

Contents

Introduction	Error! Bookmark not defined.
Step 1 – Download demo data.....	3
Step 2 – make MVR dataset for beads and co-register dOPM views1&2	3
Step 3 –transformation and coregistration of dOPM view 1&2 bead data	4
Step 4 - Visually inspect coregistration quality bead data	5
Step 5 - Setup MVR dataset for sample data	6
Step 6 – copy transformations in MVR bead dataset to sample dataset	7
Step 7 – resave the MVR sample dataset as ‘.hdf5’ format for viewing with ‘Bigdataviewer’ to check quality of coregistration between dOPM view 1&2	8
Step 8 - Visually inspect coregistration quality of sample data	10
Step 9 - Define a bounding box to crop out a sub-volume, optional step.....	11
Step 10 - Extract processed single view or fused datasets for generic downstream processing of 3D datasets.....	13

Introduction

Code to process data generated by dOPM microscopes as of 2023. DOPM paper:

<https://doi.org/10.1364/BOE.409781>

Sharing dOPM data processing scripts based in ImageJ and relying mainly on Multiview Reconstruction (MVR) plugin <https://imagej.net/plugins/multiview-reconstruction>

Following steps in this guide outline procedures for deskewing and fusion of raw dOPM data. Another one for deconvolution operations will be made soon.

dOPM acquires two overlapping volumes in the sample from two views we call 'view 1' and 'view 2' – see paper for details. To process the data, we assume the user has acquired both views and they want to fuse the two views or extract single views. The raw data is skewed so we transform the data with affine transformations to rotate and deskew the raw data into the microscope coordinate frame. We then do bead-based image registration using a dataset of beads in 3D acquired with the same scan settings as the sample data. This registration information is then assumed to be valid for other datasets without beads i.e. cells/organoids so we apply all the affine transformations we used on the bead data including the coregistration information to any other datasets acquired during the experiment. Read the MVR paper and the dOPM paper for more information on Multiview reconstruction based on bead datasets. Note there are other ways to register Multiview data such as by using features in the samples such as nuclei or membrane labels that are common to both dOPM views. The benefit of this option is we do not require a bead volume and we might have a better alignment of the data as the assumption that registration information for bead data is same as that needed for sample data is not needed.

Use demo data see 'Step 1' below, to learn the steps in this guide. Demo data includes:

- Sample data – some cells in 3D, folder named 'data'
- Bead data - volume of beads with identical scan settings as sample data, folder named 'beads'

The key processing workflow steps we use with dOPM data (note these bullet points summarise detailed steps guidance on following pages).

1. Download demo bead and sample data volumes and setup MVR dataset only for bead data
2. Use theoretical estimates of rotations and skew of raw data to transform the raw bead data into lab space coordinates
3. Use ImageJ MVR plugin to get coregistration information from the de-skewed and rotated bead data so can align dOPM view 1 volume with view 2 bead volume
4. Visually inspect coregistration quality of the bead data dOPM view 1&2 to check OK
5. Now setup MVR dataset for sample data
6. copy transformations in MVR bead dataset to sample dataset Visually inspect coregistration quality of the sample data dOPM view 1&2 to check OK
7. resave the MVR sample dataset as '.hdf5' format for fast viewing with 'Bigdataviewer' to check quality of coregistration between dOPM view 1&2
8. Visually inspect coregistration quality of sample data
9. Define a bounding box to crop out a sub-volume
10. Extract processed single view or fused datasets – resliced, de-skewed and dOPM view1&2 co-registered data
11. At this point the user has 3D datasets transformed into lab/microscope Cartesian coordinates so can carry out routine 3D image analysis from here on in.

Prerequisites

For deeper understanding of the MVR plugin read the following links, and references within

<https://imagej.net/plugins/multiview-reconstruction>

<https://imagej.net/plugins/bigstitcher/>

<https://forum.image.sc/> - community support, can make an account to post questions

Summary of code

It is based on ImageJ scripts and written in Python within ImageJ's scripting functionality. I used GUI dialog boxes to prompt the user to make decisions that are common to dOPM data processing. Often the dialog boxes that appear are from the MVR plugin. The aim of the code is to convert raw data to resliced data and to use it as single view or fused data. All the steps can be run from a script instead using the MVF batch processing commands. At some point we will share such scripts. The GUI steps in this implementation are intended to be a quick way to use the MVF functionality with little experience of coding needed. However, it is more powerful to use scripts as it is easier to automate steps and run similar processing steps on batches of data.

Step 1 – Download demo data

Installation – follow this exact recipe to ensure it works

Put these scripts in fiji subfolder -> path-to-fiji\Fiji.app\plugins\Scripts\dOPM

Use this version of Fiji (all others not compatible) ->

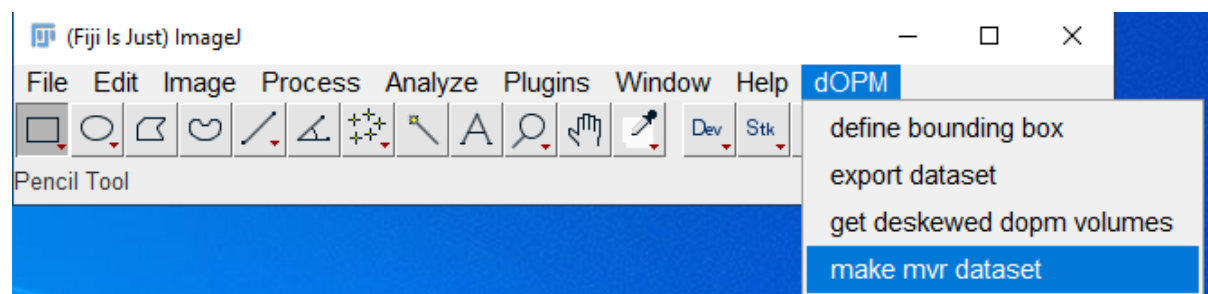
<https://imperialcollegelondon.box.com/s/555qs9ufjrrh8b43ocry4gp4x0yhh36a>

Demo data:

<https://imperialcollegelondon.app.box.com/s/kweg5c8w3r8hhkhuxy80du6binwhlhkv>

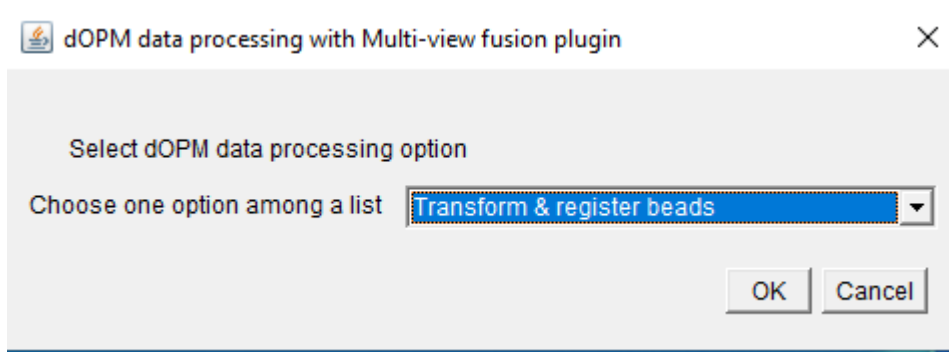
Step 2 – make MVR dataset for beads and co-register dOPM views1&2

