DeepContact v1.0 tutorial

May 25, 2022

1. Overview

Our main objective in writing this tutorial is to help you access a user-friendly GUI of DeepContact for High throughput quantification of membrane contact site. To enable more users access to DeepContact, we provide two options for users to apply the software. The first choice was for users who can employ our models and algorisms directly by programming, which would require professions in computing. The other is to incorporate DeepContact into Amira, a popular commercial software for electron microscopic image analysis. This tutorial will guide you through the process of incorporation of DeepContact to the AI module of Amira, then invoking DeepContact functions upon the user-friendly GUI of Amira, and finally output the visual and statistical result via automatic organelle segmentation by DeepContact.

2. Algorithms

The DeepContact source code is available on github: https://github.com/LX-doctorAI1/DeepContact. We encourage you to go through the code directly for a detailed understanding of the algorithms.

3. Installation and Requirements

- GPU: >= GeForce GTX 1080Ti；
- CUDA：9.0
- GPU driver:
  https://www.nvidia.cn/Download/index.aspx?lang=cnhttps://www.nvidia.cn/Download/index.aspx?lang=cnhttps://www.nvidia.cn/Download/index.aspx?lang=cn
- CUDA Install according to documentation of GPU;

https://developer.nvidia.com/search?page=1&sort=relevance&term=cuda%20 9.0

- cuDNN：7.0

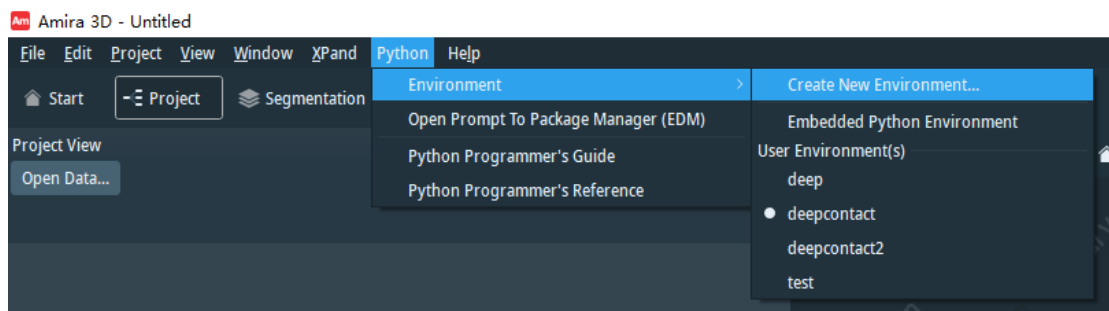 https://developer.nvidia.com/cudnn-download
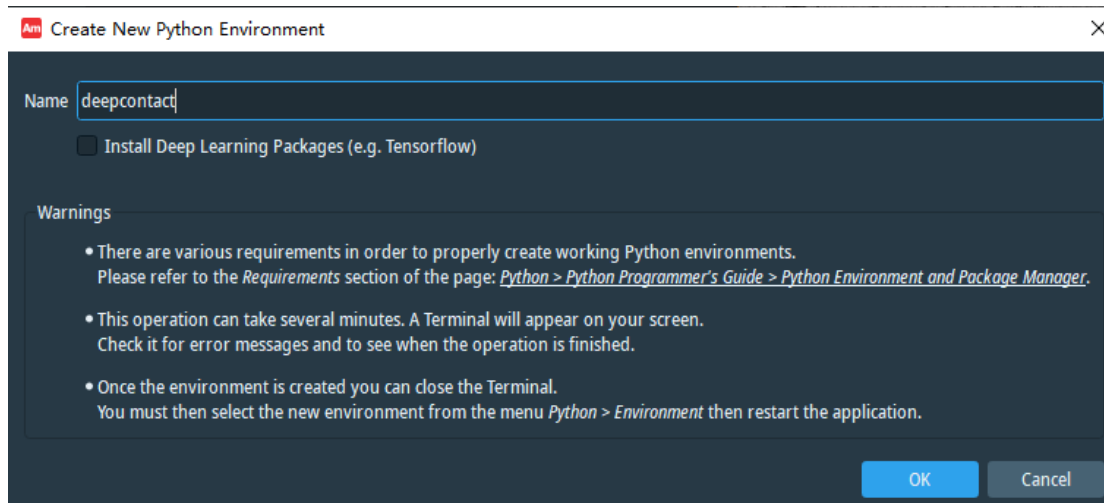
- Amira 2021.2, commercially available

