

Workflow summary

The main goal of this tutorial is to illustrate the combination of different tomography software packages in the same Scipion project. Therefore it does not pretend to achieve high quality reconstructions or analyze the biology of the sample, it only attempts to illustrate some possibilities that ScipionTomo offers in image processing.

Getting Started.

About Scipion

Scipion is an integrative framework for image processing of cryoEM data, in particular: Single Particle analysis, modeling/modelling and electron tomography data. It incorporates as plugins the most common software packages as they are: RELION, cryoSPARC, IMOD, XMIPP, EMAN, DYNAMO among many others. Scipion provides a standard way of communication to allow the interoperability between packages, full tracing of the image processing, and analysis/visualization tools.

Important: Scipion does not modify the original binaries of the integrated software, it just provides a wrapper, where those binaries can be executed in a user-friendly environment.

If you use Scipion, we really appreciate it if you cite it (J.M. de la Rosa-Trevin et. al. 2016) For more information visit our website: <http://scipion.i2pc.es/> Scipion installation

To use this tutorial Scipion must be installed in the machine. The instructions about how to install Scipion can be found in the next link <https://scipion-em.github.io/docs/index.html>

Plugin Requirements

The installation of the Plugins is independent on the Scipion, and they can be installed with the Plugin Manager, or alternatively by command line.

This tutorial will make use of the next ones: Scipion-em-tomo Scipion-em-imod Scipion-em-tomo3d Scipion-em-gctf Scipion-em-cistem Scipion-em-emantomo Scipion-em-novactf Scipion-em-xmipp Scipion-em-xmipptomo

Data Download

The used data for this tutorial can be found in this link, which contains two small subsets of the EMPIAR entries EMPIAR-10453 (SarsCov2 - Turonova et. al. 2020) and EMPIAR-10364 (E. coli minicells - Burt et.al. 2020)

Download the data tutorial here, in the folder day 1

<http://scipion.cnb.csic.es/downloads/scipion/data/tutorials/tomography/>

Contact us

We want to hear from you! Any comment, question, or complaints regarding this tutorial, the use of Scipion or xmipp can be sent to these emails: scipion@cnb.csic.es, xmipp@cnb.csic.es.

Also you can follow us on our social media

Twitter: <https://twitter.com/instructi2pc>

Tutorials about Scipion use, and cryoEM seminars can be found on your YouTube channel

Youtube: <https://www.youtube.com/user/BiocompWebs>

We also have a slack channel where our most active members keep in touch daily. You can request access on scipion@cnb.csic.es

Importing Tilt series

X-ray eraser

Dose filter

Tilt series alignment

IMOD alignment

IMOD - coarse alignment

IMOD - generate fiducial model

IMOD - fiducial alignment

AreTomo alignment

Tomogram reconstruction

Subtomogram picking

Initial model

Subtomogram refinement

Quality analysis

Note: If you are new in Scipion, check our documentation about how to create a new project (<https://scipion-em.github.io/docs/docs/user/scipion-gui.html#scipion-gui>)

First step is to have the raw images in the Scipion project. Scipion provides many different kinds of imports: SetOfTomograms, SetOfCTF3D, SetOfCoordinates, or SetOfSubTomograms, among others. In this tutorial we will show how to import a SetOfTiltSeries.

Note: The import is the unique action (leaving out some exceptions) where a user has to interact with files. Once the data is imported into the project, Scipion carries out the files management during the image processing, enhancing the usability.

Import of the Tilt series

The import of tilt series is carried out by means of the Scipion protocol called Import tilt-series, located in the left-side panel in the Tomography tab. Alternatively, it can be found by searching in the search box of Scipion (Ctrl+F), and typing key words like: import. The Scipion search box also allows searching by keywords like import, or tilt, or tilt series... and therefore identifying the protocol in the deployed list of protocols. The form of the import looks like Figure 2.1.

Figure 2.1. Import tilt series form The import tilt series has different possibilities to load the data into the Scipion project. The different options are the ones provided in the Tilt info box of the form, see Figure 2.1. However, Scipion also offers a more convenient and flexible import, the import from mdoc files. These files are generated during the acquisition session by SerialEM (and more recently, Tomo5), and contain all microscope parameters, the most important and relevant to the image processing: pixel size, tilt axis orientation, defocus or dose among others. It also contains many other parameters like the voltage of the electron gun, magnification, exposure time or spot size of the beam...that in general are not relevant for the image processing.

Note: Scipion team highly recommends importing with mdoc files.

In this tutorial, we will use the import tilt series using the mdoc files. To do that we select in the files directory box of the form the folder where the mdoc files are (see Figure 2.2). The option of importing with mdoc files is carried out by typing *.mdoc in the pattern box. In the moment when the user type *.mdoc the options of Tilt info box should disappear (see figure with the form filled in). In the help button of the pattern box, tips about how to introduce the pattern are explained. In this tutorial the user will type *.mdoc, meaning that all files with extension mdoc of the Files directory folder will be imported. Finally, just click on execute and Scipion will import the tilt series.