- Click on the dOPM link and select 'make mvr dataset'

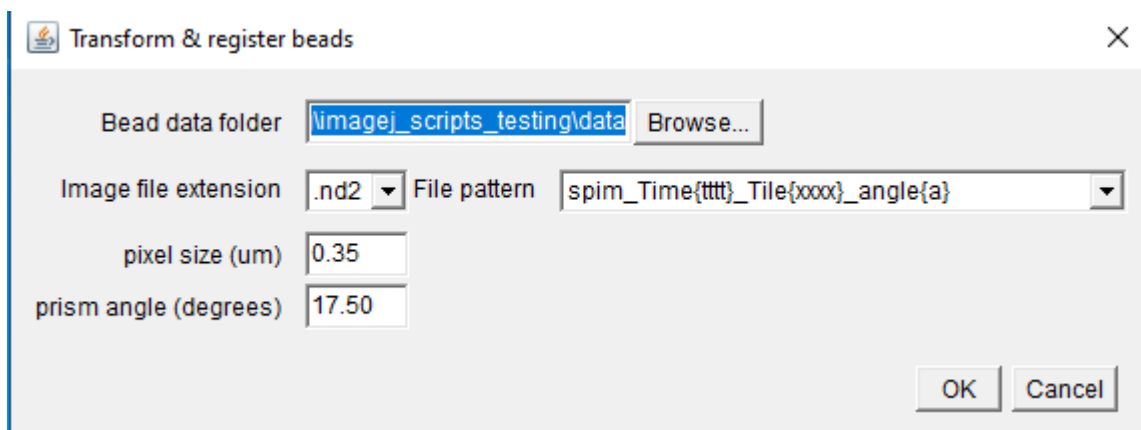


- Next dialog box chooses 'transform and register beads'

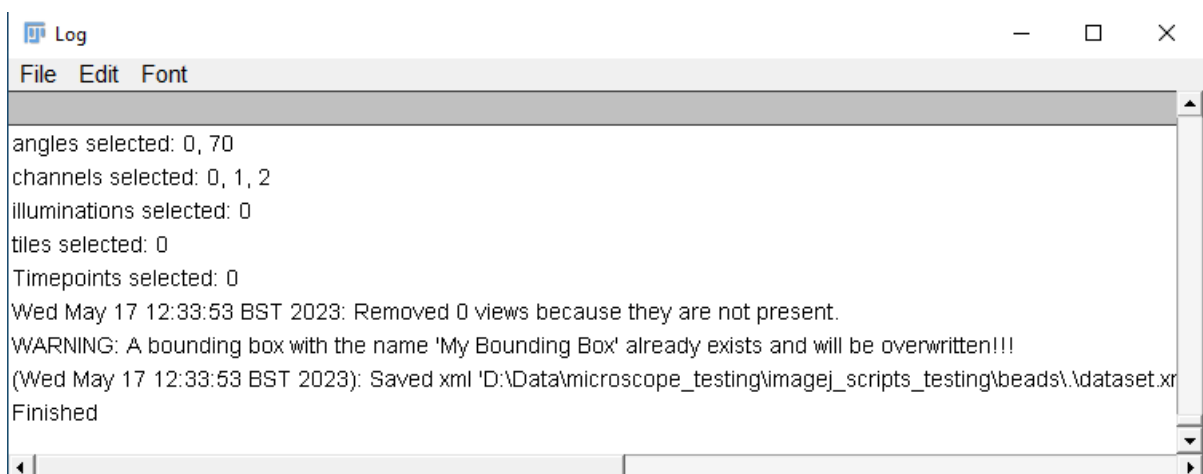
Step 3 –transformation and coregistration of dOPM view 1&2 bead data



- Next dialog box choose folder where bead data folder address
- We are using '.nd2' files for the raw data so leave 'image file extension as '.nd2'
- Leave the file pattern as below
- Leave the pixel size and prism values below, the user needs to keep track of pixel size and prism angle in the specific dOPM system being used – ideally would be in image metadata

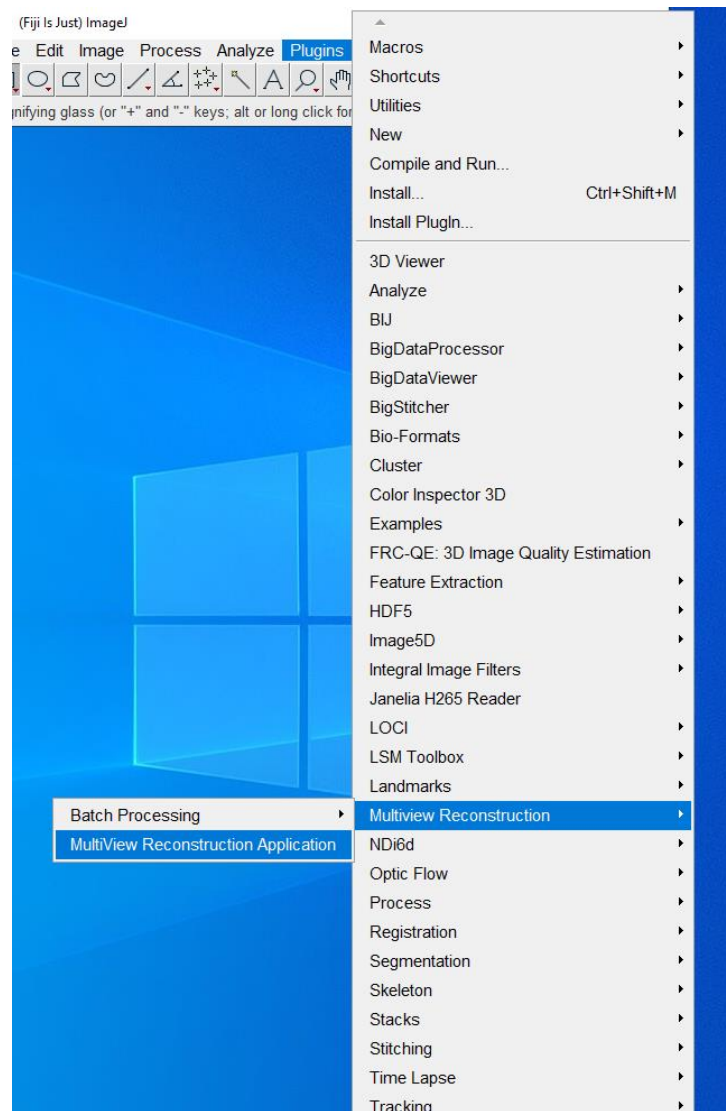


- When 'OK' pressed, the log window will print stuff and will complete with 'Finished' – wait for this. Now the bead dataset is setup.