4. Install operating environment for DeepContact in Amira

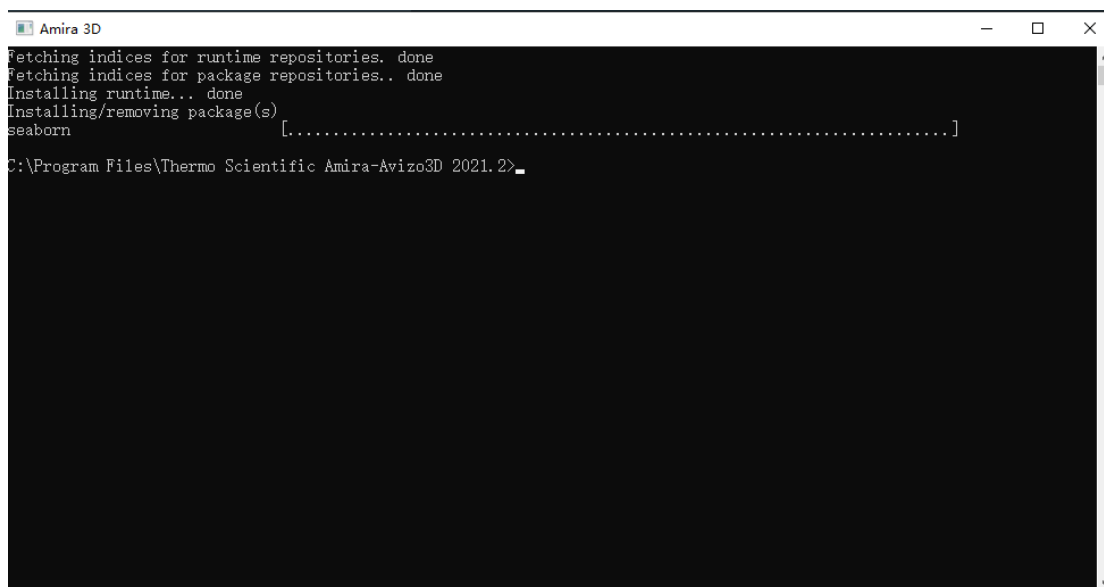### 4.1 Creating Python environment

- Click menu bar Python" -> "Environment" -> "Create New Environment"

- Type "deepcontact" as the title of the new environment in the input frame as indicated below, then Click "OK".

- The command-line interface popup and the basal environmental will be installed automatically. Installation is accomplished if the interface below appears.

- After the creation of deepcontact environment, select "Python"->"Environment" ->deepcontact, restart of Amira will be required,

and the deepcontact environment will come into operation after the restart of Amira.
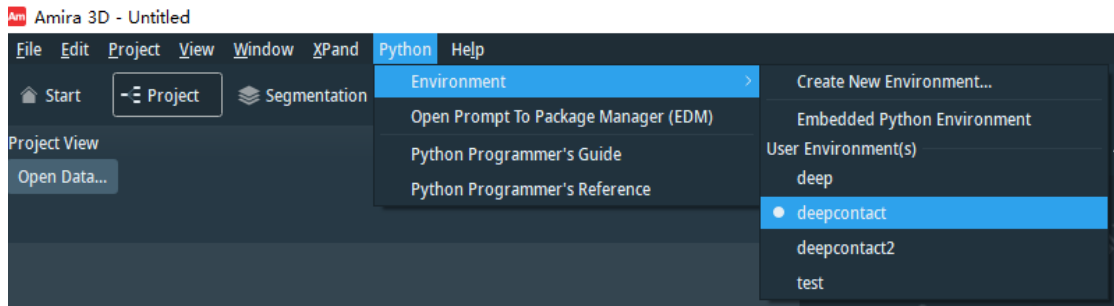

