Application Note #56
Image Registration for Multiplexing Experiments in arivis Vision4D

How to:
import and align multiple 3D images based on the nuclei staining

The application note goal is to guide the user in the correct installation of the Anaconda3 Python package and the image registration module. This script will read multiple image files in a given folder in a tile-wise manner and create the SIS files for each of the original files. During the registration process, the new SIS file will be created with each tile as a new imageset. The first cycle image will be copied from the original file and the image stacks from the following cycles will be registered to the first cycle image based on the indicated channel, typically, nuclei staining. In the last step, the user needs to run the tile sorter tool to stitch the registered mosaic stack together.

Vision4D 3.6.2

Warnings

Vision4D runs multiple applications (image registration) using external and independent Python libraries and tools produced by third parties.

These tools must be installed by the user under its responsibility, strictly following the instruction on this document. arivis has tested the setup protocol on several computers, however, due to the different and not predictable hardware and software configuration of each computer, the results can be different case by case. Therefore, arivis declines any responsability concerning the correct tools installation and setup on the user computer. arivis cannot be blamed about any malfunctioning or failure of the deep learning environment setup. arivis will not give technical support on the setup task. Both activities are totally on the user charge.

Arivis also declines any responsibility about the scientific results gathered from this application.

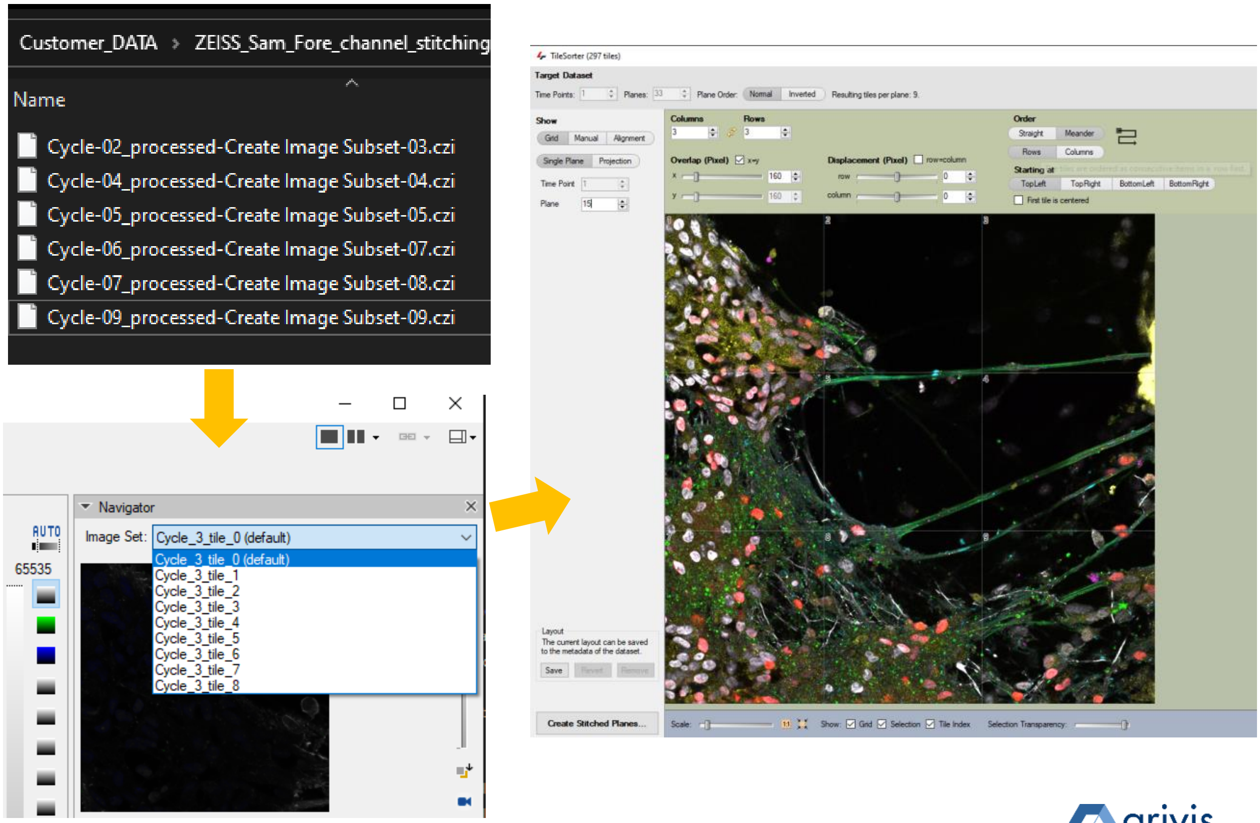
The images used in this application note were published in https://www.sciencedirect.com/science/article/pii/S0165027022001807

Application Overview

Image registration in a nutshell

Image registration is the process of transforming different sets of data to overlay images from different experiments, taken at different times, positions or angles, or with different imaging modalities. The algorithm attempts to discover the matching areas and align them together. Image registration methods can be divided into two large groups: rigid and non-rigid transformations. For the multiplexing experiments, we will use the nuclei staining in each of the experiment cycles to register the subsequent experiments to the first image. This method is applicable both to 2D and 3D images.

The application workflow starts from importing the original image files (for example, CZI) in Vision4D and registering the subsequent experimental cycles to the first one based on the indicated nuclei staining channel (i.e. DAPI or Hoechst). The images will be imported as a stack of individual tiles with all the channels from all the experiment cycles, each registered to the nuclei staining in the first experiment cycle. After the registration, the mosaic image should be stitched using the Tile Sorter tool.



Customer_DATA › ZEISS_Sam_Fore_channel_stitching

Name
- Cycle-02_processed-Create Image Subset-03.czi
- Cycle-04_processed-Create Image Subset-04.czi
- Cycle-05_processed-Create Image Subset-05.czi
- Cycle-06_processed-Create Image Subset-07.czi
- Cycle-07_processed-Create Image Subset-08.czi
- Cycle-09_processed-Create Image Subset-09.czi

Navigator

Image Set: Cycle_3_tile_0 (default)
- Cycle_3_tile_0 (default)
- Cycle_3_tile_1
- Cycle_3_tile_2
- Cycle_3_tile_3
- Cycle_3_tile_4
- Cycle_3_tile_5
- Cycle_3_tile_6
- Cycle_3_tile_7
- Cycle_3_tile_8

TileSorter (297 tiles)

Target Dataset
Time Points: 1 Planes: 33 Plane Order: Normal | Inverted Resulting tiles per plane: 9.

Show: Grid | Manual | Alignment; Single Plane | Projection
Time Point: 1
Plane: 15

Columns: 3 Rows: 3
Overlap (Pixel) ☑ x=y — x: 160 y: 160
Displacement (Pixel) □ row=column — row: 0 column: 0

Order: Straight | Meander; Rows | Columns
Starting at: TopLeft | TopRight | BottomLeft | BottomRight
□ First tile is centered

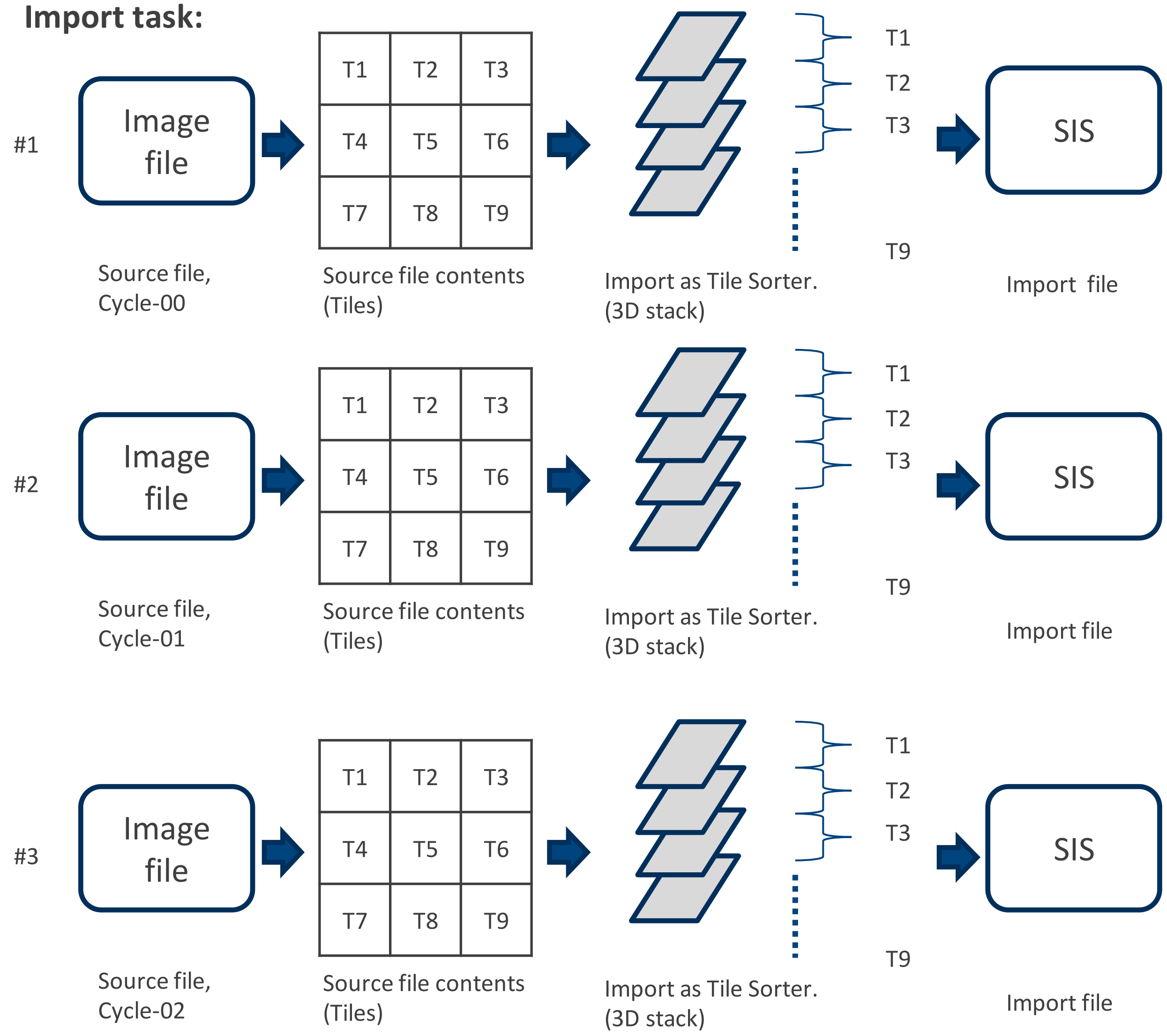
Layout
The current layout can be saved to the metadata of the dataset.
Save Revert Remove