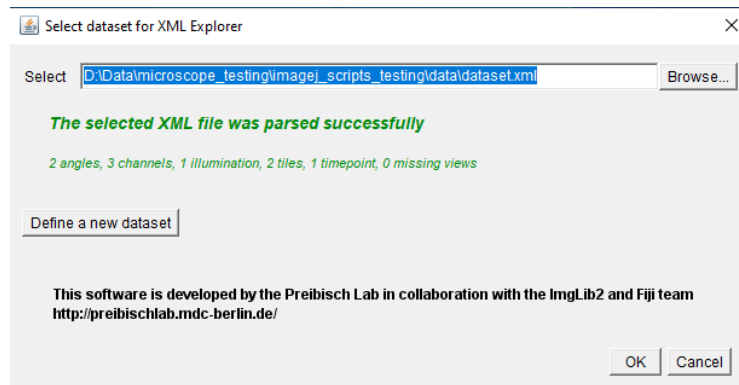


Step 4 - Visually inspect coregistration quality bead data

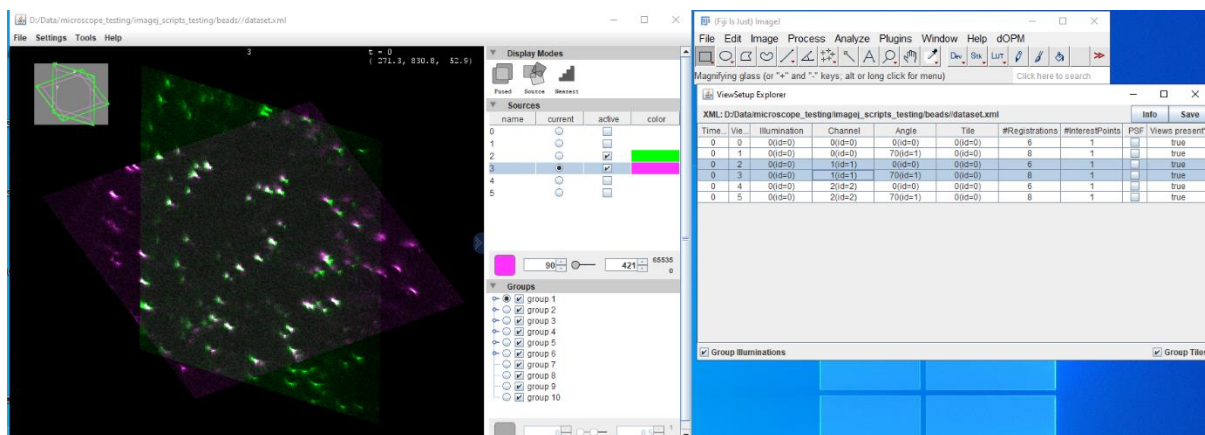
- The bead dataset consists of a dataset based on the raw '.nd2' data and within that there is a version where the raw data has been resaved as '.hdf5' – take a look and read the wiki pages for info on MVF plugin for more information about MVR dataset definition.
- In the next step we will visualise the bead dataset with MVF use of the 'Bigdataviewer' and check by eye if the beads from views 1&2 are visibly overlapping to be judged as coregistered
- Go on the plugins menu and click on the 'Multiview Reconstruction' option then the application option as in the image below



- Look for your bead data folder and within that should be a folder named 'hdf5' i.e. your-path\beads\hdf5\dataset.xml
- Click on this 'xml' file – this file defines the MVR dataset we use for previewing with the 'Bigdataviewer'
- Press OK to explore this dataset with the MVR GUI, the 'Bigdataviewer' tool window of the data should automatically appear – **this automatic opening is because you opened the hdf5 converted dataset.**

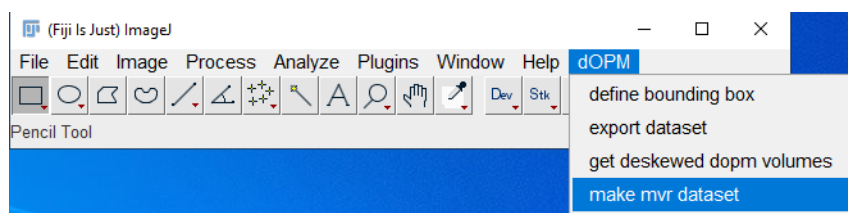


- To get to grips with using MVR application go to the plugin's documentation and help
- The screenshot below shows you an example of using the 'Bigdataviewer' tool within the plugin to visualise the dataset and the two dOPM views for one spectral channel are visualised in green and magenta - read the plugin help documentation for a deeper understanding.
- In the screenshot the two views have clearly been successfully rotated, deskewed and coregistered since the bead's volumes from the two views overlap. The registration process relied on calling some MVR plugin functions in the background of my GUI based code. If the bead data has not been successfully coregistered you need to get help from an expert user or be prepared to start to understand more deeply how to use the MVR plugin.
- it is possible that bead registration fails or there is an error in the way the data is generated. In this case expert help or deeper understanding of the MVR plugin is needed.



Step 5 - Setup MVR dataset for sample data

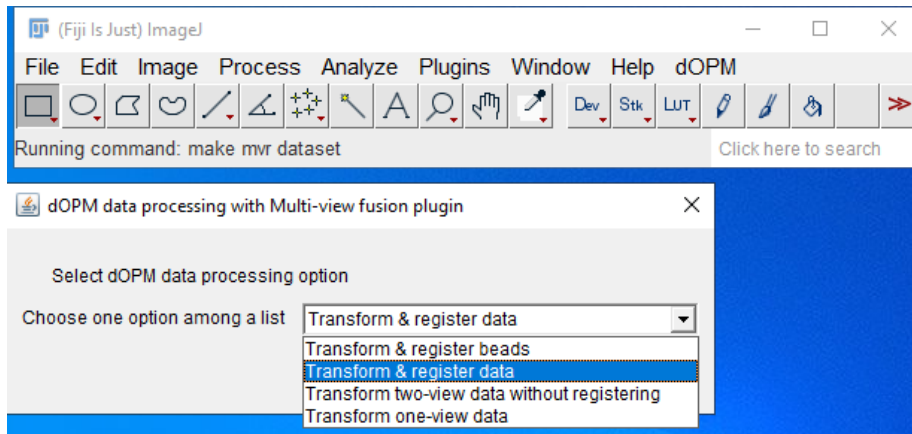
- Now assuming the bead data is successfully transformed we setup the sample data dataset doing a simpler operation to before with beads – we copy over the bead transformations



