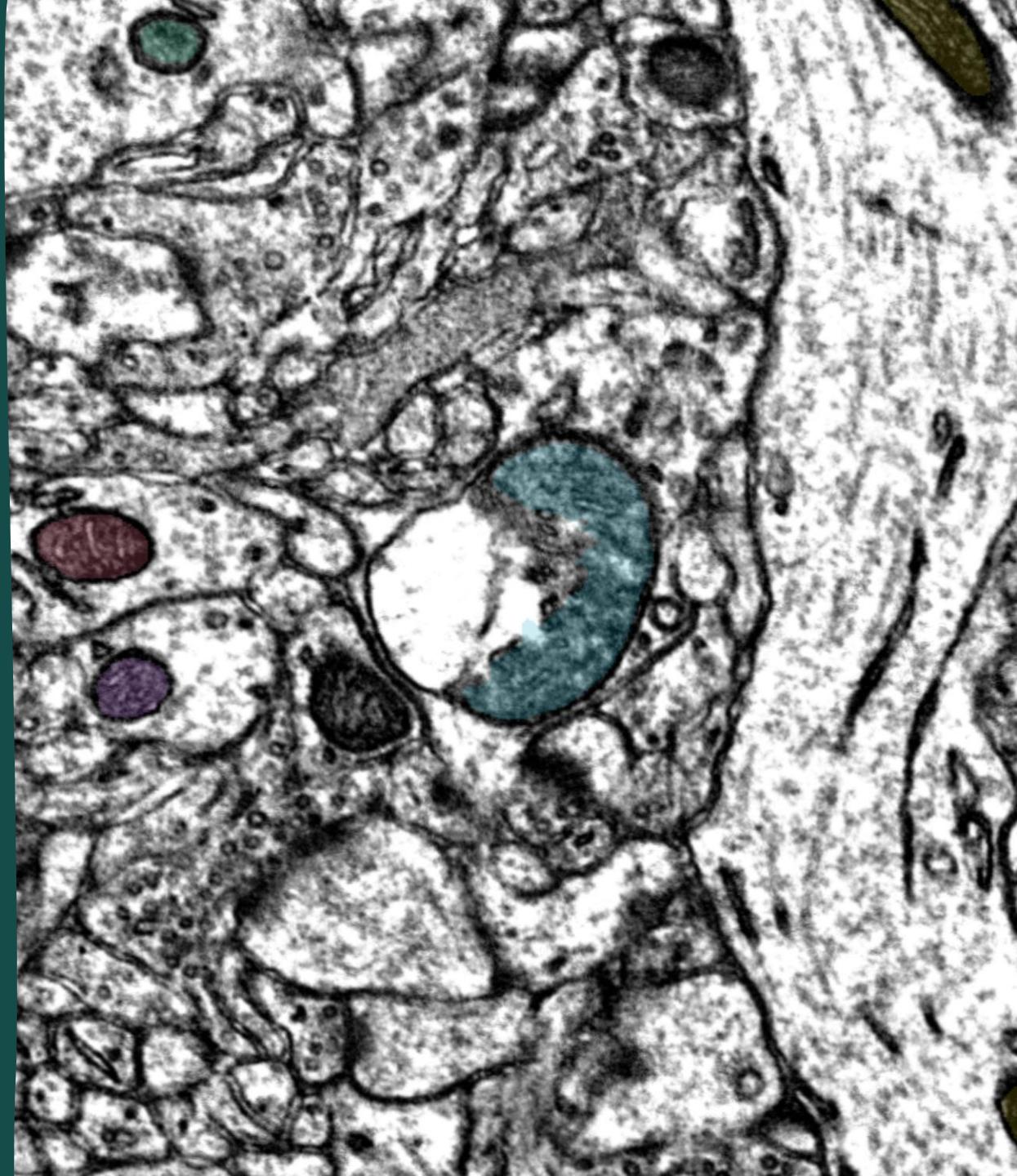


Inclusion example 5

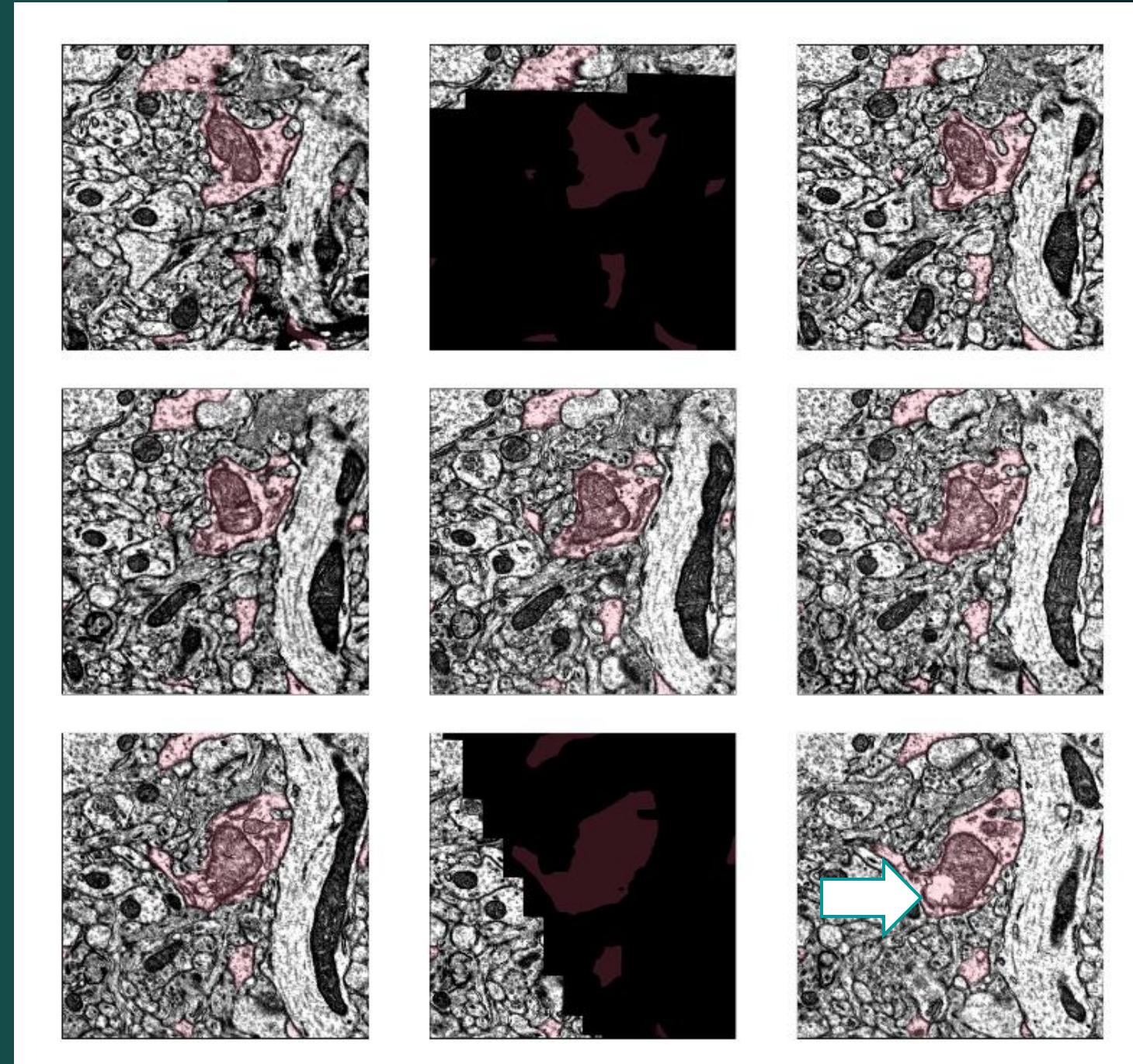