Create Stitched Planes...
Scale: 11 Show: ☑ Grid ☑ Selection ☑ Tile Index Selection Transparency:

Application Workflow

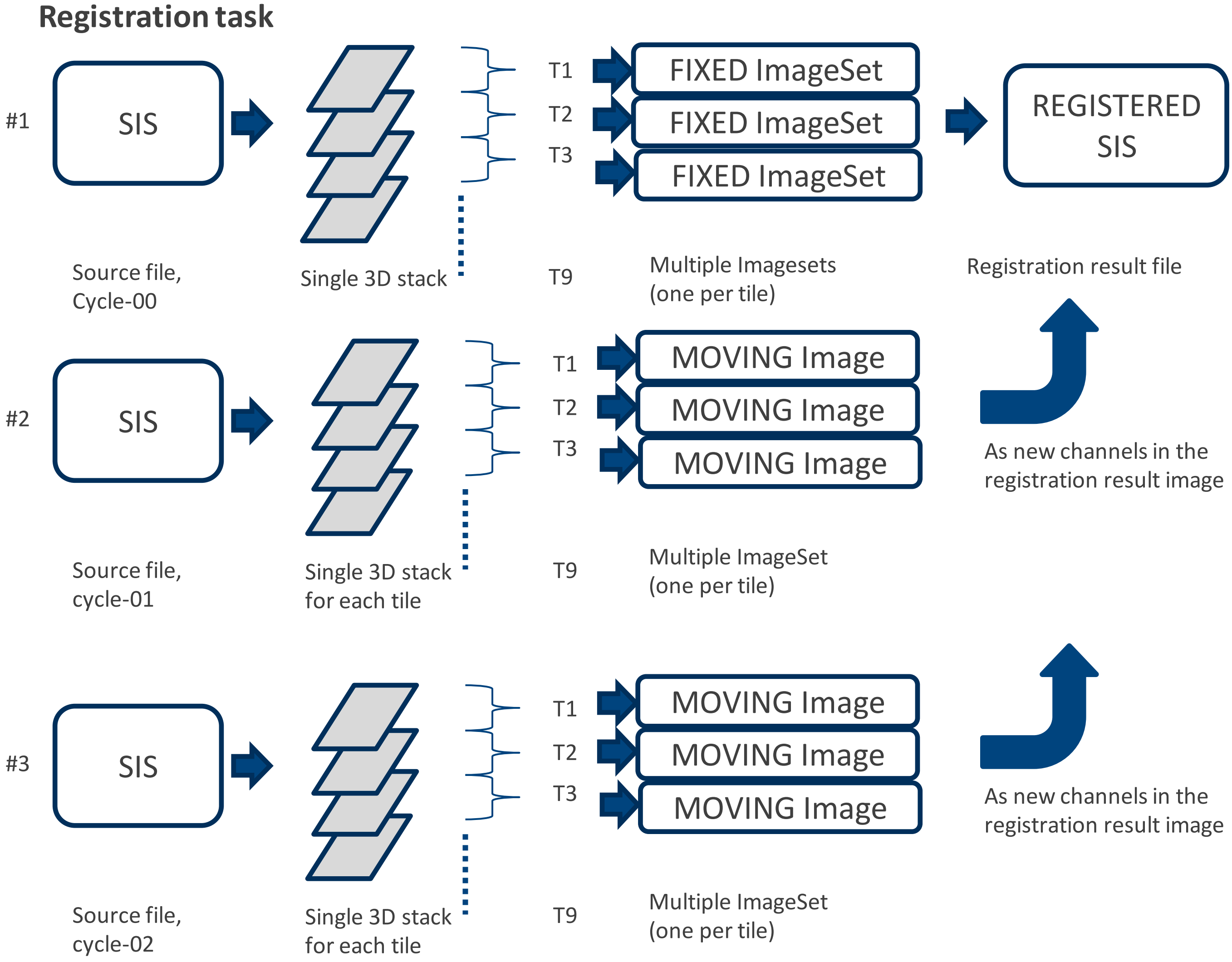
The application workflow starts by importing the original image files in Vision4D. For each staining/hybridization cycle the new SIS file will be created. All Z planes from all the tiles will be imported separately into a single stack (single imageset). The import step is optional. Alternatively, the user can start the registration workflow by simply opening the already stored SIS files.

Import task:



Application Workflow

The registration workflow starts from creating the new SIS file to store the registration results. The first cycle image (*fixed*) will be identified automatically based on the filename. This image will remain untransformed. The 3D stack for each subsequent cycle (*moving*) will be transferred as new channels in the corresponding tile in the registered image file. The number of tiles should be consistent for all staining cycles.



1. Download the Anaconda Package

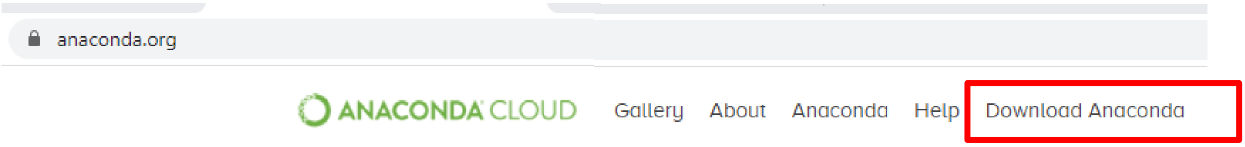
Step 1.1

Open the Anaconda web pages using the following link:

https:/anaconda.org

Step 1.2

Select the Download Anaconda item.

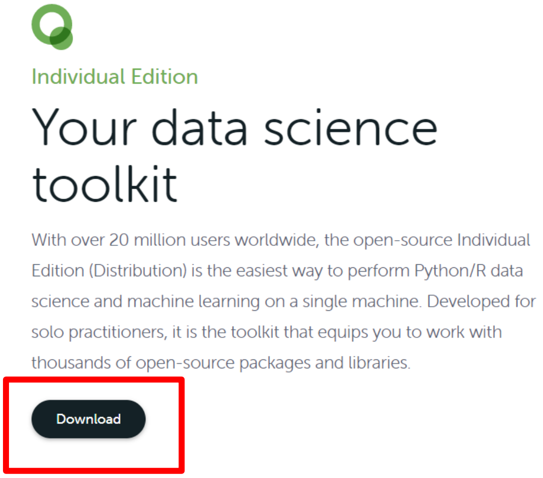


anaconda.org

Step 1.3

Click on ***Download***

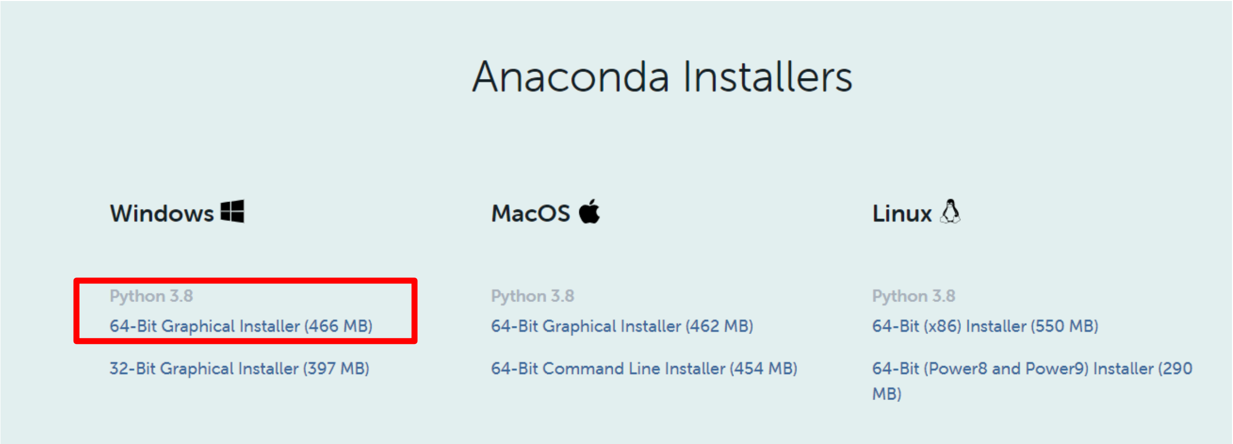
The Individual Edition is Open source and freely distributable



Individual Edition

Your data science toolkit

With over 20 million users worldwide, the open-source Individual Edition (Distribution) is the easiest way to perform Python/R data science and machine learning on a single machine. Developed for solo practitioners, it is the toolkit that equips you to work with thousands of open-source packages and libraries.



Anaconda Installers

Windows	MacOS	Linux
Python 3.8	Python 3.8	Python 3.8
64-Bit Graphical Installer (466 MB)	64-Bit Graphical Installer (462 MB)	64-Bit (x86) Installer (550 MB)
32-Bit Graphical Installer (397 MB)	64-Bit Command Line Installer (454 MB)	64-Bit (Power8 and Power9) Installer (290 MB)

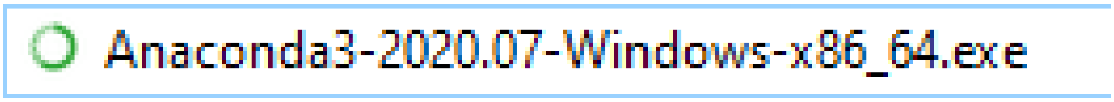
Click on ***64-bit Graphical Installer***

The download task starts.