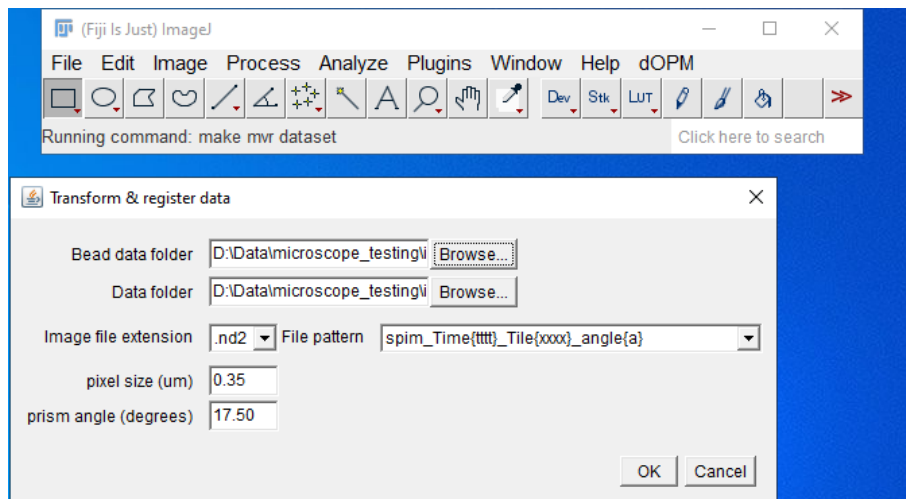
- This step is different to bead dataset (steps 1-4 above), select 'transform and register data'

- This step assumes you have completed the setting up of the bead dataset (steps 1-4 above), steps need to be completed in the order given essentially.

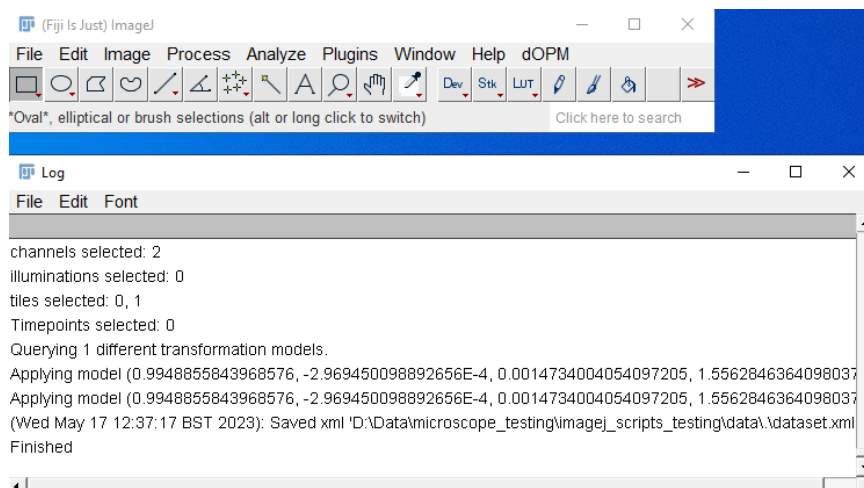
Step 6 – copy transformations in MVR bead dataset to sample dataset



- dialog box requires you to enter the bead data folder and the sample data folder paths
- Keep the pixel size, file pattern prism angle settings the same as the bead data setup process (steps 1-4) and click OK

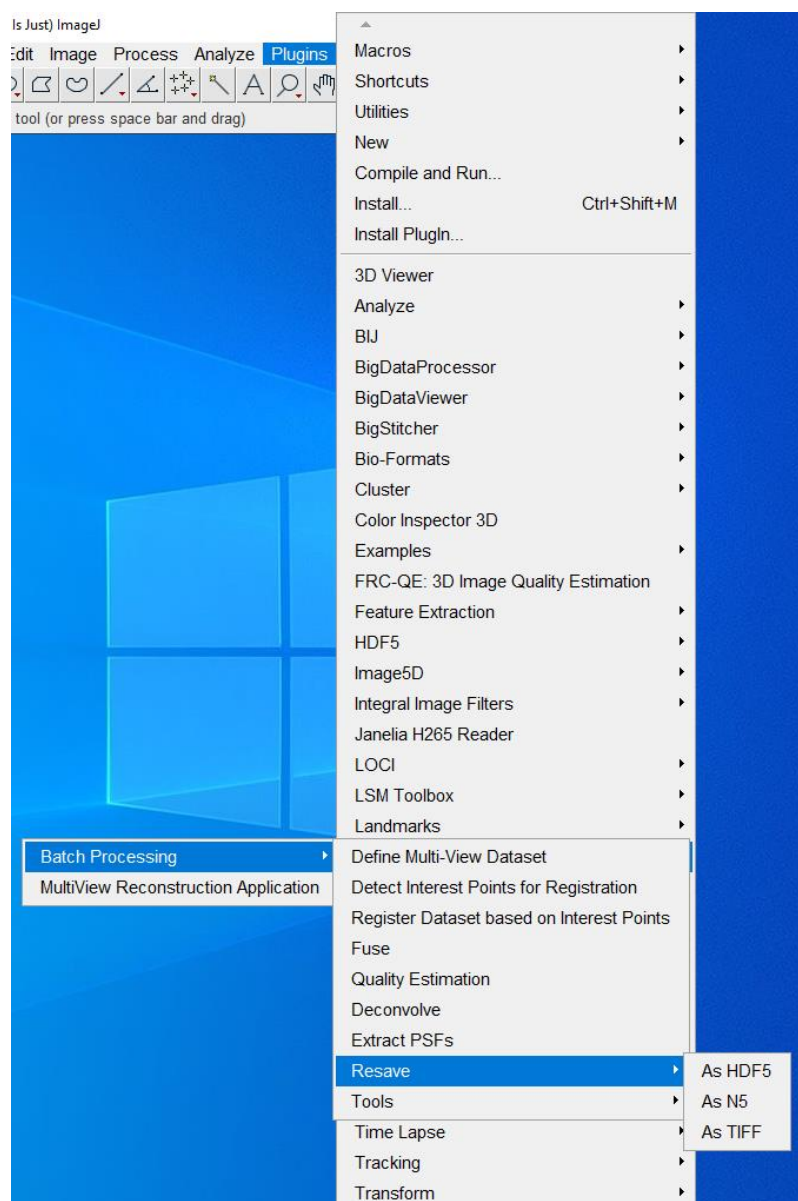


- Wait till the dialog box prints 'Finished'