4.2 setup specific environment for DeepContact

- Decompress and open installation package of DeepContact:

  `https://doi.org/10.6084/m9.figshare.19904095.v1`

i. Copy requirements.txt to folder C:\Program Files\Thermo Scientific Amira-Avizo3D2021.2\bin

ii. Download and Run rustup-init.exe in default installation path

   https://www.rust-lang.org/tools/install
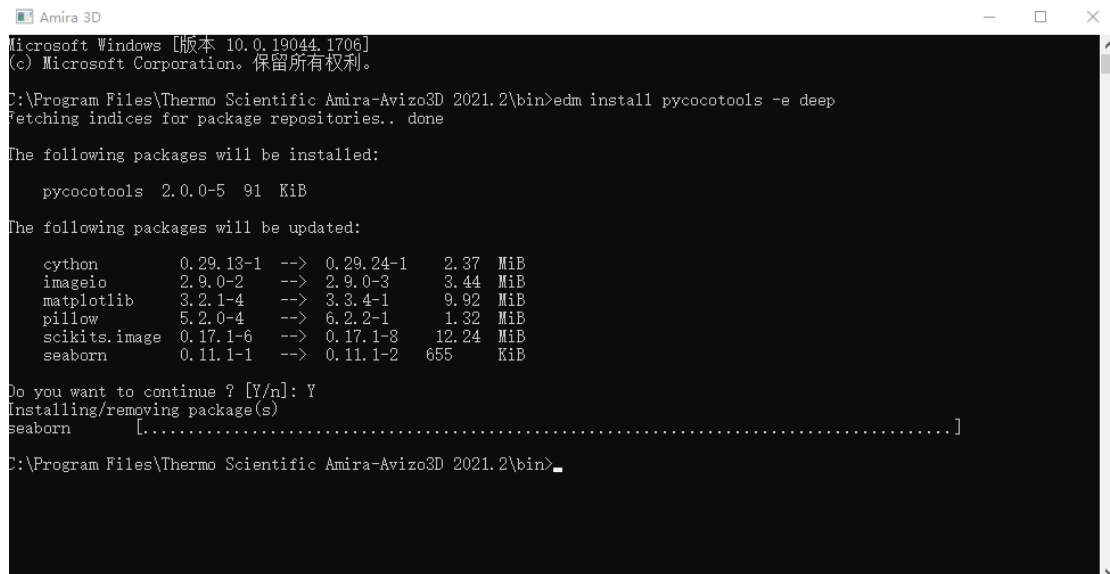
iii. Click menu bar Python" -> "Environment" -> " Open Prompt To Package Manager(EDM)"
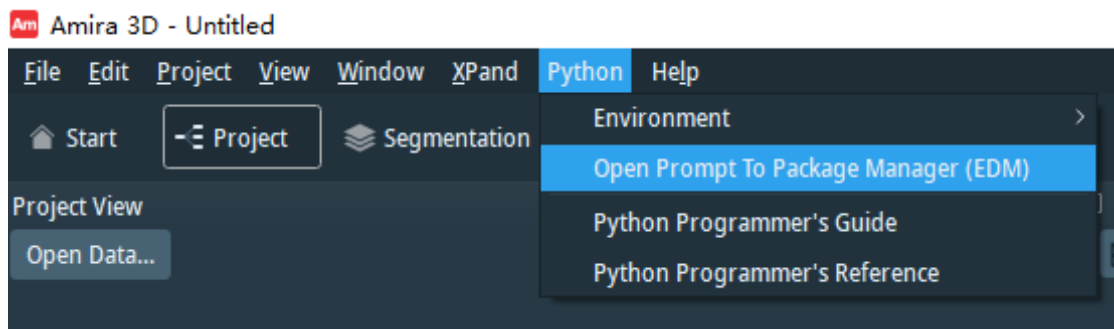
iv. Input the following command line for installation, type "Y" in confirmation lines.