2. Install the Anaconda package

Step 2.1

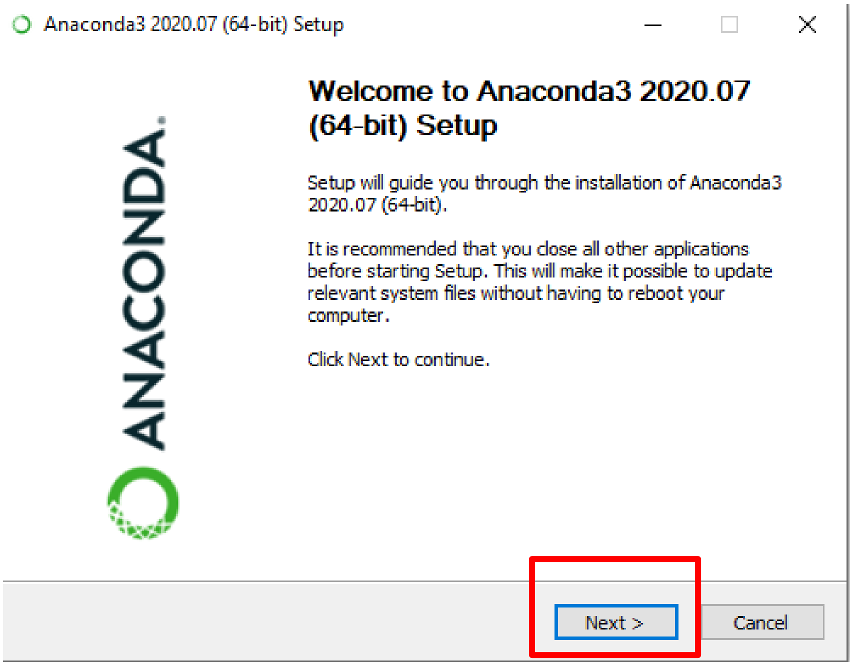
Open the ***Download folder*** and locate the Anaconda3 setup file.



Anaconda3-2020.07-Windows-x86_64.exe

Step 2.2

Run it and press the ***Next*** button on the setup dialog



Anaconda3 2020.07 (64-bit) Setup

Welcome to Anaconda3 2020.07 (64-bit) Setup

Setup will guide you through the installation of Anaconda3 2020.07 (64-bit).

It is recommended that you close all other applications before starting Setup. This will make it possible to update relevant system files without having to reboot your computer.

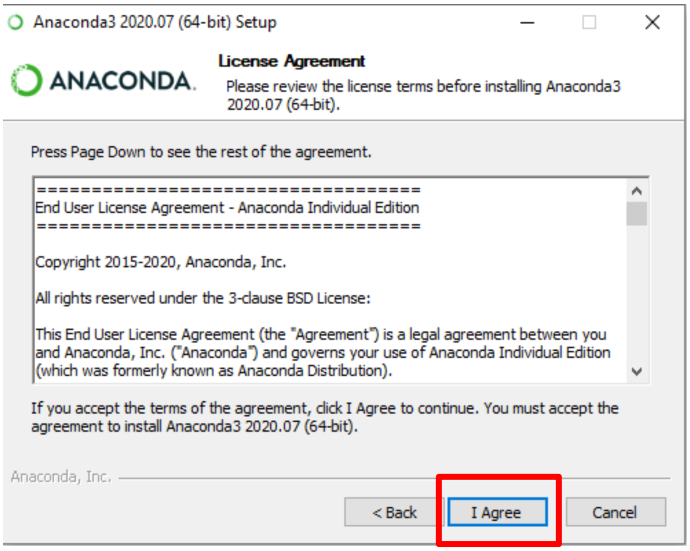
Click Next to continue.

Note :
The Python release must be the 3.8 or higher.

2. Install the Anaconda package

Step 2.3

Click on the „**I Agree“ button** to accept the License Agreement terms.



Anaconda3 2020.07 (64-bit) Setup

License Agreement

Please review the license terms before installing Anaconda3 2020.07 (64-bit).

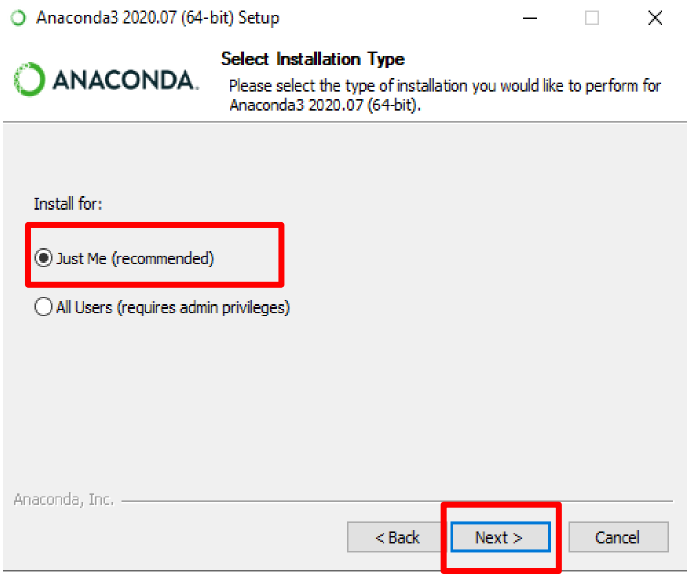
Press Page Down to see the rest of the agreement.

```
===================================
End User License Agreement - Anaconda Individual Edition
===================================

Copyright 2015-2020, Anaconda, Inc.

All rights reserved under the 3-clause BSD License:

This End User License Agreement (the "Agreement") is a legal agreement between you
and Anaconda, Inc. ("Anaconda") and governs your use of Anaconda Individual Edition
(which was formerly known as Anaconda Distribution).
```

If you accept the terms of the agreement, click I Agree to continue. You must accept the agreement to install Anaconda3 2020.07 (64-bit).

Step 2.4

Select the „**Just Me**“ option.



Anaconda3 2020.07 (64-bit) Setup

Select Installation Type

Please select the type of installation you would like to perform for Anaconda3 2020.07 (64-bit).

Install for:

● Just Me (recommended)

○ All Users (requires admin privileges)

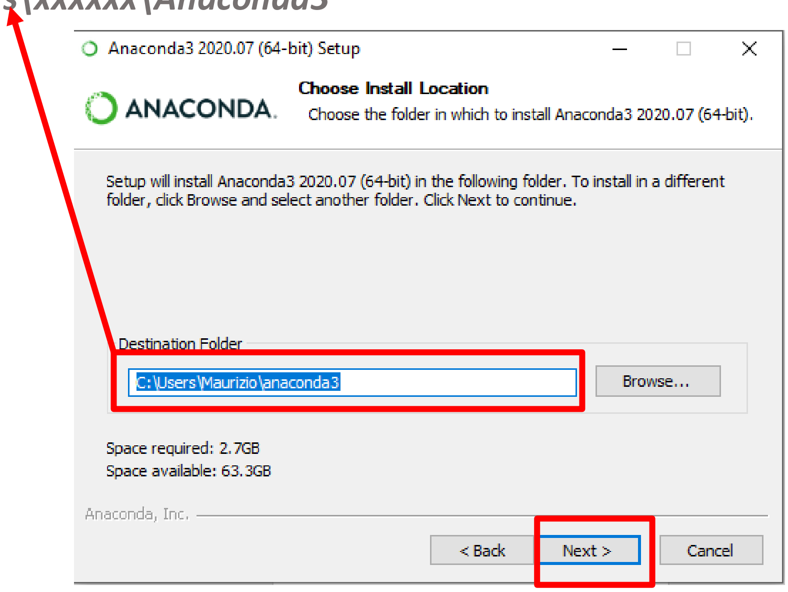
Step 2.5

Press the „Next“ button to complete the installation.

2. Install the Anaconda package

Step 2.6

Leave the default install path. Usually, it is located under the user's directory
C:\Users\xxxxxx\Anaconda3



Anaconda3 2020.07 (64-bit) Setup

Choose Install Location
Choose the folder in which to install Anaconda3 2020.07 (64-bit).

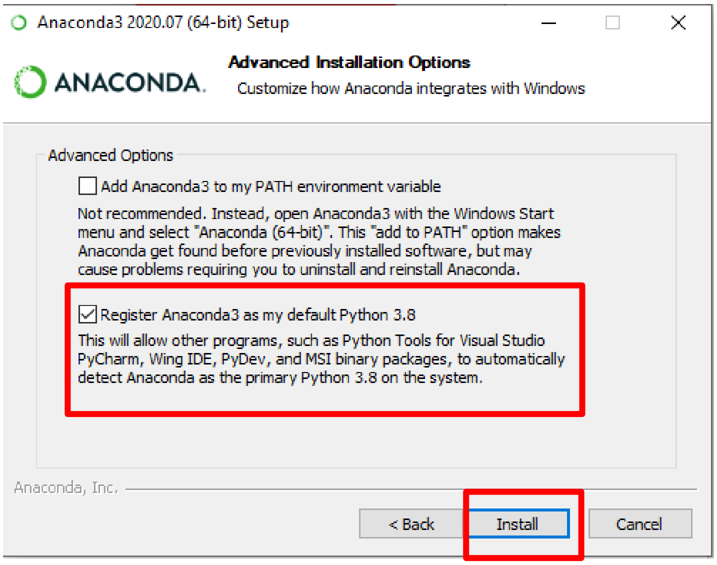
Setup will install Anaconda3 2020.07 (64-bit) in the following folder. To install in a different folder, click Browse and select another folder. Click Next to continue.

Destination Folder: C:\Users\Maurizio\anaconda3

Space required: 2.7GB
Space available: 63.3GB

Step 2.7

Enable the option «Register Anaconda3 as my default Pythoon 3.8» package. Press «***Install***» to start the installation.



Anaconda3 2020.07 (64-bit) Setup

Advanced Installation Options
Customize how Anaconda integrates with Windows

Advanced Options

□ Add Anaconda3 to my PATH environment variable

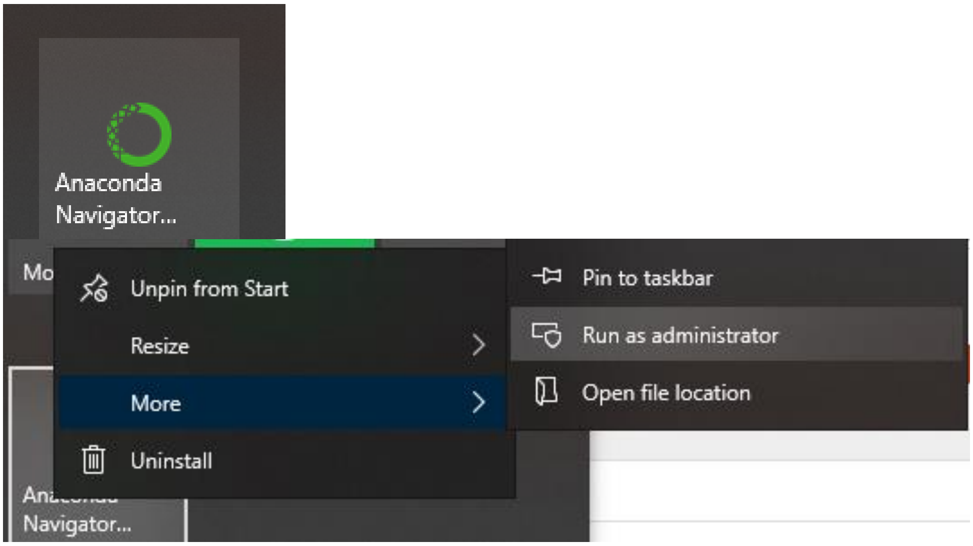
Not recommended. Instead, open Anaconda3 with the Windows Start menu and select "Anaconda (64-bit)". This "add to PATH" option makes Anaconda get found before previously installed software, but may cause problems requiring you to uninstall and reinstall Anaconda.