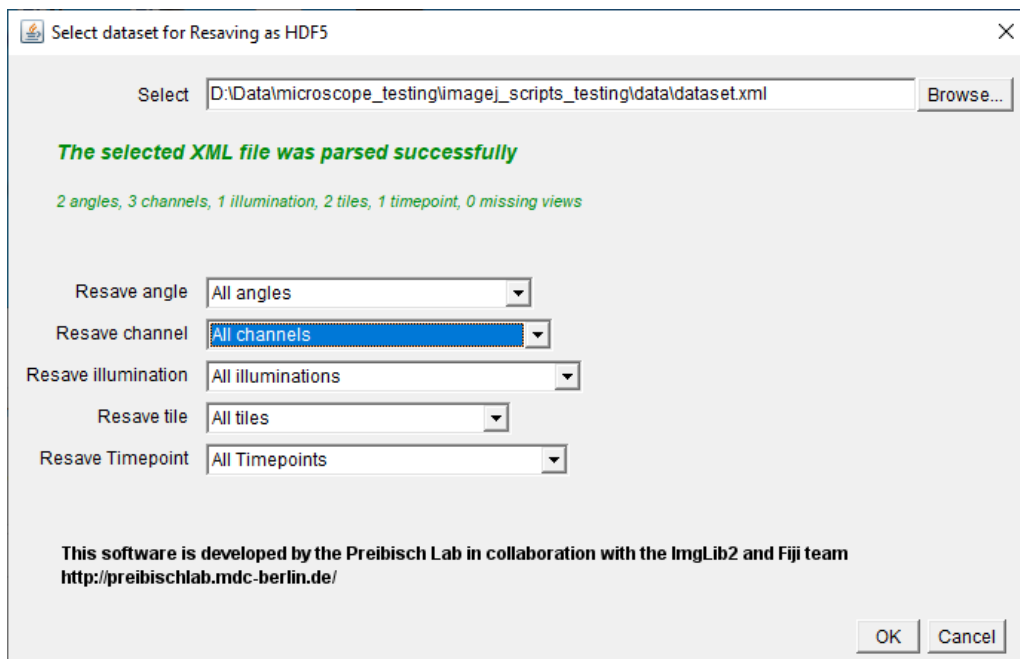
Step 7 – resave the MVR sample dataset as ‘.hdf5’ format for viewing with ‘Bigdataviewer’ to check quality of coregistration between dOPM view 1&2

- For the next step we want to manually resave the data in hdf5 format as this allows for fast visualisation of the data via the MVR application’s use of the ‘Bigdataviewer’ tool – see MVR documentation for details about Bigdataviewer.
- While for the bead data I setup the protocol in the code to automatically transform raw data to hdf5 for fast viewing with Bigdataviewer we do not do this automatically for the sample data. This is because while the bead data will typically be a small, single time point dataset, the sample data could be huge so converting to hdf5 will take a long time to process and fill up the storage space, so this step is not automatic.
- Follow the screenshot below to resave as hdf5

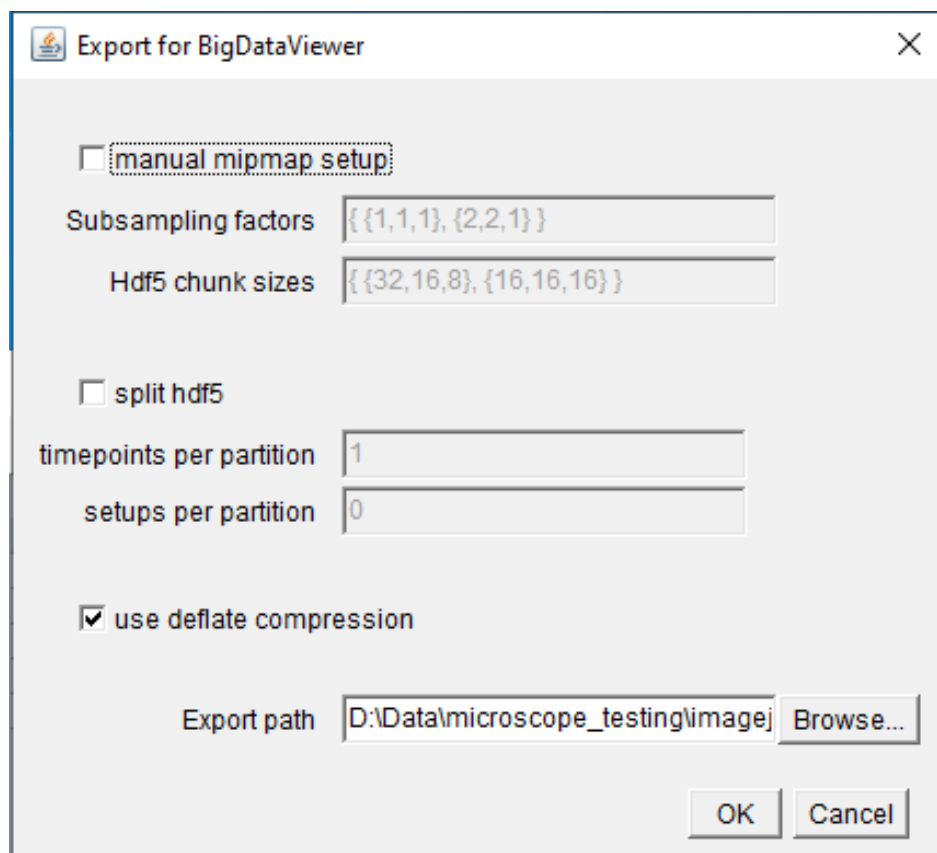


- The next dialog box that appears asks for location of data, go to the data folder, and click on the xml

- Resave for all angles, channels, all dimensions – the entire dataset

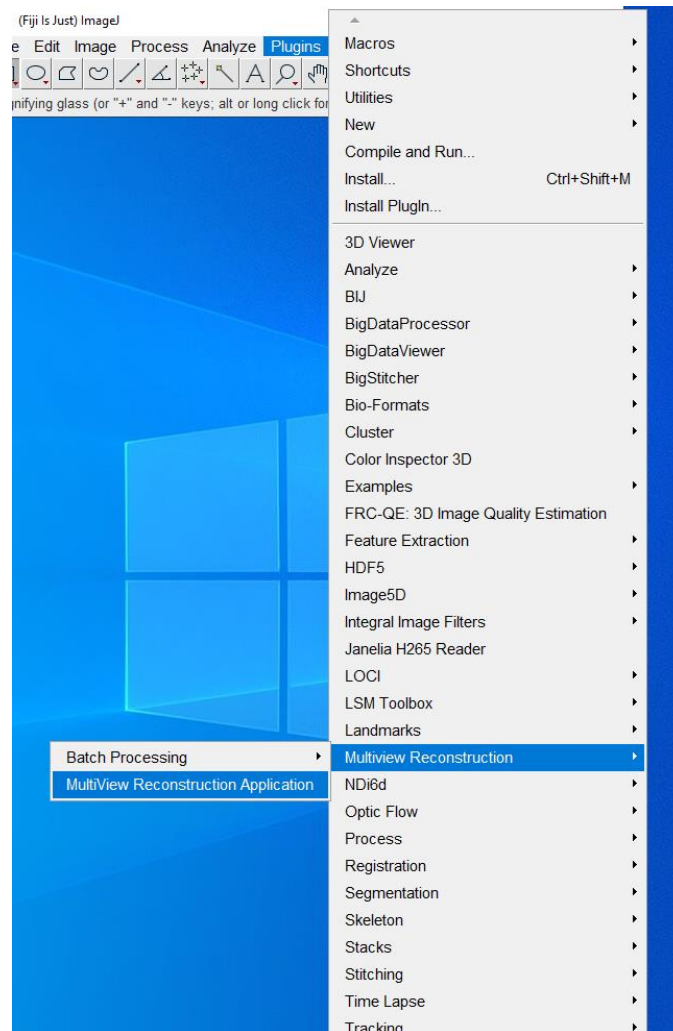


- Next use the default export settings, do not change anything – keep it same as the screenshot below