Note: Importing with mdoc the parameters of the Acquisition info box are not required. However, they are enabled to be fulfilled just in case the information of the mdoc file could be not fully correct, or even if the user wants to refine some acquisition parameters.

Figure 2.2. Tilt series import from .mdoc file. Only the Files directory and the pattern are mandatory, the rest of boxes can be empty.

The result of the import will be a SetOfTiltSeries that can be observed as output in the lower part of the Scipion main window (see Figure 2.3). In this case, 3 tilt series composed of 41 images with dimensions 5760x4092 images have been imported with pixel size 1.33A. Moreover, Scipion points out that they were imported from mdoc files as well as the path of the mdoc. By right-clicking on the output (in

the summary - lower part of the main Scipion window) we can analyze the output with a different viewer. The Analyze result button (in red) will open the output with the default viewer Scipion GUI

Figure 2.3: Result of the import of tilt series

Preprocessing Tools In this section, some preprocessing tools previous to the tilt series alignment will be shown. These tools are addressed to remove undesired effects (X-ray peaks), consider the radiation damage (dose weighting) and speed up the computational time (binning). In all cases the input is a `SetOfTiltSeries` as well as the output. X-ray eraser Imod References: Kremer 1996, Mastronarde 2017

The interaction of electrons with the sample can result in an X-ray generation that can be detected by the camera. It results in very bright pixels in the images, therefore x-ray peaks are an undesired effect to be corrected. Such a task can be carried out with the protocol imod-Xray eraser. In Figure 3.1 the form of the protocol imod- Xrays eraser shown. The input will be the imported `setOfTiltSeries`. The protocol also provides advanced options (in grey) to modify the criterion of detection, in our case default values are enough.

Note: The output of each step can be checked in the lower panel of the main Scipion window. The results can be visualized by right-clicking on the output object. A menu appears and a viewer can be selected. See: <https://scipion-em.github.io/docs/docs/user/scipion-gui.html#analyzing-results>

Figure 3.1: imod-XRay eraser form, with the execution values (default values in this case)

The current state of the workflow can be observed in Figure 3.2.

Figure 3.2: State of the workflow. Result in the Scipion tree of the XRay eraser.

Dose Weighting Imod References: Kremer 1996, Mastronarde 2017

Once the `SetOfTiltSeries` are free of X-rays peaks, the effect of the radiation damage is taken into account by applying a dose correction. During the image acquisition the sample is exposed to the radiation, and some images have received more dose than others. The dose weighting attempts to compensate for this over exposure of some images by considering the accumulated dose. The result will be another `SetOfTiltSeries`, but in this case with a dose correction that makes the images look like a low pass filtered version of the original ones (with a soft graining).

To carry out the dose whitening in Scipion the imod - dose filter protocol can be opened in the left side panel (or Ctrl+F and searching dose filter in the search box). The form is shown in Figure 3.3 in the advanced mode.

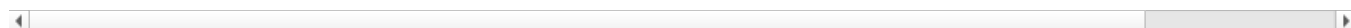
Note: The dose is managed in two different ways. 1) if the dose was introduced in the import or taken from the imported mdoc files, then Scipion will take it automatically from the database. This is the most normal scenario. 2) Alternatively, the user can set the dose manually.

Figure 3.3: Imod - dose filter form with the corresponding parameter of this tutorial

The output is a `SetOfTiltSeries` dose filtered and can be found in the Summary of the lower panel in the main Scipion window. Finally, the current state of the workflow is shown in Figure 3.4.

Figure 3.4: Current state of the Workflow after applying the dose filter Binning Imod References: Kremer 1996, Mastronarde 2017 As we can observe in the summary of the previous protocol (imod - dose filter or any other previous protocol), the `SetOfTiltSeries` is composed of 41 images with dimensions of 5760x4092 px. The large size of the tilt series implies a high computational burden, to reduce it the tilt series will be binned. The binning can be found in the left panned tilt-series->preprocess->imod - normalization (or Ctrl+F imod normalization). Alternatively Tilt-series->preprocess->Xmipp- tilt series resize can be used, however in this tutorial imod - normalization will be the chosen one. The form of imod - normalization can be observed in Figure 3.5. A bin 4 parameter will be introduced in the form, which means that the output `SetOfTiltSeries` will reduce their dimensions by a factor 4 presenting a size of 1440x1024px.

Note: In this step we carry out a binning step with the aim of speeding up the computational times of this tutorial for teaching purposes.