☑ Register Anaconda3 as my default Python 3.8

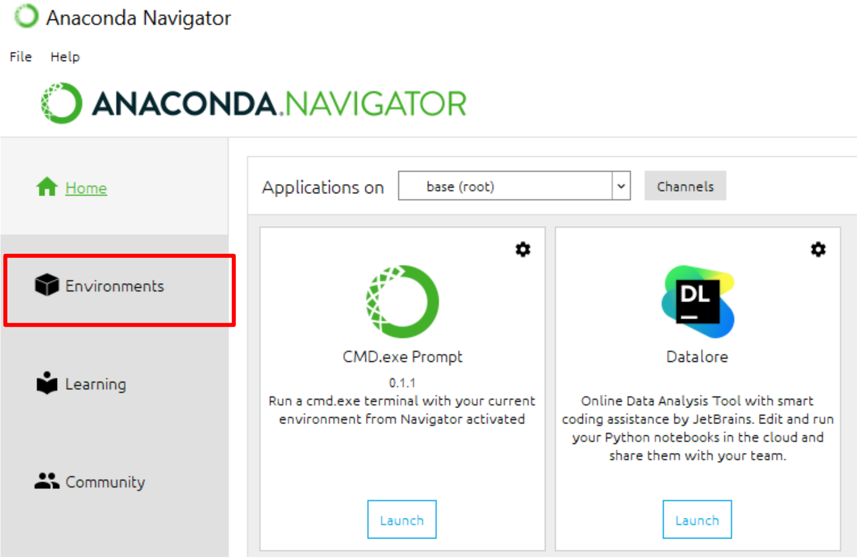
This will allow other programs, such as Python Tools for Visual Studio PyCharm, Wing IDE, PyDev, and MSI binary packages, to automatically detect Anaconda as the primary Python 3.8 on the system.

3. Install the Image registration modules

Step 3.1 Run the Anaconda Navigator in the Administrator account:



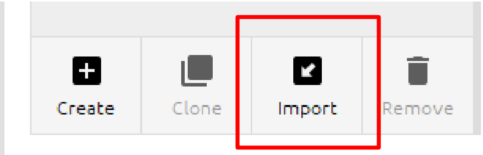
Step 3.2 Select the Environments tab:



Run a cmd.exe terminal with your current environment from Navigator activated

Online Data Analysis Tool with smart coding assistance by JetBrains. Edit and run your Python notebooks in the cloud and share them with your team.

Step 3.3 Select Import the Environment on the bottom :

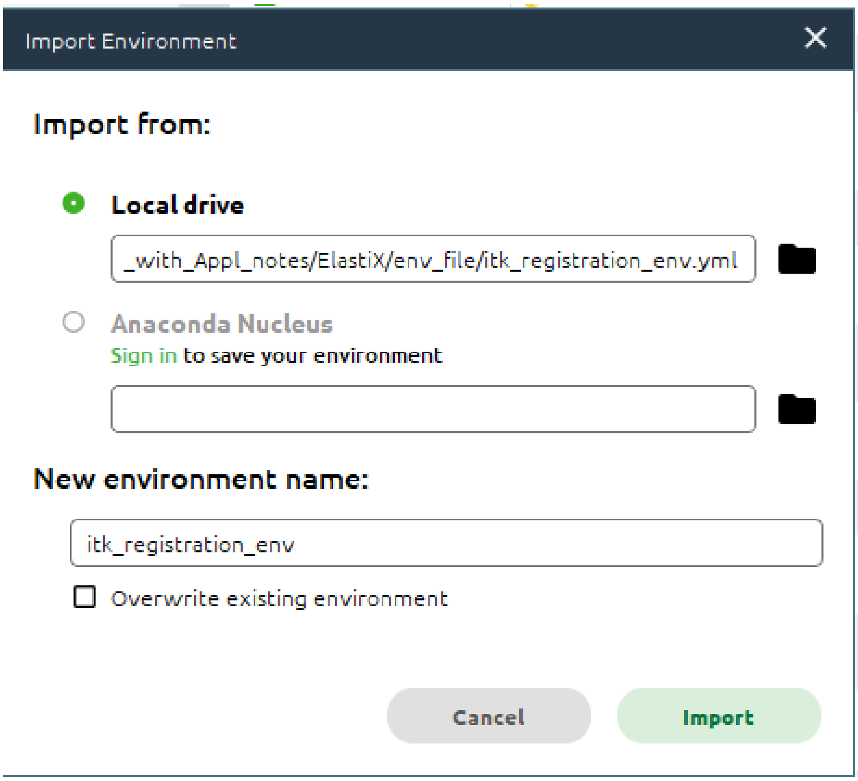


Note :

To install the Cellpose modules, the computer must be connected to internet.

3. Install the Image registration modules

Step 3.4 In the Dialog box, give a name to your new image regostration environment and select the downloaded *itk_registration_env.yml* file and press Import:



Import Environment

Import from:

● Local drive

_with_Appl_notes/ElastiX/env_file/itk_registration_env.yml

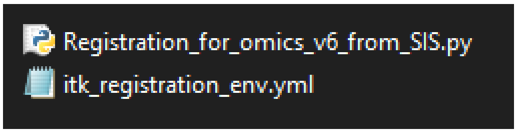
○ Anaconda Nucleus

Sign in to save your environment

New environment name:

itk_registration_env

□ Overwrite existing environment



Registration_for_omics_v6_from_SIS.py

itk_registration_env.yml

Note :

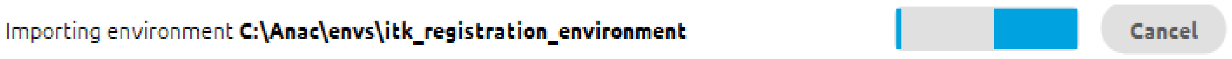
To install the Cellpose modules, the computer must be connected to internet.

3. Install the Image Registration modules

Step 3.3 Continued

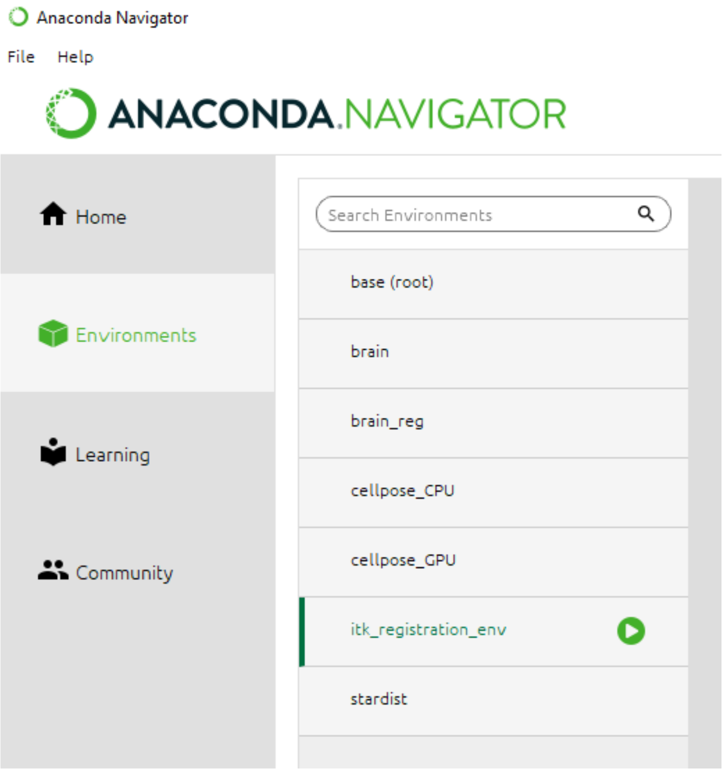
This process can take 5-10 min depending the computer. If the Import button is not green, the given environment likely already exists:

You can monitor the progress on the bottom of the window:



Importing environment **C:\Anac\envs\itk_registration_environment**

The existing environments and the currently importing one can be checked here:

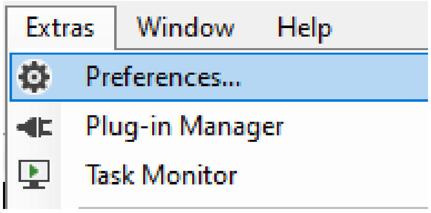


Note :
To install the Cellpose modules, the computer must be connected to internet.
In case of problems with the environment setup, please update to the latest Anaconda Navigator release. Please also check the Troubleshooting section of this Application Note.

4. Vision 4D preferences setup

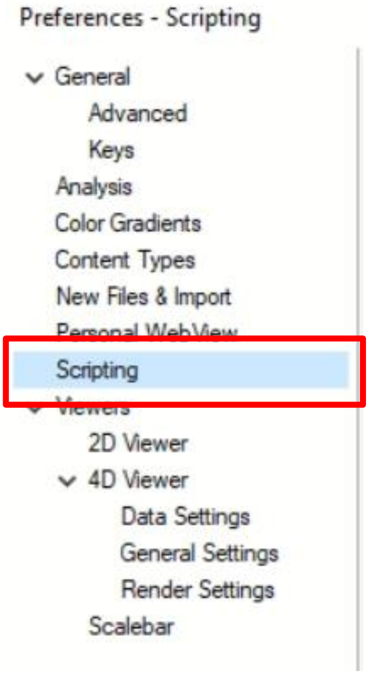
Step 4.1

Run Vision4D (3.4 and next release) select the ***Preferences*** item from the ***Extras*** menu.



Step 4.2

On the left panel, click on the „***Scripting***" item.