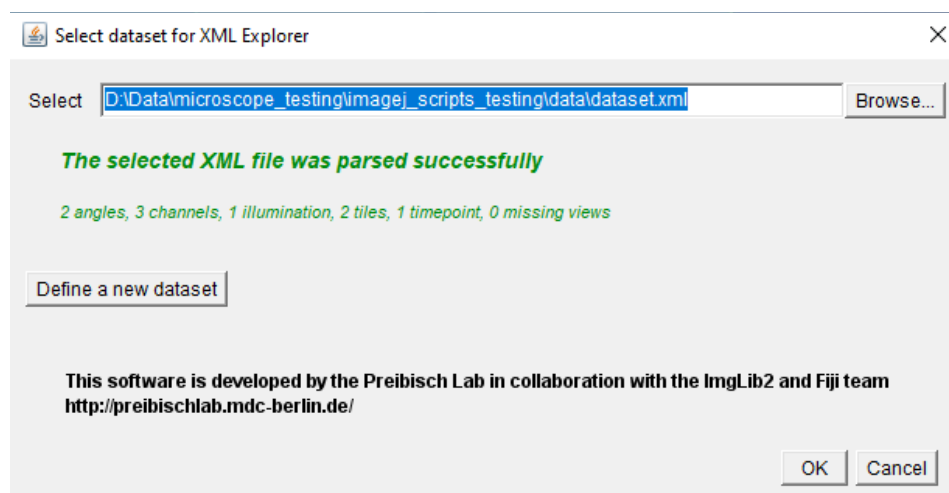


Step 8 - Visually inspect coregistration quality of sample data

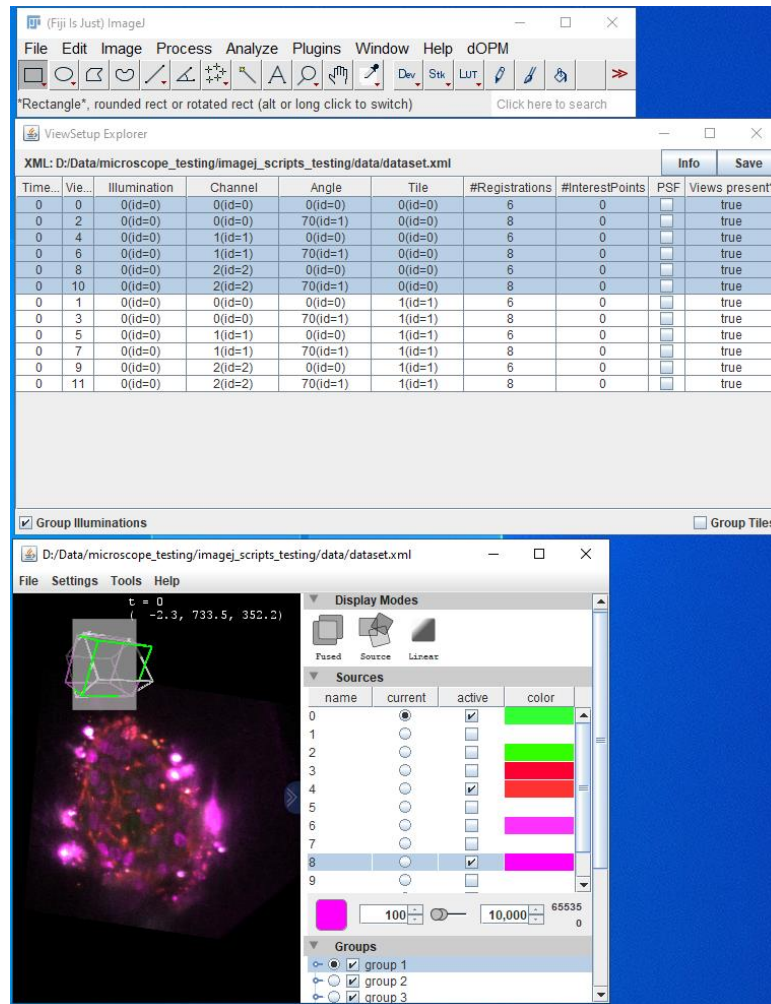
- Now open the dataset as in Step 4, the dataset.xml now contains instructions to expect a hdf5 dataset instead of the raw dataset in the data folder. The data will be automatically shown in the 'Bigdataviewer' tool for visualisation



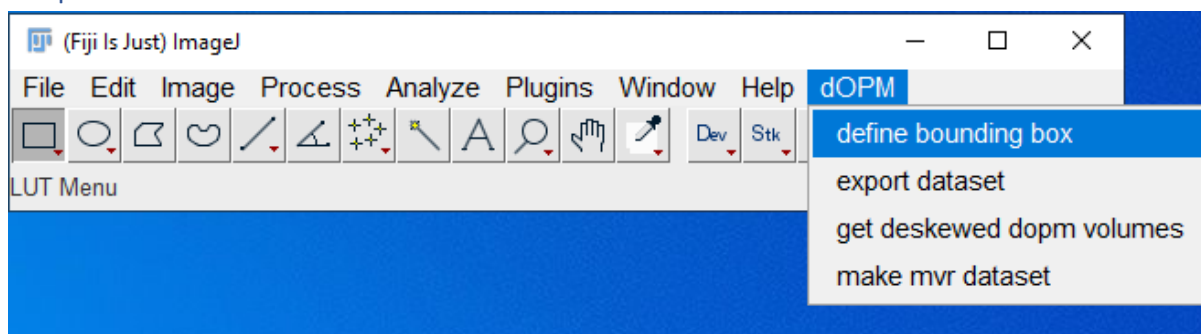
- Click on the dataset path and load once GUI dialog appears below appears



- Worth remembering that we copied all the affine transformations applied to the bead dataset including deskewing and bead-based registration and applied it to the sample data. If the alignment is not as expected, then it might indicate that the sample data and bead data were not recorded under identical acquisition settings and optical alignment.
- The screen shot below shows the ImageJ MVR plugin application for viewing the data. Explore this tool and the online help to understand it better.