   edm install pycocotools -e deepcontact

edm install pywinpty -e deepcontact
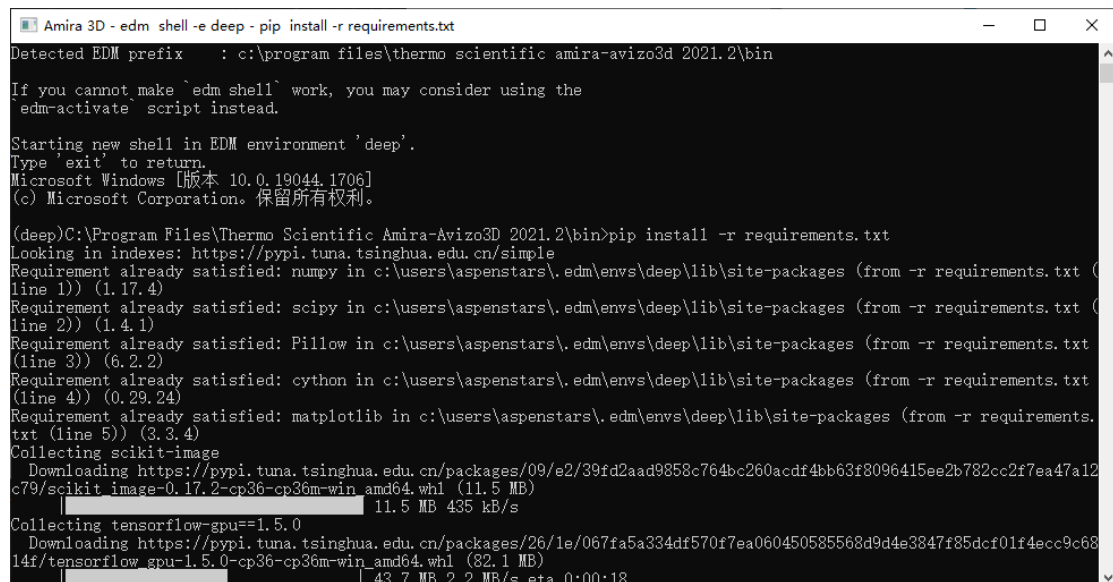
v.   Input edm shell -e deepcontact to activate Python environment and input

the following two command lines to install the remaining Python packages.

pip install labelme

pip install -r requirements.txt

vi.　　Then execute the following command lines:

pip uninstall protobuf

pip uninstall google

pip install google

pip install protobuf

4.3 Installation of DeepContact in Amira

Open installation directory: C:\Program Files\Thermo Scientific Amira-Avizo3D2021.2\share

Decompress and open installation package of DeepContact software from Figshare (https://doi.org/10.6084/m9.figshare.19904095.v1), copy the 3 files in 3 folders under the DeepContact package to the corresponding homonymous folders in the installation directory, restart Amira to complete the DeepContact installation.

eg.

　　i.　　Copy python_modules folder to C:\Program Files\Thermo Scientific Amira-Avizo3D 2021.2\share\python_modules
　　ii.　　Copy python_script_objects to C:\Program Files\Thermo Scientific Amira-Avizo3D 2021.2\share\python_script_objects
　　iii.　　Copy xml to C:\Program Files\Thermo Scientific Amira-Avizo3D 2021.2\share\xml