Cellid 648518346349538089

Mitoid 2528399

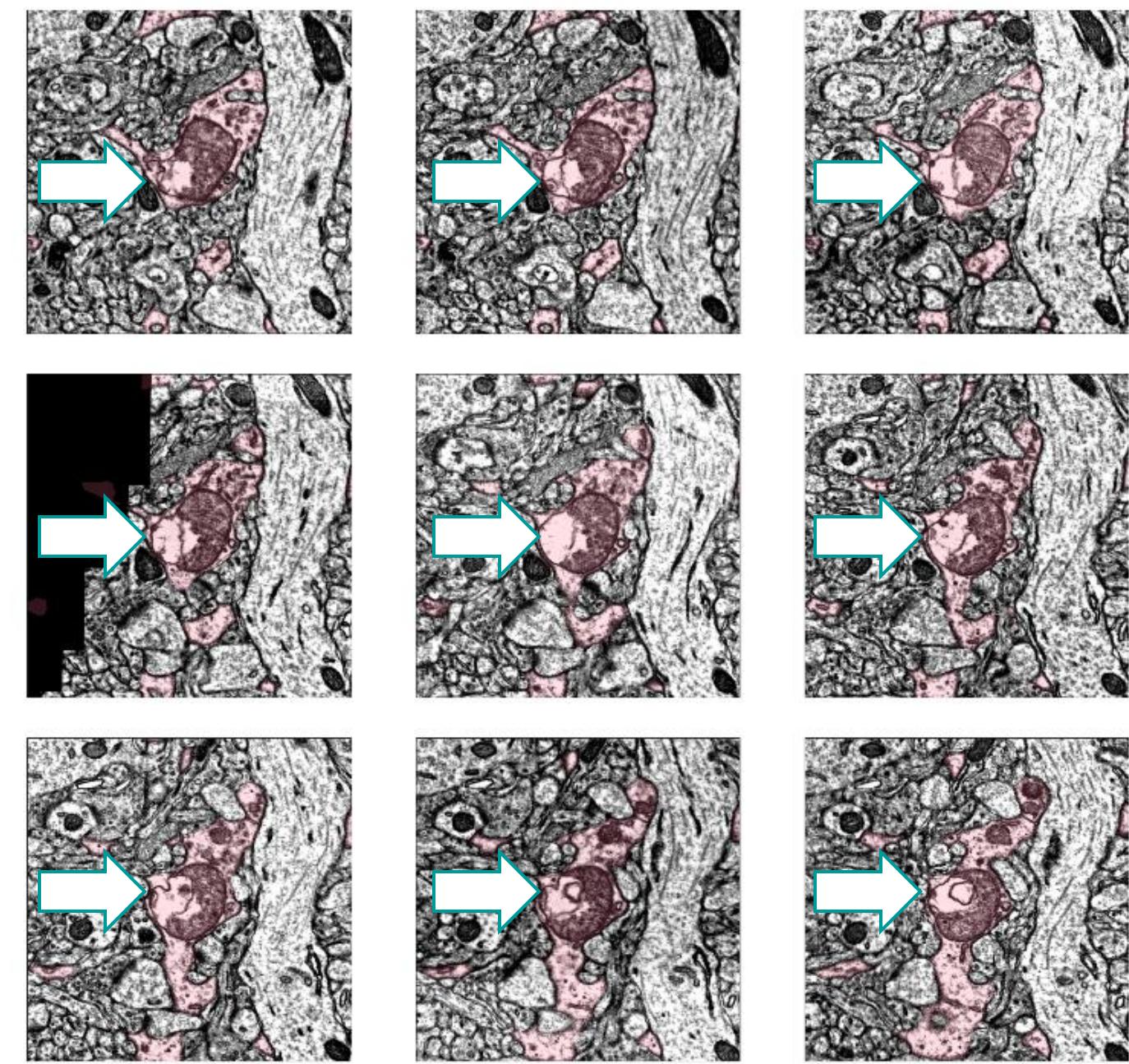
Mito 2528399