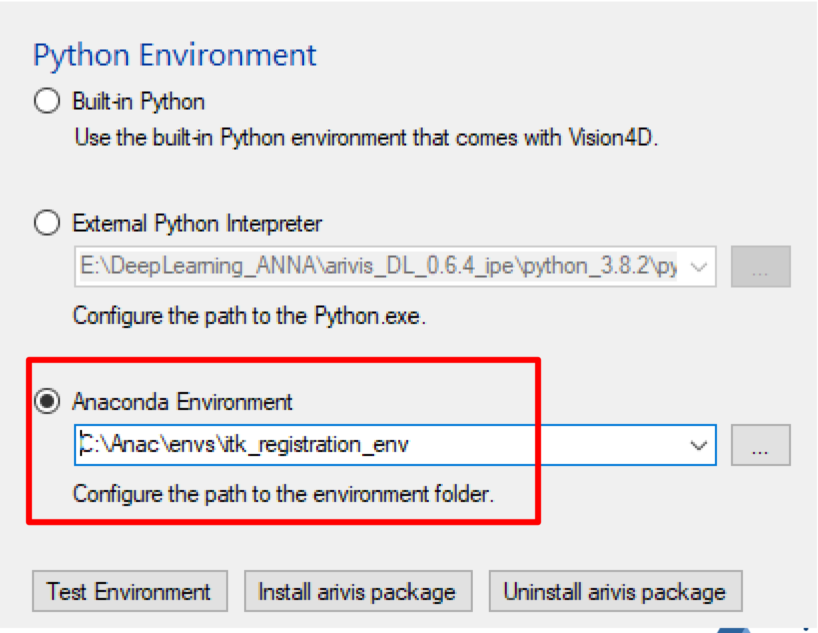


Step 4.3

Enable the "***Anaconda Environment***" option. Browse the Anaconda3 installation folder and select the ***scellpose_v4d*** environment previously created.

By default, the new enviroments are stored under the **\envs** folder located in the Anaconda3 installation folder

e.g.
C:\users\xxxxxx\Anaconda3\envs\scellpose_v4d



Python Environment

○ Built-in Python
Use the built-in Python environment that comes with Vision4D.

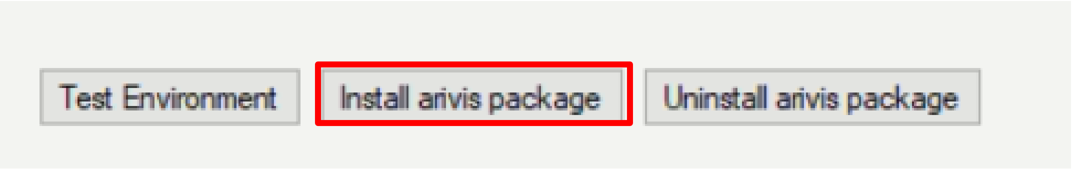
○ External Python Interpreter
E:\DeepLearning_ANNA\arivis_DL_0.6.4_ipe\python_3.8.2\py
Configure the path to the Python.exe.

● Anaconda Environment
C:\Anac\envs\itk_registration_env
Configure the path to the environment folder.

4. V4D preferences setup

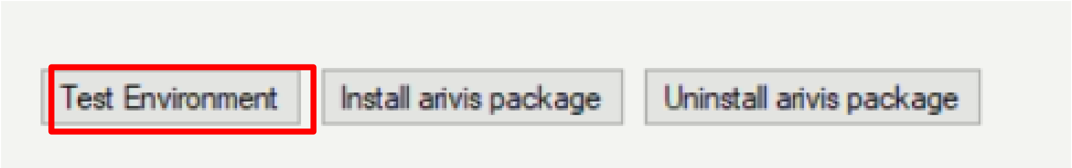
Step 4.4

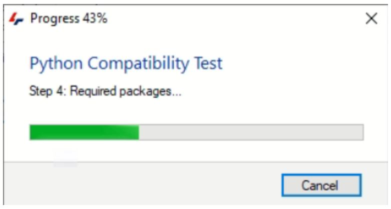
Install the arivis package



Step 4.5

Run the Compatibility test

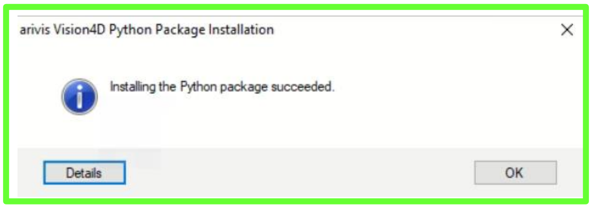




Python Compatibility Test

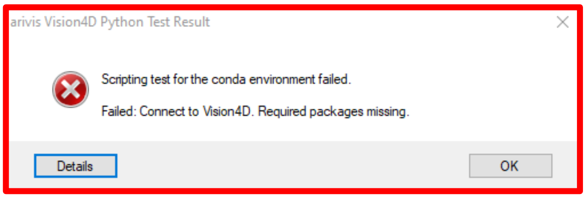
Step 4: Required packages...

Test completed successfully



Installing the Python package succeeded.

Test failed



Scripting test for the conda environment failed.

Failed: Connect to Vision4D. Required packages missing.

Note :
If the left above error message is issued, try again to install the arivis python package.

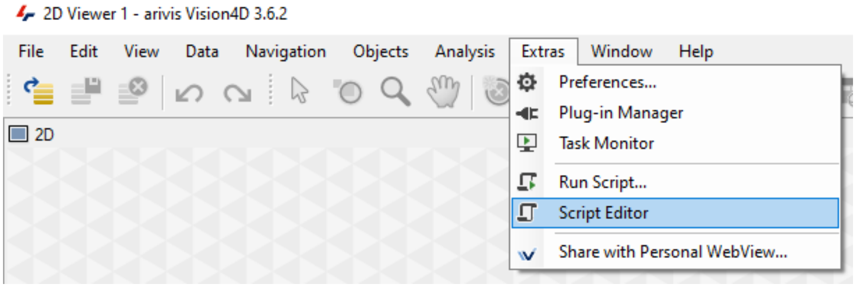
Note :
Apply the settings and close the preferences panel.

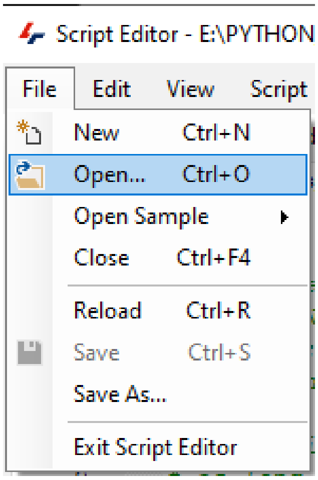
5. Run the Image Registration script in V4D

In order to run the image registration script, a V4D window must be opened.

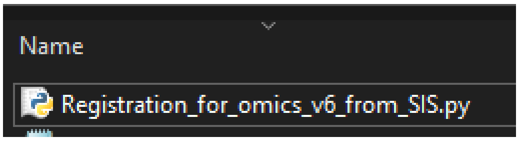
- This could be a new window with no imageset open. In this case the new SIS file will be created in the same folder with the czi files are stored.
- If the SIS file is already open/created the new imagesets will be imported into the existing SIS file. If theexisting imageset names are identical, they will be overwritten.

To run the image registration script, the Script Editor must be open in Vision4D:

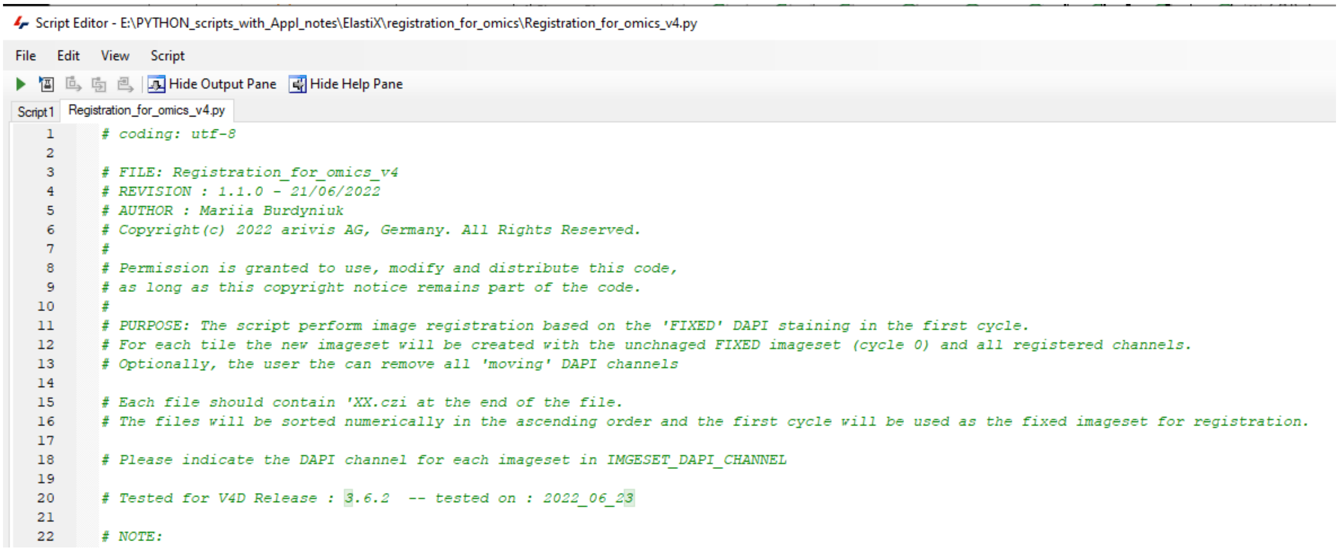




Please open the Registration_for_omics_v4.py file from the Script Editor:



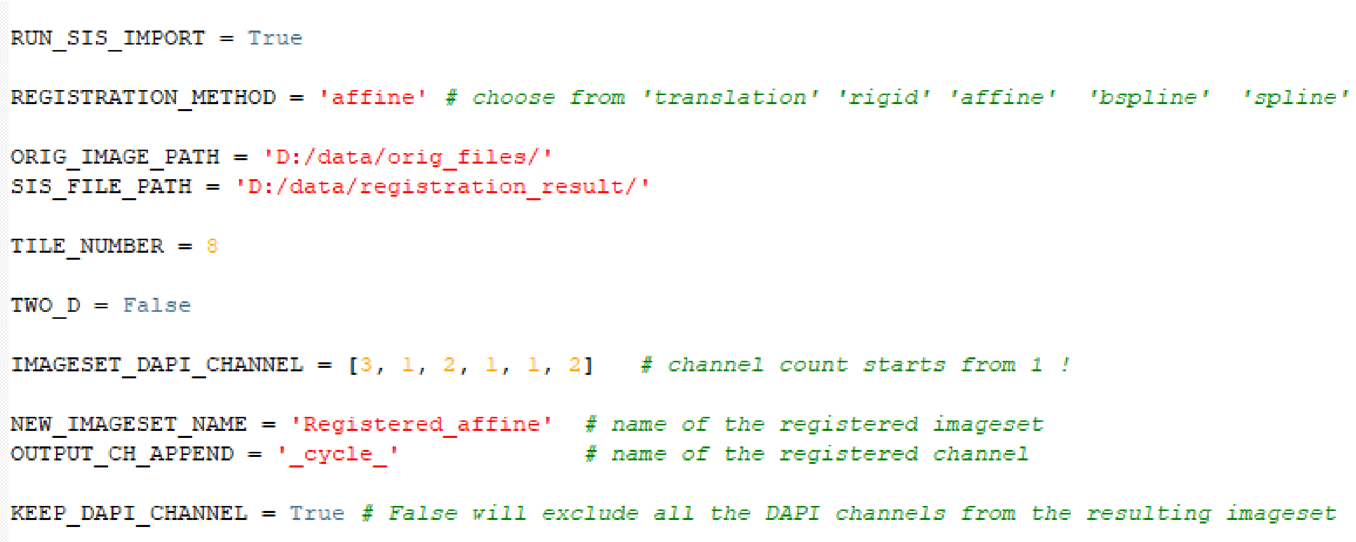
The summary of the script functions is provided on top of the script:


Script Editor - E:\PYTHON_scripts_with_Appl_notes\ElastiX\registration_for_omics\Registration_for_omics_v4.py
File
Edit
View
Script
Hide Output Pane
Hide Help Pane
Script 1
Registration_for_omics_v4.py

```
# coding: utf-8

# FILE: Registration_for_omics_v4
# REVISION : 1.1.0 - 21/06/2022
# AUTHOR : Mariia Burdyniuk
# Copyright(c) 2022 arivis AG, Germany. All Rights Reserved.
#
# Permission is granted to use, modify and distribute this code,
# as long as this copyright notice remains part of the code.
#
# PURPOSE: The script perform image registration based on the 'FIXED' DAPI staining in the first cycle.
# For each tile the new imageset will be created with the unchnaged FIXED imageset (cycle 0) and all registered channels.
# Optionally, the user the can remove all 'moving' DAPI channels

# Each file should contain 'XX.czi at the end of the file.
# The files will be sorted numerically in the ascending order and the first cycle will be used as the fixed imageset for registration.

# Please indicate the DAPI channel for each imageset in IMGESET_DAPI_CHANNEL

# Tested for V4D Release : 3.6.2  -- tested on : 2022_06_23

# NOTE:
```

5. Run the Image Registration script in V4D

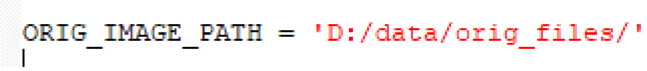
Several script variables must be customized by the customer:



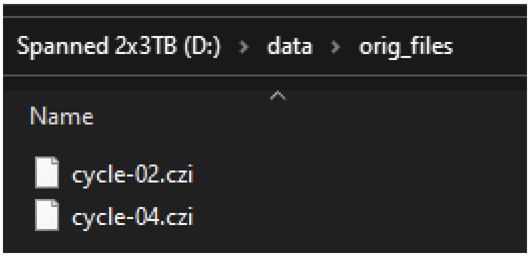
```
RUN_SIS_IMPORT = True

REGISTRATION_METHOD = 'affine' # choose from 'translation' 'rigid' 'affine'  'bspline'  'spline'

ORIG_IMAGE_PATH = 'D:/data/orig_files/'
SIS_FILE_PATH = 'D:/data/registration_result/'

TILE_NUMBER = 8

TWO_D = False

IMAGESET_DAPI_CHANNEL = [3, 1, 2, 1, 1, 2]   # channel count starts from 1 !

NEW_IMAGESET_NAME = 'Registered_affine'  # name of the registered imageset
OUTPUT_CH_APPEND = '_cycle_'              # name of the registered channel

KEEP_DAPI_CHANNEL = True # False will exclude all the DAPI channels from the resulting imageset
```

1. Path to the original files must be provided:

Important: Each image file must have ***'-00'*** (a dash symbol and a number) and the end of the filename. The file with the lowest number will be the *fixed* imageset and all the following images will be registered to this first imageset. The folder should have at least two images to run the registration on.



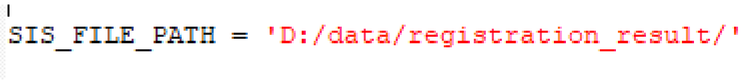
```
ORIG_IMAGE_PATH = 'D:/data/orig_files/'
```



Spanned 2x3TB (D:) › data › orig_files

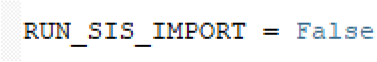
Name
- cycle-02.czi
- cycle-04.czi

2. We need to provide the path to the folder where the imported and the resulting SIS file will be stored:



```
SIS_FILE_PATH = 'D:/data/registration_result/'
```

Alternatively, the file import can be run earlier. The SIS files for each of the cycles must be opened in separate Vision4D windows at the same time. If the imports were run earlier, we need to indicate that



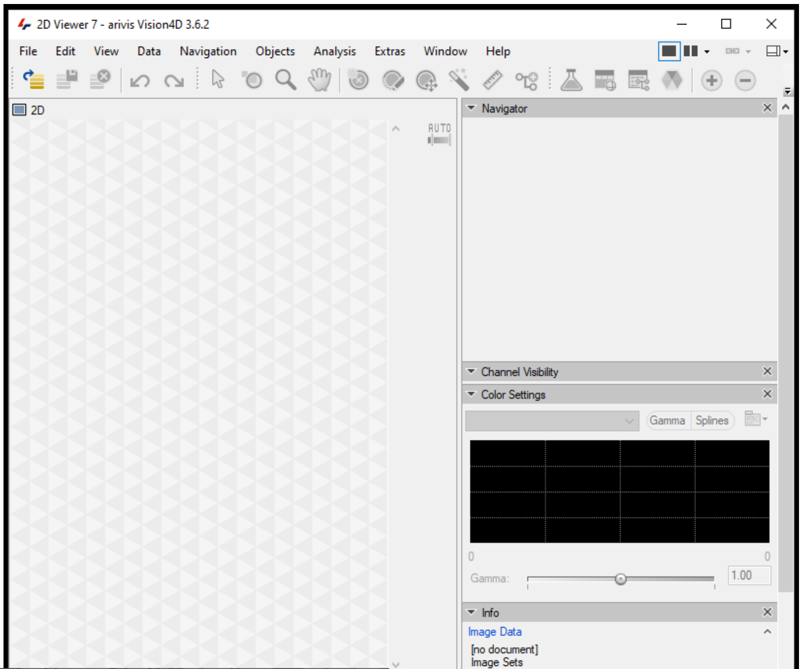
```
RUN_SIS_IMPORT = False
```

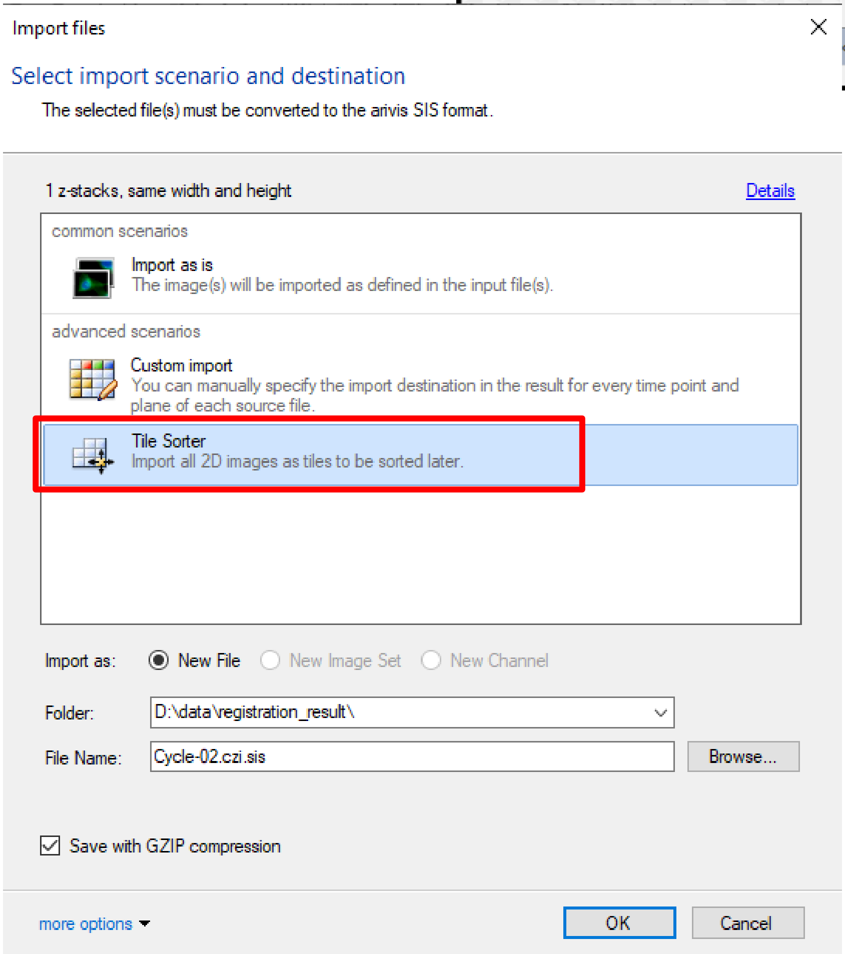
5. Run the Image Registration script in V4D

If the import is run from the script as the first part of the workflow,

```
RUN_SIS_IMPORT = True
```

the empty Vision4D window should be opened. The Import Dialog window will appear. We need to run the import using the Tile sorter option, but do not start the stitching process.





Import files

Select import scenario and destination

The selected file(s) must be converted to the arivis SIS format.