4.4 Download model files of DeepContact:

`https://doi.org/10.6084/m9.figshare.19845940` **and copy the files to the folder C:\Program Files\Thermo Scientific Amira-Avizo3D 2021.2\share\python_modules\Deepcontact\checkpoint**
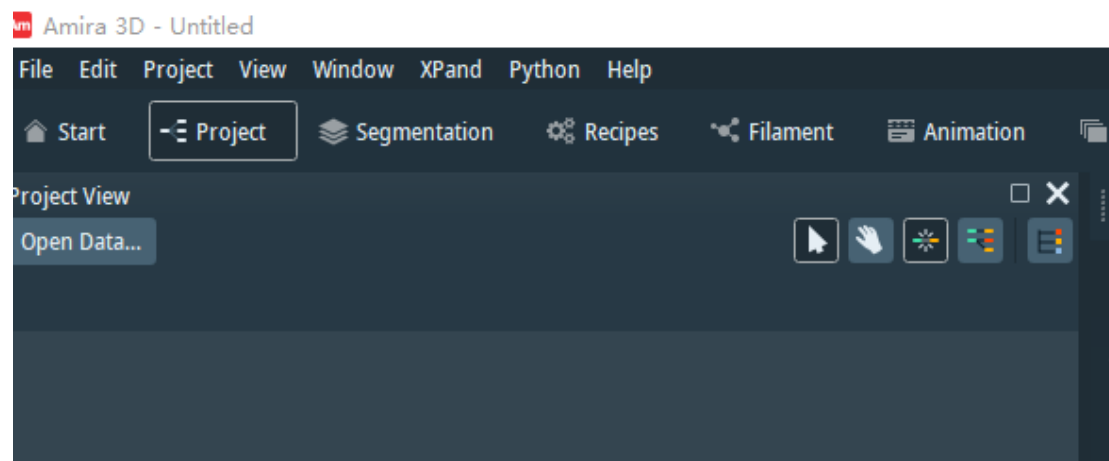
Model types:

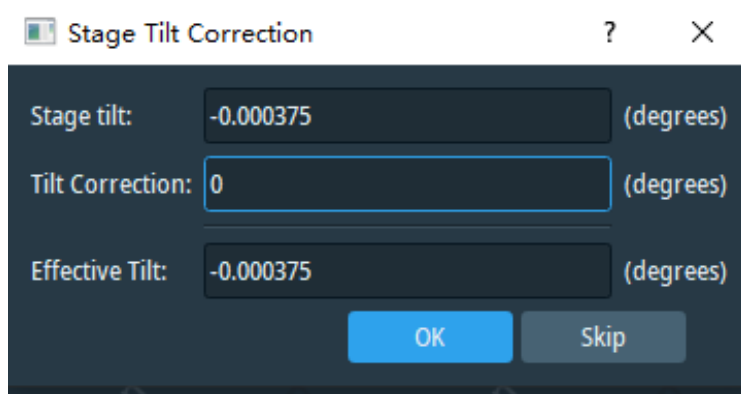|	ER	Mito	LD
Culture cell (SEM+TEM)	cell_er.pth	cell_mito.h5	cell_LD.h5
Tissue SEM	sem_er.pth	sem_mito.h5	
Tissue TEM	tem_er.pth	tem_mito.h5	