The result can be observed in Figure 3.6, showing the current state of the workflow and the summary.

Figure 3.5: Imod - Normalization protocol used to bin the tomograms. The form contains the used parameters of this tutorial.

Figure 3.6: Current state of the workflow after the binning of the tilt series

Tilt series alignment Once the tilt-series are properly imported and preprocessed into Scipion it is time to perform the last step before the reconstruction of the tomogram, the tilt-series alignment. This task is a 3 process step that is composed of a previous prealignment of the tilt-series based on cross-correlation, followed by the construction of the landmark (or fiducial) models, that will be finally used to obtain the properly aligned tilt-series. Tilt-series prealignment Imod References: Kremer 1996, Mastronarde 2017

The first step in the alignment process is the prealignment of the tilt-series. In this step only the translational alignment (shifts) is solved (no angle correction yet). This calculation is performed by cross correlating the successive images from the tilt-series and stretching the images with the larger tilt angle perpendicular to the tilt axis (cosine stretching).

In order to carry out this process inside Scipion environment, the binned SetOfTiltSeries generated in the previous step will input the protocol, imod - Xcorr prealignment, from the imod plugin, that can be found in the left menu under Tilt-series > alignment, or searching with the Ctrl.+F hotkey, along with the following parameters.

Figure 4.1: Imod - Xcorr prealignment form with the corresponding parameters for this tutorial

Note: It is also possible to modify the alignment algorithm using cumulative correlation (another cross-correlation algorithm that follows a different strategy) or modify the filter parameters. Since a cosine stretching is applied to the tilt-series, it is important to properly determine the tilt-axis angle in order to ensure the quality of the obtained results.

Since we selected the option “Generate interpolated tilt-series”, as it can be seen in the following figure, a double output will be generated. First, a non-interpolated tilt-series whose alignment transformation information will be saved but not applied (so, if we visualize this tilt-series we would see no changes respect to the input one), and secondly, an interpolated (and binned) output from which we can judge the qualitygoodness of the prealignment.

This is a common behaviour for all the alignment protocols inside Scipion pursuing the minimizerecreasereducction in the number of interpolations applied to the same tilt series and reducing the disk space used.

Figure 4.2: Current state of the workflow after the imod - Xcor prealignment

Fiducial model generation Imod References: Kremer 1996, Mastronarde 2017

Once the prealignment process is finished, it is possible to generate the landmark (fiducial) models associated with each tilt series. Before diving into the theory behind these models it is important to be aware that they are designed to work with tilt-series presenting gold beads as fiducial markers, in order to perform the posterior alignment. These protocols will underperform (or simply do not work) with fiducial-less data, so this approach is not recommended if this is the case. Anyway, it is possible to perform this kind of alignment using other tools, as Etomo, presented in the following section.

The landmark (fiducial) model generation consists of a model that provides information of the position of each gold bead in the image. Specifically, it defines the 2D coordinates of one fiducial along all the images in which it has been detected (not necessarily the whole tilt-series). Thus, it is possible to track the position of several landmarks along the whole tilt series, making it possible to posteriorly align them, correcting its tilt axis position.

We will use as input of thisinput to this algorithm with the previous non-interpolated SetOfTiltSeries. The reason to do so, as it has been said previously, is to avoid excessive interpolation of our data. The protocol will apply the alignment matrices to input data and calculate the landmark (fiducial) models to posteriorly offer this information.

In order to carry out this process inside Scipion environment, the prealigned SetOfTiltSeries generated in the previous step will input the protocol, imod - generate fiducial models, from the imod plugin, that can be found in the left menu under Tilt-series > alignment, or searching with the Ctrl.+F hotkey, along with the following parameters.

Figure 4.3: Imod - generate fiducial models form with the corresponding parameters for this tutorial

Note: In this protocol, it is important to properly set the fiducial radius (in nanometers) since, if the indicated size is significantly different from the real one, the algorithm will fail in the fiducial location and posterior tracking. Also, it is possible to set the algorithm to differentiate between those gold beads that are in front (or over) the sample and the ones that are in the rear part (or under it), using the “Find on two surfaces option”.

The output of this protocol is a new Scipion abstraction that has not appeared before, the SetOfLandmarkModels. This object is able to store the position information of each gold bead through the tilt-series for every tilt series belonging to the set.

Figure 4.4: Imod - generate fiducial models form with the corresponding parameters for this tutorial

Fiducial alignment Imod References: Kremer 1996, Mastronarde 2017

Once the landmark (fiducial) models are generated it is possible to calculate the final alignment of the tilt-series. In this final alignment not only the translational movements (shifts) are corrected, basically inherited from the prealignment correction, but also the rotation of the images (angle), aligning the tilt axis with the vertical (Y) axis of the image.