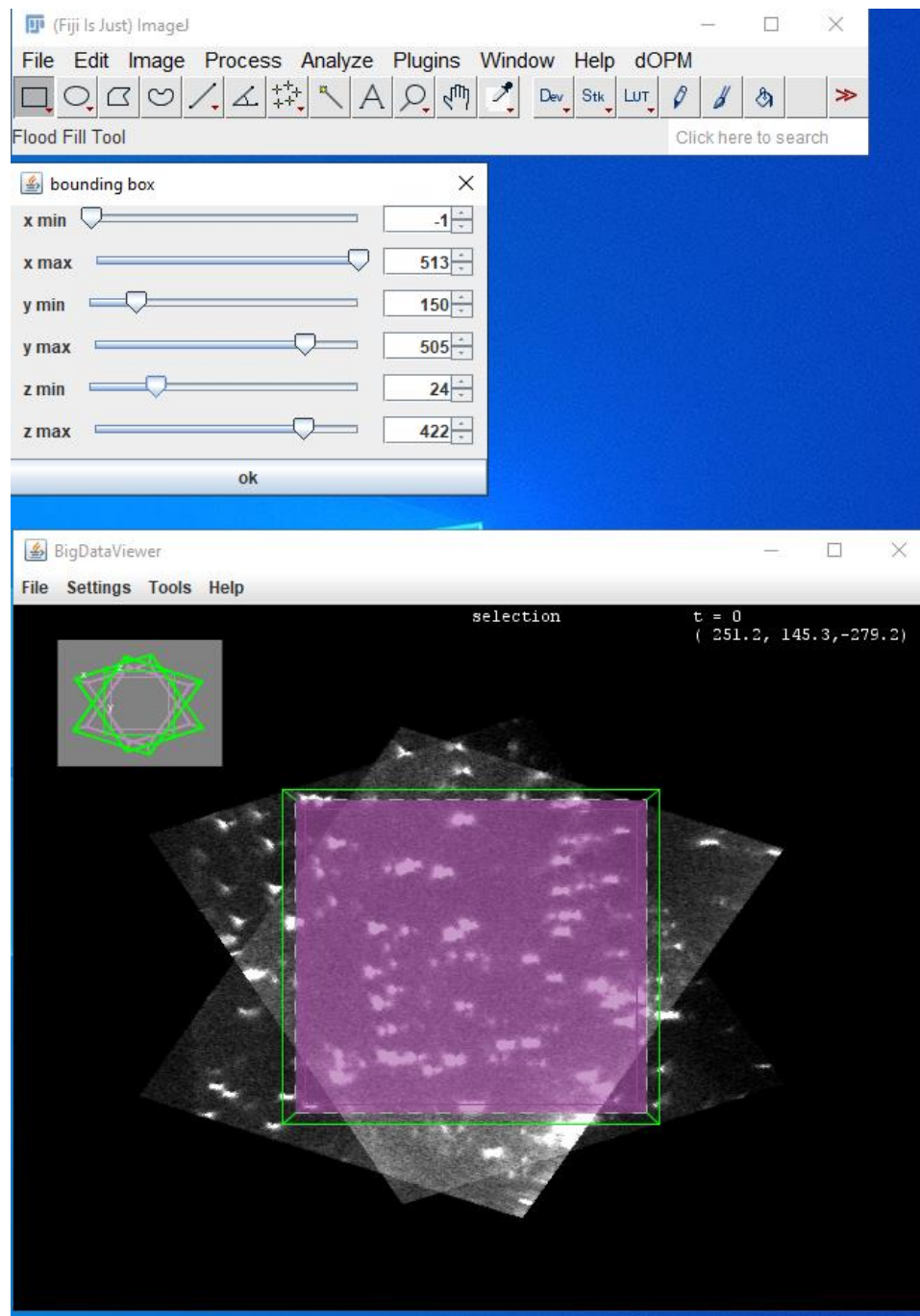


Step 9 - Define a bounding box to crop out a sub-volume, optional step



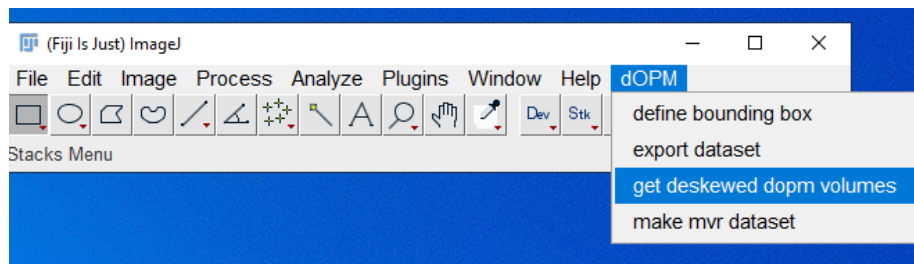
- Click on the 'define bounding box' option

- This step uses the MVR bead dataset that includes the 'hdf5' file for fast viewing with the 'Bigdataviewer' tool while defining a bounding box using a GUI with sliders for changing the bounding box dimensions and visualising the extent of the bounding box for cropping as a green wire frame. The purple shows the perspective of the plane being intersected from the current rendering of the 3D orientation
- The screen shot below shows the ImageJ MVR plugin application for viewing the data while changing the bounding box for cropping. Explore this tool and the online help to understand it better.

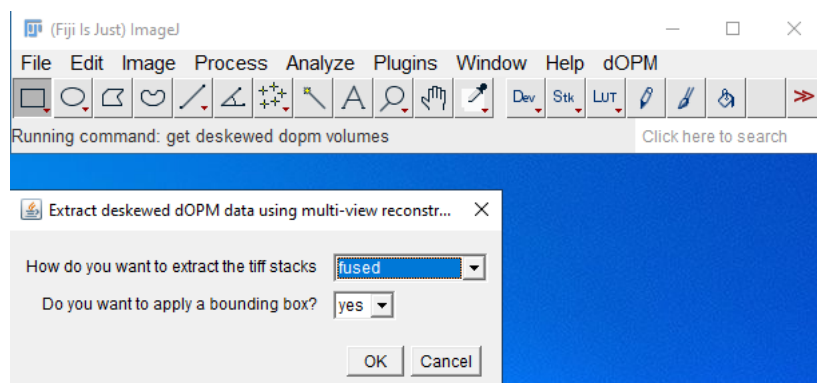


Step 10 - Extract processed single view or fused datasets for generic downstream processing of 3D datasets