4.4 Restart Amira to validate the models

5. Usage of DeepContact in Amira GUI

5.1 Select an image file

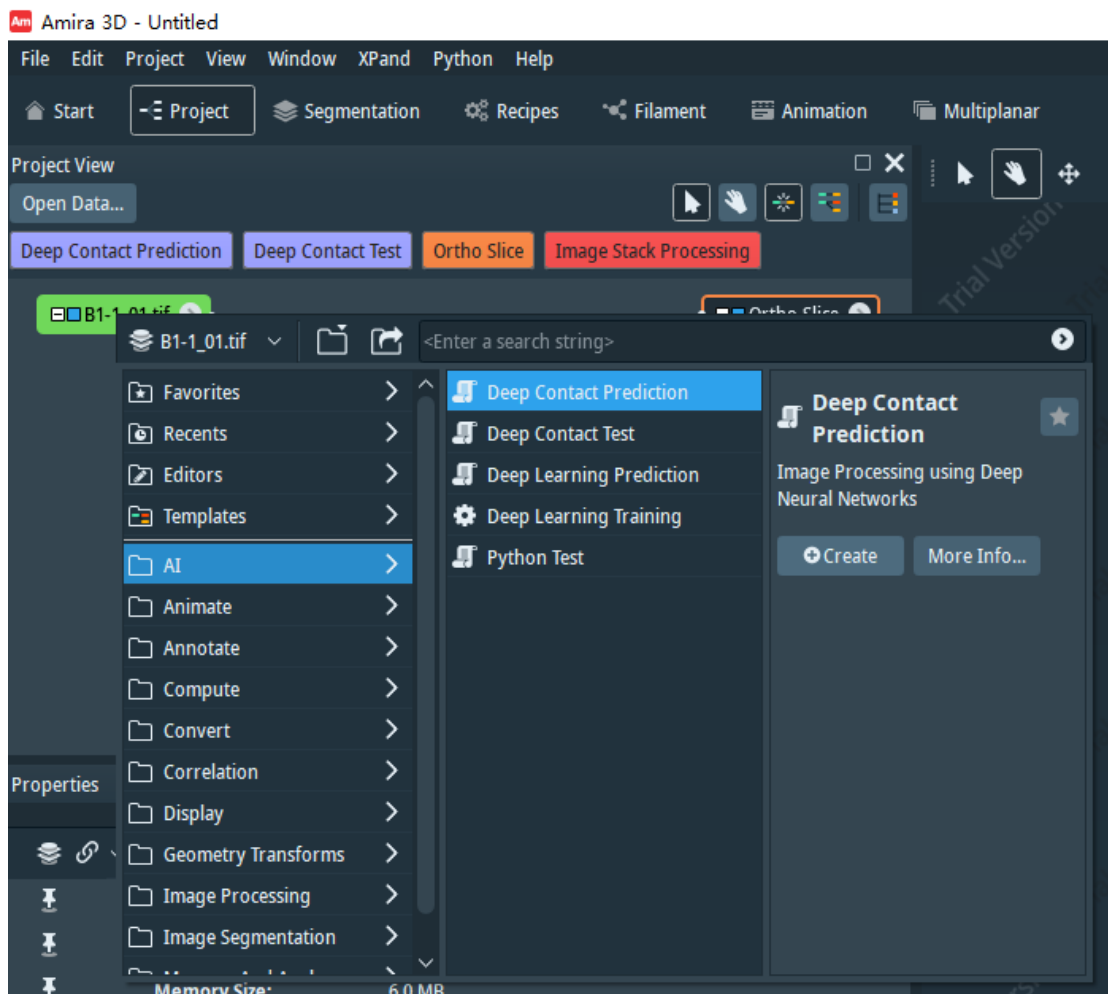Click Project -> open data, and select an Image file for processing

5.2 confirm the property parameters of the image file.

5.3 Create Deep Contact Prediction function

Right click on the image panel, select AI -> Deep Contact Prediction -> Create.

5.4 Define "Deep Contact Prediction" functional parameters

i.   Inputs:" Single" or "Batch" can be selected here.

Single: process initially selected data, and the visualization result will be shown directly in Amira;

Batch: An input folder must be selected, and all the images will be processed automatically. An output folder should be chosen if Batch option is selected here.

ii.   Outputs: an output folder is selected to save the processed to save the processed visual and statistical results.

iii.   Args：

Data Type: Select data type, including Cell, Tissue SEM and Tissue TEM and the following path for model file will be refreshed automatically (you can also choose your own model file);

Crop Size: the size of the cropping file is calculated systematicly, which can't be modified. It is recommended to crop the image to the same size for result visualization, otherwise the cropping area will be selected randomly.