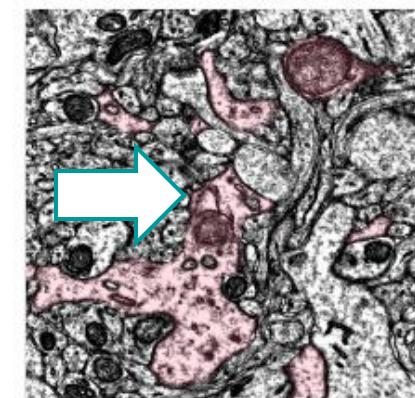
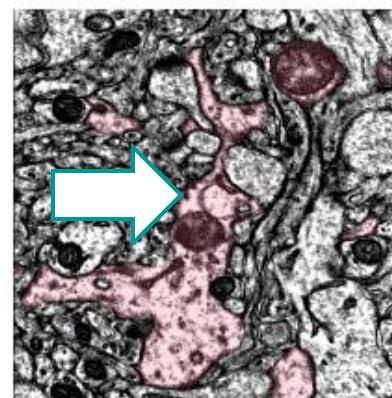
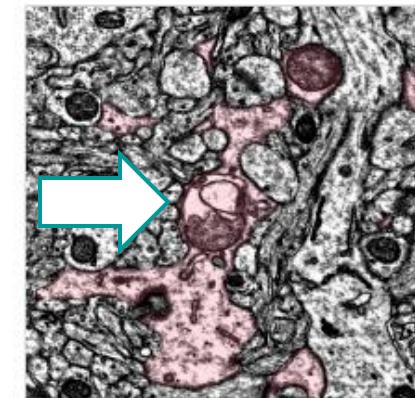
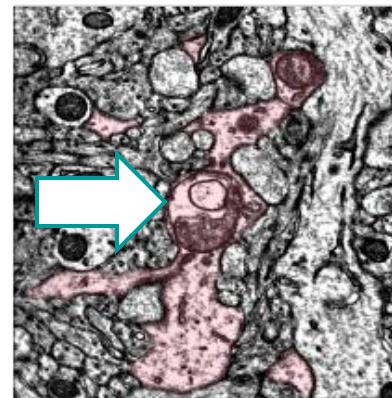
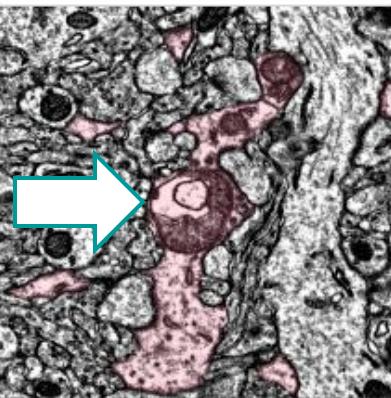
Mito 2528399