1 z-stacks, same width and height — Details

common scenarios

- Import as is
  The image(s) will be imported as defined in the input file(s).

advanced scenarios

- Custom import
  You can manually specify the import destination in the result for every time point and plane of each source file.
- Tile Sorter
  Import all 2D images as tiles to be sorted later.

Import as: ● New File ○ New Image Set ○ New Channel

Folder: D:\data\registration_result\

File Name: Cycle-02.czi.sis

Browse...

☑ Save with GZIP compression

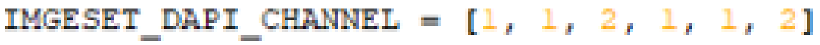
more options ▾

OK Cancel

Important: if the SIS file conversion is run earlier, the import should be run the exact same way using the Tile Sorter import scenario.

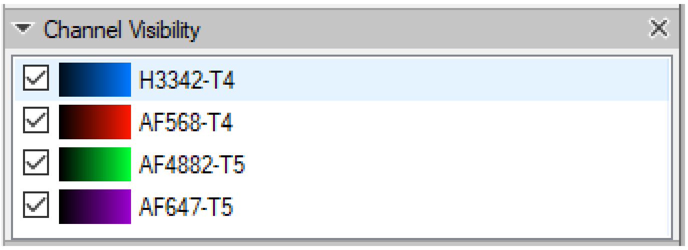
5. Run the Image Registration script in V4D

3. We need to provide the channel with the nuclei staining (i.e. DAPI, Hoechst) for each of the images/cycles in the experiment. The channel count starts from one. The number of the image files in the import folder or open SIS files must be identical or shorter than the number of channels in the IMAGESET_DAPI_CHANNEL list.



```
IMGESET_DAPI_CHANNEL = [1, 1, 2, 1, 1, 2]
```

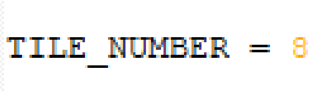
The number of the channel with the nuclei staining (for registration) can be checked in the working panel:



Channel Visibility

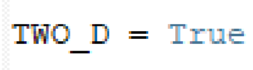
☑ H3342-T4
☑ AF568-T4
☑ AF4882-T5
☑ AF647-T5

4. The tile number must be identical for every image stack in the experiment, as well as the Z spacing. The number of planes in each cycle image may differ. The registration result will have the same number of planes as the fixed ImageSet (cycle_01).



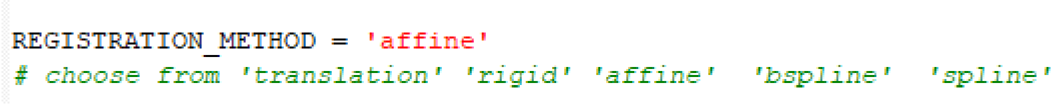
```
TILE_NUMBER = 8
```

4. The workflow is applicable both on 2D and 3D image stacks. For the 3D images, the thickness of the single tile stack is computed based on the total number of planes in the stack divided by the number of tiles, provided by the customer. For the 2D images, please indicate:



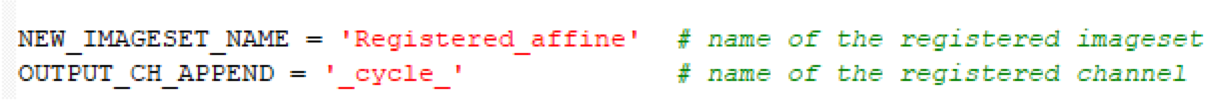
```
TWO_D = True
```

5. Please indicate the registration method:



```
REGISTRATION_METHOD = 'affine'
# choose from 'translation' 'rigid' 'affine'  'bspline'  'spline'
```

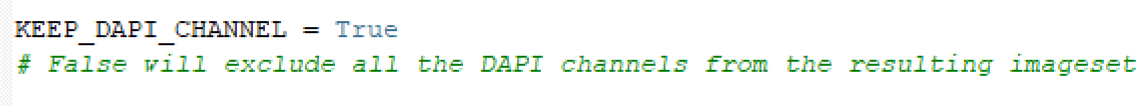
as well as the name of the resulting SIS file and the registered channel names:



```
NEW_IMAGESET_NAME = 'Registered_affine'  # name of the registered imageset
OUTPUT_CH_APPEND = '_cycle_'             # name of the registered channel
```

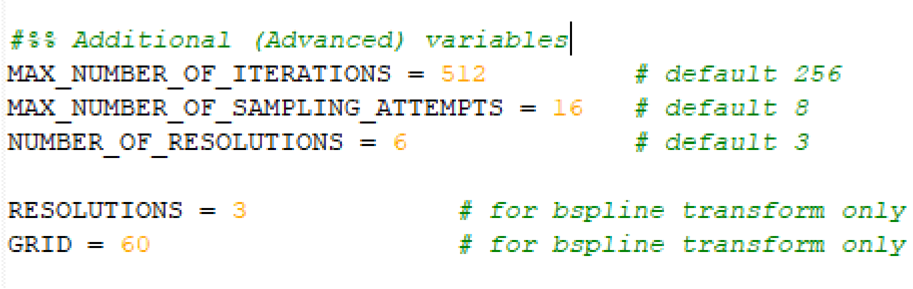
5. Run the Image Registration script in V4D

6. We may choose to exclude all the subsequent nuclei staining channels from the final imported file. If KEEP_DAPI_CHANNEL is set to False, we will only import the nuclei staining channel from the first image/cycle:



```
KEEP_DAPI_CHANNEL = True
# False will exclude all the DAPI channels from the resulting imageset
```

7. The script includes some advanced settings, which enable parametrizing the registration algorithm, regardless of the method used (rigid, affine, etc.)



```
#%% Additional (Advanced) variables
MAX_NUMBER_OF_ITERATIONS = 512          # default 256
MAX_NUMBER_OF_SAMPLING_ATTEMPTS = 16    # default 8
NUMBER_OF_RESOLUTIONS = 6               # default 3

RESOLUTIONS = 3                    # for bspline transform only
GRID = 60                          # for bspline transform only
```

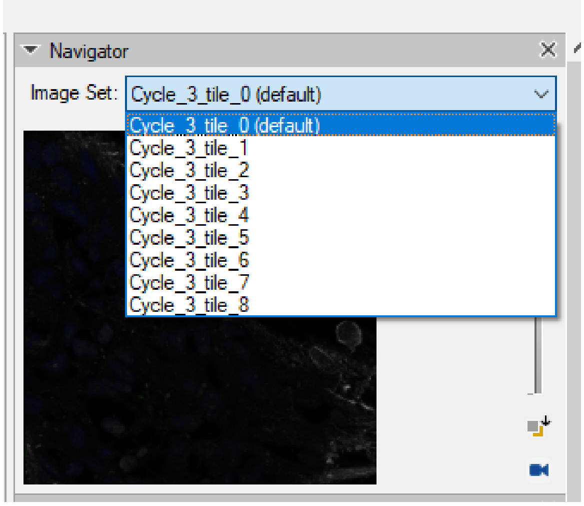
MAX_NUMBER_OF_ITERATIONS: The maximum number of iterations in each resolution. Default/recommended value: 500. When in a hurry, we may go down to 250 for example. When you have plenty of time and want to be absolutely sure of the best results, a setting of 2000 is reasonable. In general, 500 gives satisfactory results.

MAX_NUMBER_OF_SAMPLING_ATTEMPTS: Sometimes not enough corresponding samples can be drawn, upon which an exception is thrown. With this parameter it is possible to try to draw another set of samples. In case the script throws an exception message, please, increase this parameter.

NUMBER_OF_RESOLUTIONS: the number of resolutions used in the multi-step process. The default is 3.

5. Run the Image Registration script in V4D

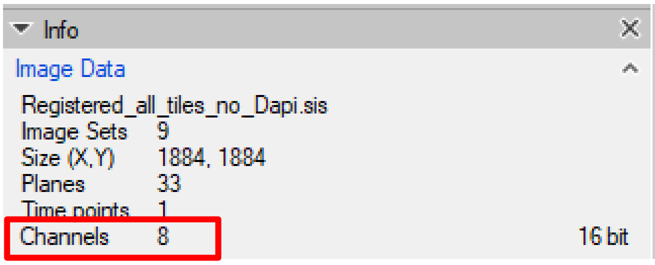
8. Once the registration process is finished, there will be an imageset created for each tile separatelly containing all the channles from all experimental cycles:



Navigator

Image Set: Cycle_3_tile_0 (default)

- Cycle_3_tile_0 (default)
- Cycle_3_tile_1
- Cycle_3_tile_2
- Cycle_3_tile_3
- Cycle_3_tile_4
- Cycle_3_tile_5
- Cycle_3_tile_6
- Cycle_3_tile_7
- Cycle_3_tile_8



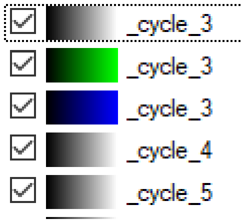
Info

Image Data

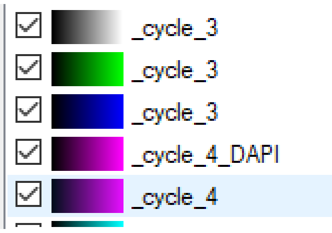
Registered_all_tiles_no_Dapi.sis

Image Sets	9	
Size (X,Y)	1884, 1884	
Planes	33	
Time points	1	
Channels	8	16 bit

Each channel will be labeled according to the cycle in the experiment. The DAPI channels, if the customer decided to import them, will the renamed accordingly.