Resolution: the system will transform the image into standard size according to the resolution parameter.

iv.   Model:

Mito Weight File: select mitochondria model file (the default model file could be changed);

ER Weight File：select ER model file (the default model file could be changed);

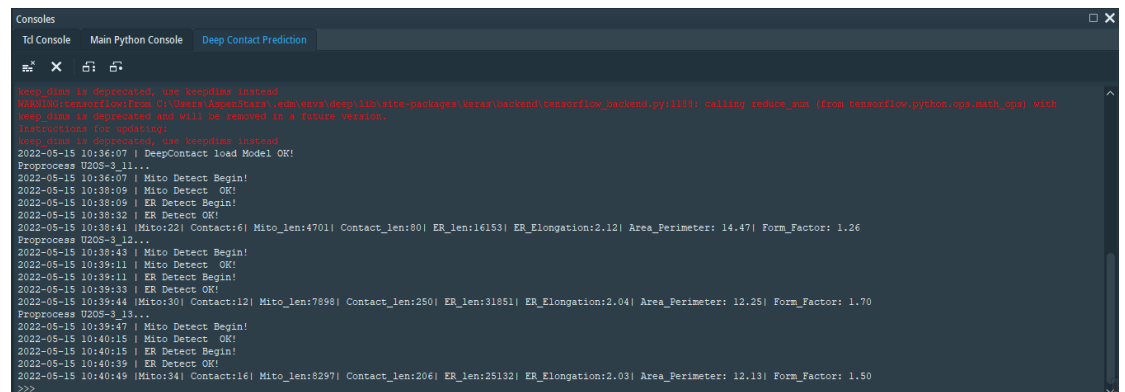v.   Console: Click "Show" to display the status of the program operation in Console.

5.5 Click "apply" to execute the task

The loading of the program will take some time in the initial run, which requires the conformation of the user. Click "Continue" to execute task immediately, Click "Quite" to quit the task.
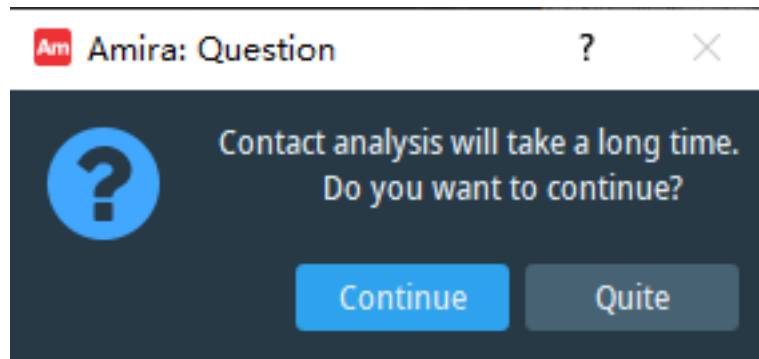
5.5 Result

i. The statistical result will be displayed in Console, and the information marking by red color can be ignored, which will not affect the execution of the program.

ii.   When select "Single" in the "Input" panel, the visualization result will be displayed in Amira with "Ortho Slice".

iii. When select "Batch" in the Input panel, both the visualization result and two *.csv files with statistical results will be given in the folder named in "Outputs" panel.

The two statistical results include:

a)   result.csv

File_name: file name

Mito_number: the total number of mitochondrial in an EM image

Contact_number: the number of all segmented contacts in an EM image

Mito_length: perimeter of all segmented mitochondrial in an EM image

Mito_length_mean: mean perimeter of the mitochondria

Contact_length: length of all segmented contact in an EM image

Ratio_number: the number of segmented contact in an EM image / the total number of segmented mitochondrial in an EM image

Ratio_length: length of all segmented contact in an EM image / perimeter of all segmented mitochondrial in an EM image

ER_Length: the perimeter of all segmented ER in an EM image

Mito_Elongation: $\text{mitochondria\_perimeter}^2 / (4\pi \times \text{area})$

ER_Elongation: $\text{ER\_perimeter}^2 / (4\pi \times \text{area})$

b) result_dist.csv

dist_N: N is pixel-based contact width intervals. 1pixel corresponds to a length of 10 nm. dist_N give the value of contact length ratio in of N pixel (N×10 nm) width interval.