Mito 2528399



Mito 2528399

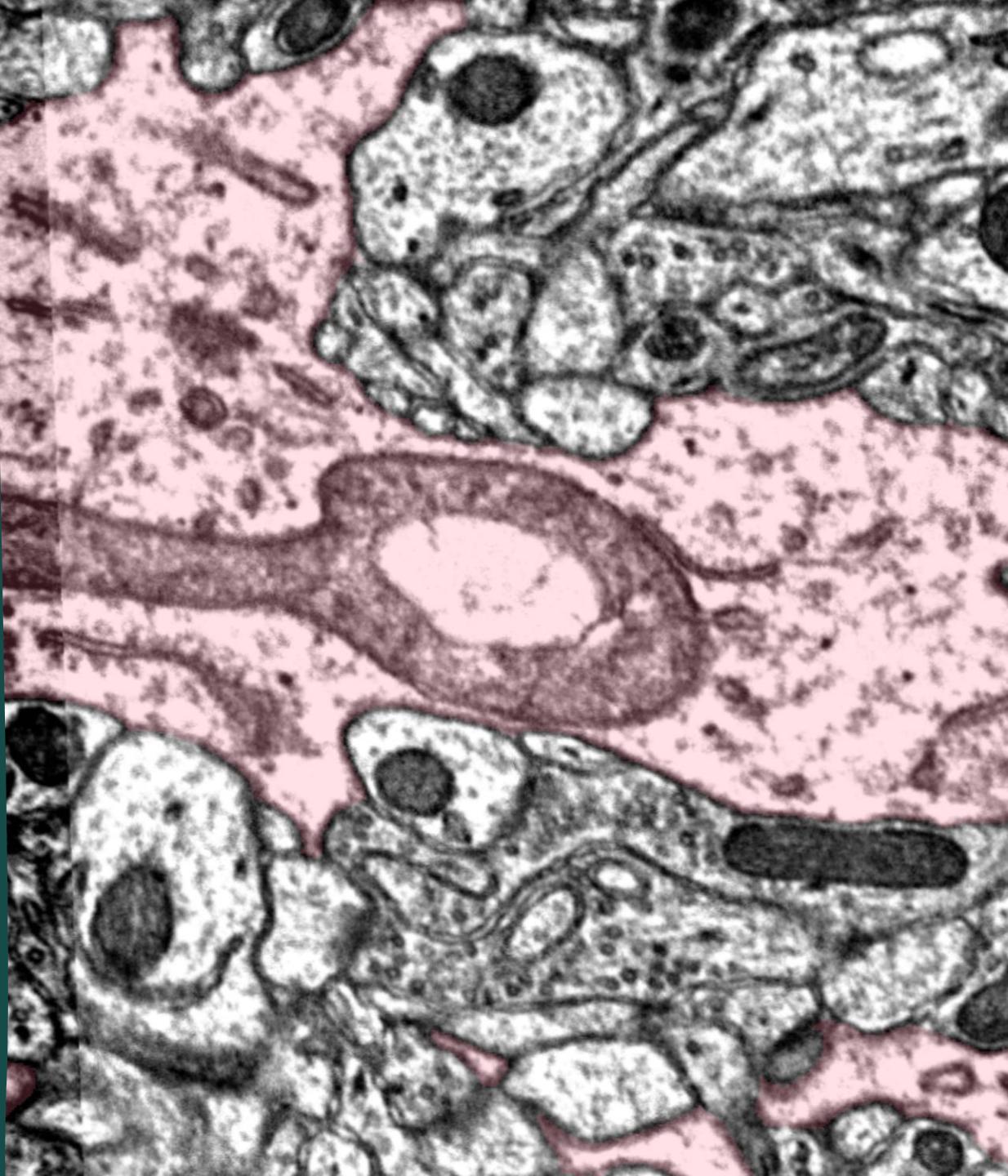


Inclusion example 6

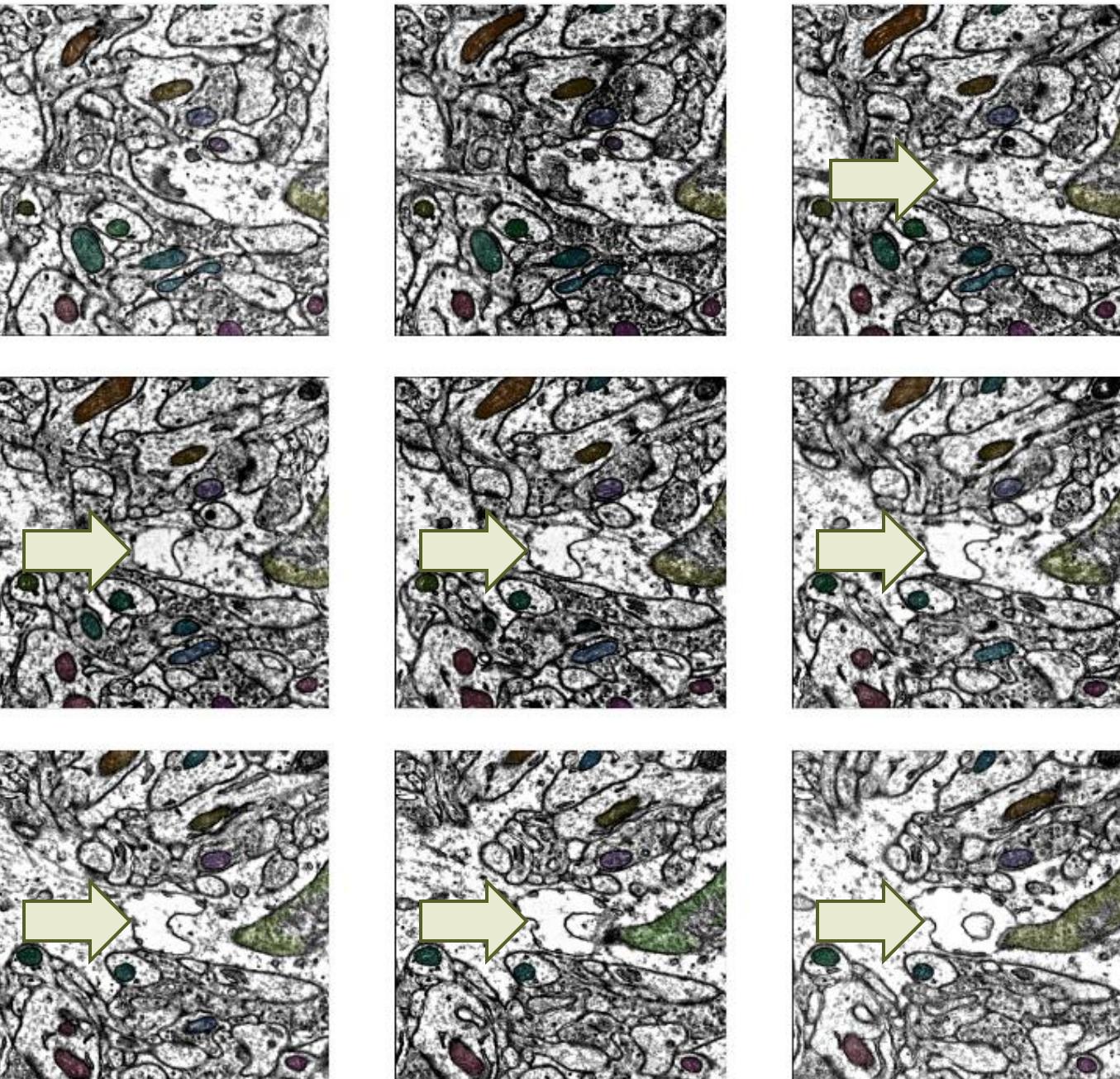
Cellid 648518346349538089

Mitoid 2646319

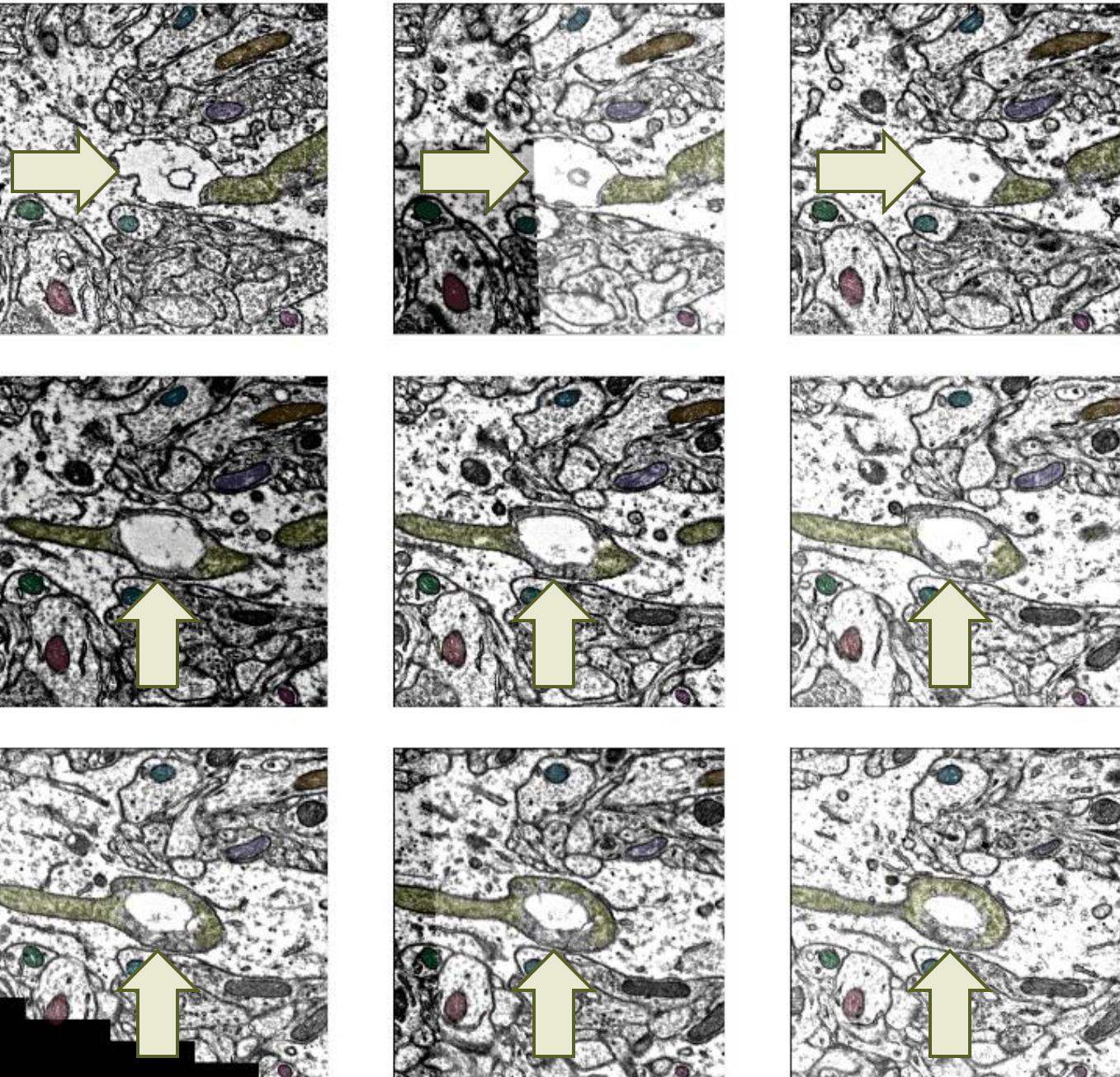
Mito 2646319



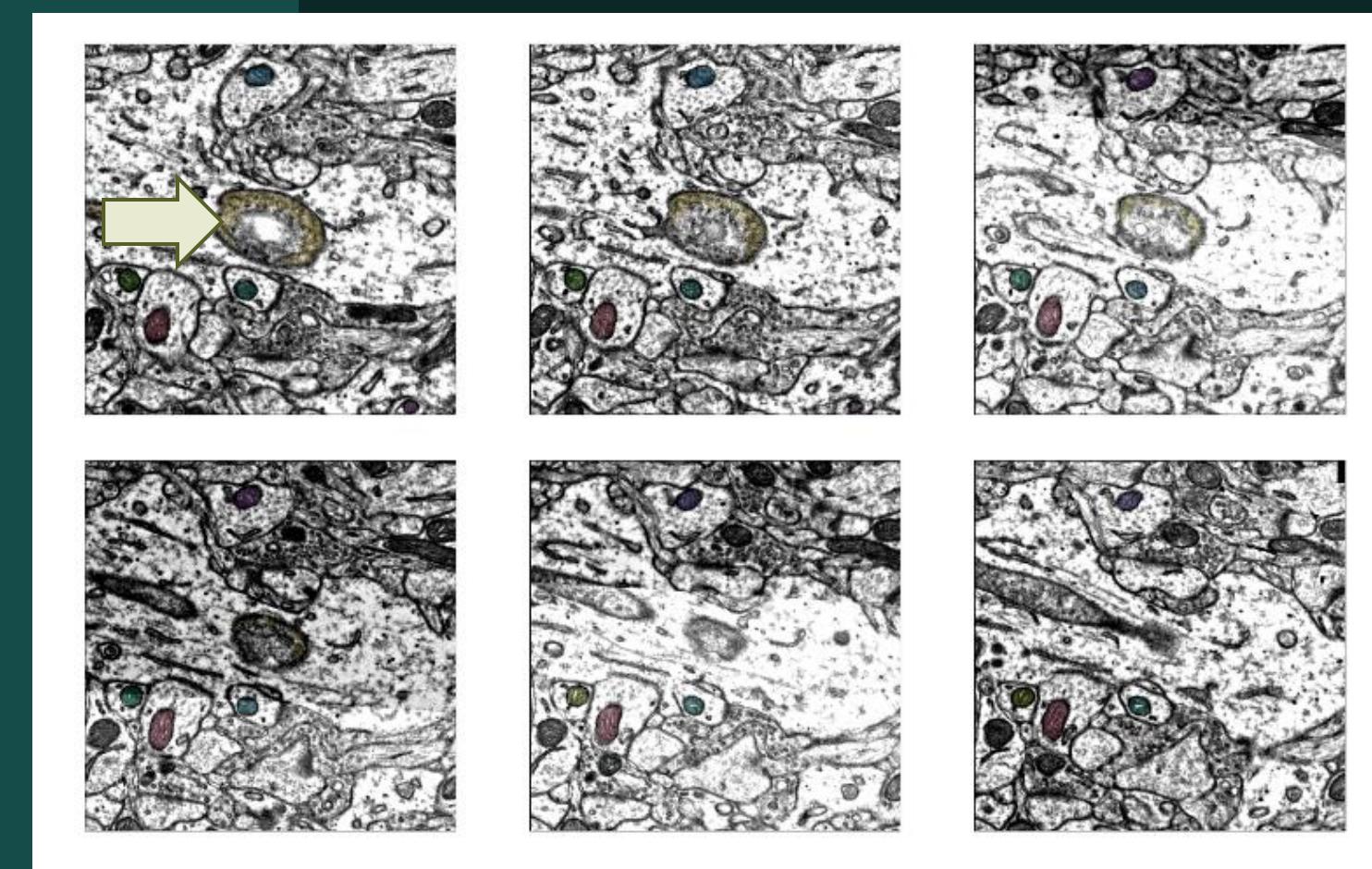
Mito 2646319



Mito 2646319



Mito 2646319



Summary

- Electron bright mitochondria inclusions are commonly seen in astrocytes