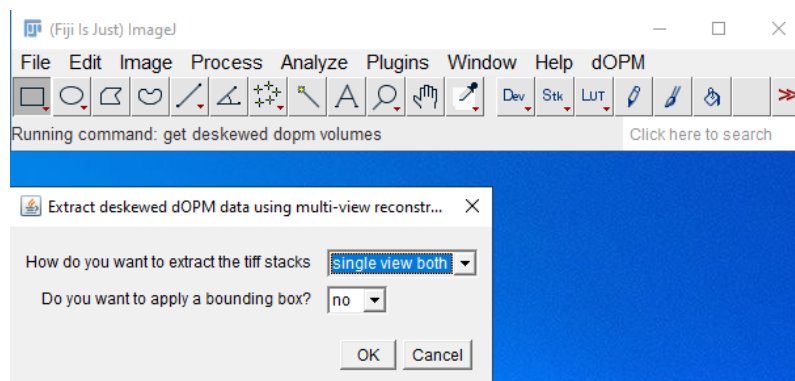
- Click on the 'get deskewed dopm volumes' option



- The 'how do you want to extract the tiff stacks' question gives you the choice of exporting single view or fused data that has had all of the deskewing and other affine transformation steps applied as covered in Steps 1-4
- Below is the fused option selected

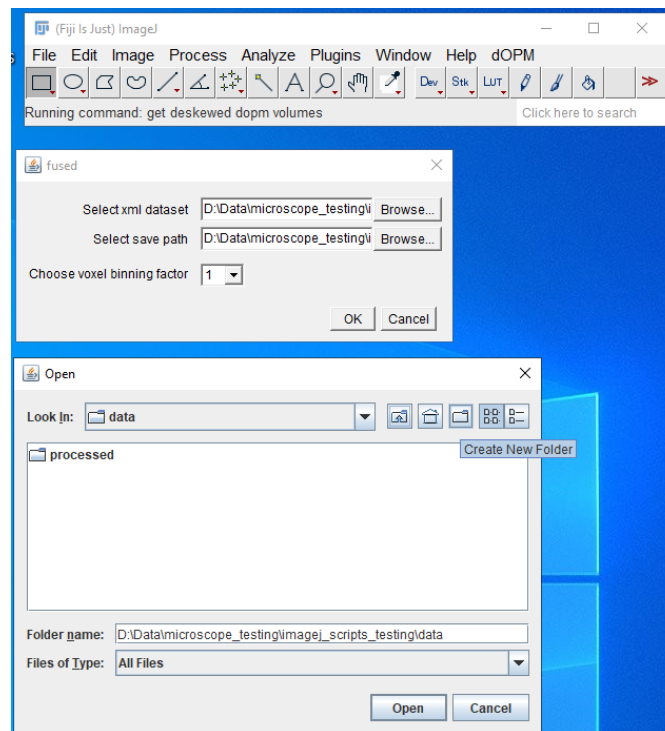


- Below is the single view option selected, you will see another prompt if chosen asking to specify which single view – dOPM view 1 or 2

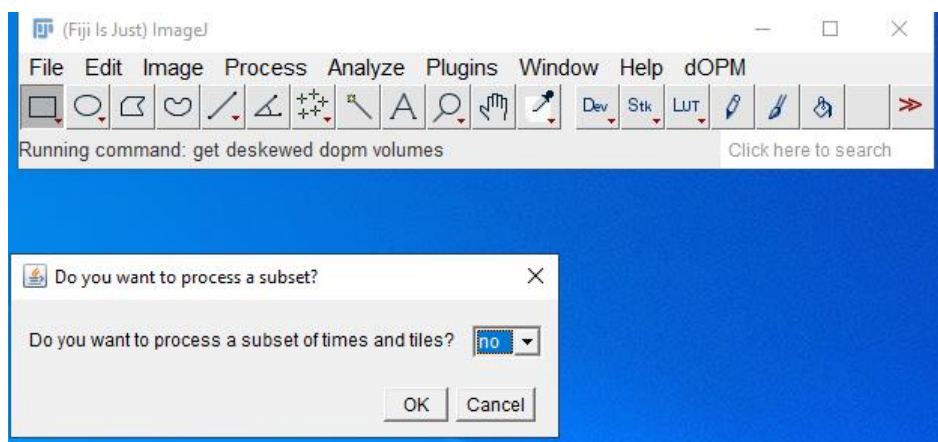


- If you have defined a bounding box as outlined in Step 9 above, you can apply it in this exporting step to reduce the data to just the volume cropped by this box – click 'yes' or 'no' in the 'do you want to apply a bounding box option'. This will be ignored if a bounding box was not defined prior to Step 10.

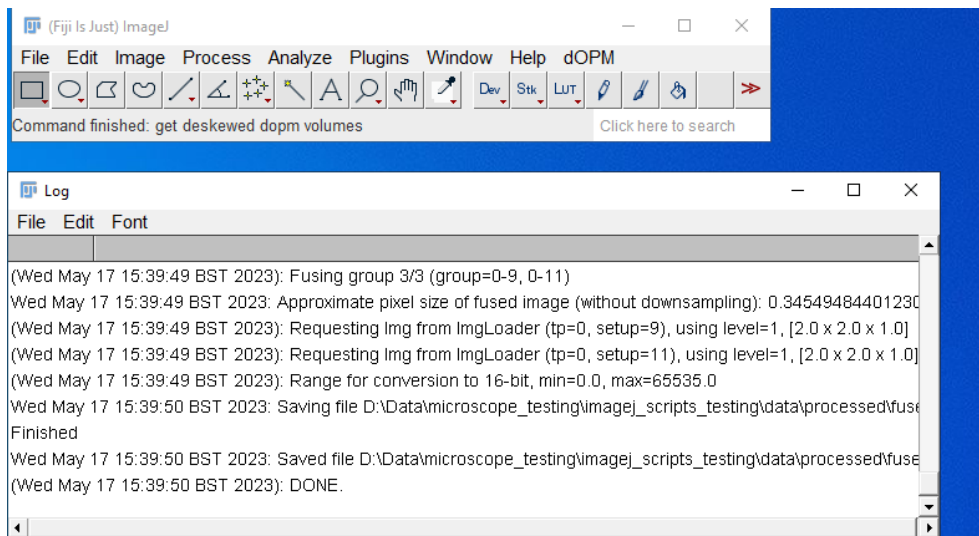
- Regardless of single or fused volumes you will see a prompt requiring a path to the dataset xml and a save path
- Note you must create the save path manually before or during this step for this to work – it will not automatically create a new folder with the chosen name
- Note the binning factor refers to the size of the exported tiff stack voxels and the data is isotropic in sampling in x,y,z dimensions of the produced z-stack and is a function of the raw image datas pixel size and z-plane spacing. For example, a binning factor of 1 would generate a tiff stack with the highest spatial resolution and a factor of 2 would be half this resolution. Explore the MVR application and read online wiki for further understanding of exporting MVR data in general.



- Finally, the option to process a subset of tiles is given to allow the user to only process a fraction of the dataset which could be handy if the dataset is large and only a subset is needed saving compute time and memory.



- Once the exporting begins the log box provides progress and is complete with a 'DONE' statement



- Then you can browse the resulting tiff stacks in the chosen save folder and load them in ImageJ or similar.
- At this point you should have a general 3D image analysis problem such as segmentation