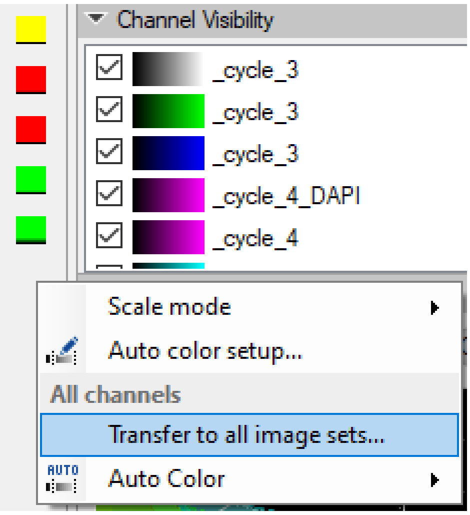


☑ _cycle_3
☑ _cycle_3
☑ _cycle_3
☑ _cycle_4
☑ _cycle_5



☑ _cycle_3
☑ _cycle_3
☑ _cycle_3
☑ _cycle_4_DAPI
☑ _cycle_4

To set the same color settings for all imagesets prior to stitching them in the Tile Sorter, we need to set the preferences for one imageset and apply them on the rest of the images:

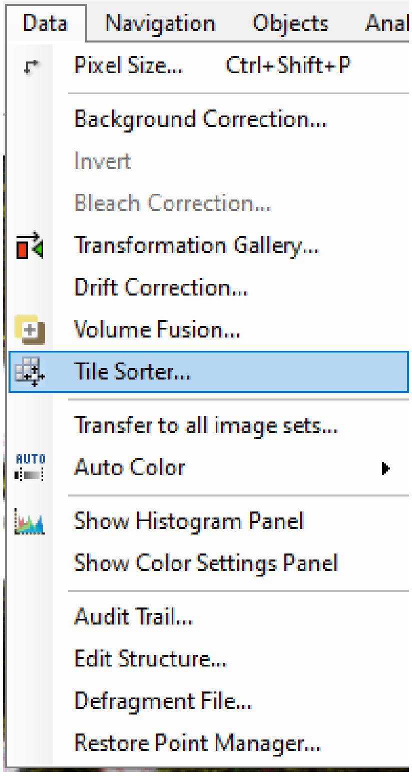


Channel Visibility

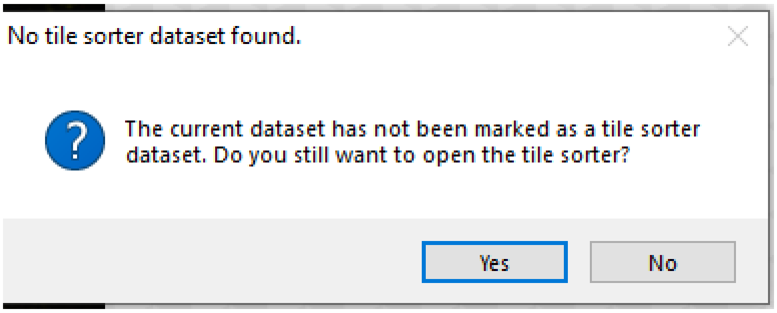
☑ _cycle_3
☑ _cycle_3
☑ _cycle_3
☑ _cycle_4_DAPI
☑ _cycle_4

6. Stitch the mosaic image in Tile Sorter

Image stacks will all the registered channels from the imaging cycles are imported separately and need stitching in the following step. The final image will be created from the individual imagesets all imported into one resulting SIS file.
First, we need to call the Tile Sorter:

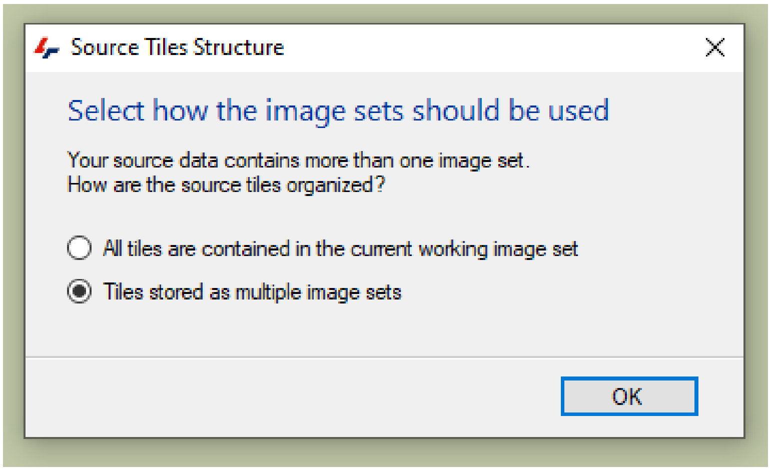


The following massage will appear:


No tile sorter dataset found.

The current dataset has not been marked as a tile sorter dataset. Do you still want to open the tile sorter?

The following massage will appear:

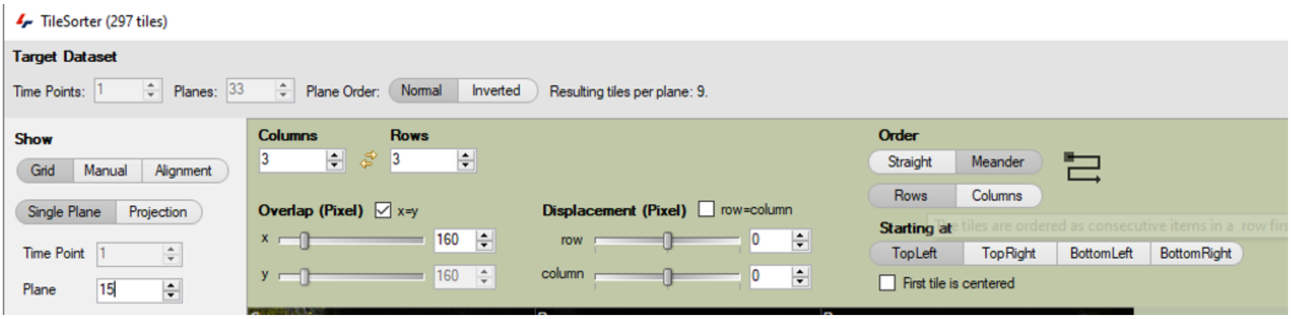

Source Tiles Structure

Select how the image sets should be used

Your source data contains more than one image set.
How are the source tiles organized?

○ All tiles are contained in the current working image set
● Tiles stored as multiple image sets

In the Tile Sorter menu, we can adjust the number of raws and columns, or their order:


TileSorter (297 tiles)

Target Dataset

Time Points: 1 Planes: 33 Plane Order: Normal Inverted Resulting tiles per plane: 9.

Show
Time Point 1
Plane 15

Columns 3 Rows 3

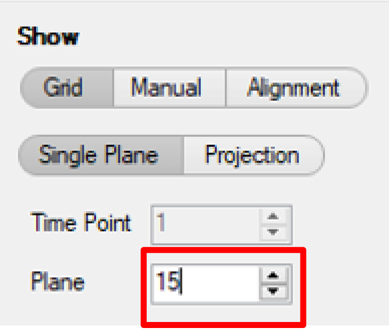
Overlap (Pixel) ☑ x=y
x 160
y 160

Displacement (Pixel) □ row=column
row 0
column 0

Order
Starting at
□ First tile is centered

6. Stitch the mosaic image in Tile Sorter

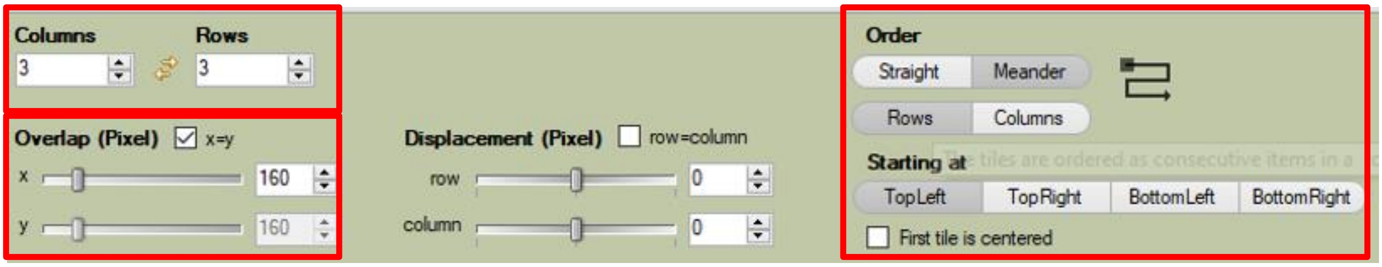
In the Tile Sorter menu, we can first arrange the tile positions as a grid:



Time Point: 1
Plane: 15

TIP: In addition, it is useful to go to the middle plane of the 3D stack or make a Maximal projection.

We can adjust the number of raws and columns, or their order, as well as the number of overlapping pixels:



Columns: 3 Rows: 3

Overlap (Pixel) ☑ x=y
x: 160
y: 160

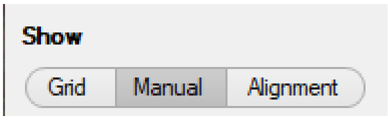
Displacement (Pixel) □ row=column
row: 0
column: 0

Order

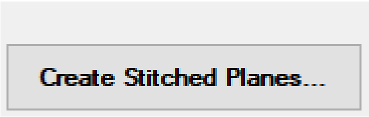
Starting at

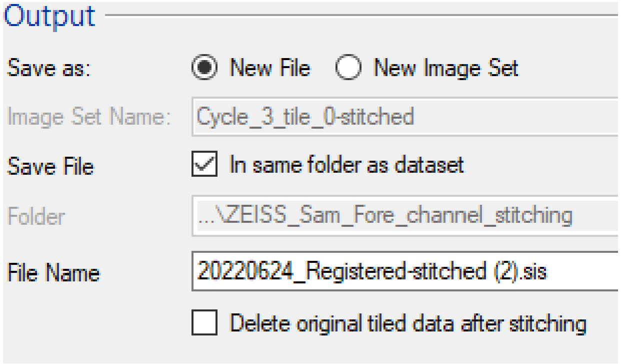
□ First tile is centered

Next, the positions of the tiles can be adjusted manually:



The resulting stitched image can be imported into the same imageset or as a separate new SIS file.




Output

Save as: ● New File ○ New Image Set

Image Set Name: Cycle_3_tile_0-stitched

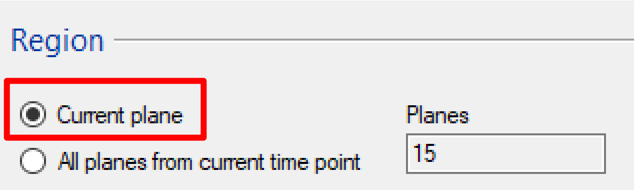
Save File: ☑ In same folder as dataset

Folder: ...\ZEISS_Sam_Fore_channel_stitching

File Name: 20220624_Registered-stitched (2).sis

□ Delete original tiled data after stitching

TIP: For the large file it is often worth first exporting and testing the alignment on a single plane and only stitching the entire 3D stack afterwards.


Region

● Current plane

○ All planes from current time point

Planes: 15

arivis
a ZEISS company

Vision4D

Contact the arivis application support to receive additional technical details about the topic described in the document.

*arivis AG, Am Kabutzenhof 21,
18057 Rostock, Germany*

Email : support@arivis.com