- Inclusions can be found relatively easily by manual inspection in Neuroglancer by looking for fragmented mitochondria in the 3D view
- Once manually selected, EM images can be generated using imageryclient module

Acknowledgements



Read the original research papers on
the [Citation](#) page at Allen Institute



Read the [Terms and Conditions](#) page



Use under Creative Commons
by Attribution 4.0 International



Code Availability

Visit my [GitHub repository](#) to view how the images and data in this presentation were generated

```
    for object to mirror
    mirror_mod.mirror_object = object

    if operation == "MIRROR_X":
        mirror_mod.use_x = True
        mirror_mod.use_y = False
        mirror_mod.use_z = False
    elif operation == "MIRROR_Y":
        mirror_mod.use_x = False
        mirror_mod.use_y = True
        mirror_mod.use_z = False
    elif operation == "MIRROR_Z":
        mirror_mod.use_x = False
        mirror_mod.use_y = False
        mirror_mod.use_z = True

    #selection at the end - add
    mirror_ob.select= 1
    mirror_ob.select=1
    context.scene.objects.active = mirror_ob
    ("Selected" + str(modifier))
    mirror_ob.select = 0
    bpy.context.selected_objects = []
    data.objects[one.name].select = 1
    print("please select exactly one object")

- OPERATOR CLASSES -
types.Operator:
    X mirror to the selected object.mirror_mirror_x"
    "mirror X"
```