Note: Scipion uses the imod convention putting the tilt axis in the vertical (Y-axis)

This realignment is preformeddone becausesince thisit is the expected disposition of the input data for reconstruction algorithms. The reason for this is a more efficient reconstruction algorithm, since the algorithm will need less resources by reconstructing the tomogram slice by slice in XZ planes (perpendicular to the aligned tilt axis in the Y axis).

The input for this algorithm is just the SetOfLandmarkModels object, obtained in the previous step. This abstraction holds all the information needed, both the Landmark chains locations and the tilt-series associated with them.

In order to carry out this process inside Scipion environment, the binned SetOfTiltSeries generated in the previous step will input the protocol, imod - Fiducial alignment, from the imod plugin, that can be found in the left menu under Tilt-series > alignment, or searching with the Ctrl+F hotkey, along with the following parameters.

Figure 4.5: Imod - fiducial alignment form with the corresponding parameters for this tutorial

Note: This is a very CPUoption intensive protocol. In the first case, and as in the prealignment we will generate the interpolated tilt-series, at a higher binning to save some space in disk. According to the alignment options, it will be solved only for one rotation and with a fixed magnification 1 (no magnification of the images calculated). Also, the tilt angles will be fixed and no distortion of the images will be calculated.

In the first place, this protocol will generate the transformation matrices according to the landmark models to posteriorly generate a finally aligned tilt series. As it has been said in the previous protocol the input tilt series will have it prealignment matrix associated to the (a non-interpolated said) and in the output the protocol will combine this information. And, in second and since we have selected again the "Generate interpolated tilt-series", this information will be applied in order to generate the final interpolated tilt-series.

Also, this protocol generates a refined SetOfLandmarkModels with no gaps, meaning that the position of the landmark lost for some images in the previous steps are now interpolated from the transformation matrices calculated. And finally, it also generates a setOfCoordinates3D. These are the coordinates of the fiducials (gold beads) in the third-dimensional space, being possible to calculate their positions because we already know the final alignment.

Figure 4.6: Imod - fiducial alignment form with the corresponding parameters for this tutorial

Tomogram Reconstruction (imod and tomo3d)

Imod References: Kremer 1996, Mastronarde 2017 tomo3d References: JI Agulleiro 2011, JI Agulleiro 2015

The last step of this workflow is the reconstruction of the tomogram. Scipion provides several methods to undertake the reconstruction task as they are: imod-reconstruction, tomo3d-reconstruction, aretomo-reconstruction, or nova-ctfreconstruction. The protocols of imod reconstruction and tomo3d - reconstruction can be found in the left side panel Tomogram->Reconstruction-> imod-reconstruction and Tomogram-> Reconstruction-> tomo3d-tomo3d respectively (alternatively searching with Ctrl+F). In Figure 5.1 the forms of both protocols fulfilled with the used parameter to reconstruct the corresponding tomograms are shown. The difference between both protocols is the reconstruction algorithm, meanwhile imod uses Weighted Back Projection (WBP), tomo3d makes use of a SIRT method.

Note: WBP is faster than the SIRT method, but SIRT provides higher contrast. The use of SIRT filter attempts to enhance the contrast of the WBP methods.

The input of the reconstruction will be the aligned SetOfTiltSeries, We recommend using the non-interpolated set, because it is always unbinned, in fact in this tutorial non-interpolated is the chosen ones as input. However, the interpolated ones can also be used. Regarding the parameters of the protocols, the most important is the Tomogram Thickness that was set to 300 voxels. In Figure 5.1 the form of both protocols is shown in the used parameters

Note: Despite when non-interpolated SetOfTiltSeries is visualized seems to be unaligned. The SetOfTiltSeries has associated the transformation matrix with the alignment parameters. The interpolated-SetOfTiltSeries has the transformation matrix applied but also can present a binning.

Figure 5.1: imod - reconstruction and tomo3d - reconstruction forms with the used parameters of this tutorial

Figure 5.2: State of the workflow after carrying out the two reconstructions with imod and tomo3d.

The output will be a SetOfTomograms with dimensions 1024x1440x300, see Figure 5.2. They can be visualized by clicking on Analyze results or alternatively by choosing the visualization tool by right-clicking on the output in the Summary box. The most suitable viewer to analyze tomogram is the imod viewer (see Figure 5.3).

Figure 5.3: Results of (left) the tomo3d - reconstruction and (right) imod - tomogram reconstruction for the tilt series TS_320. CTF estimation

The CTF can also be estimated and corrected in Scipion. This is a critical task to achieve high resolution in Subtomogram Averaging, or even to visualize fine details in cellular environments. The CTF correction attempts to compensate for the loss of information that the microscope introduces as a consequence of aberrations and defocus.

To do that Scipion provides a broad variety of CTF estimation protocols as they are: cistem - tilt series CTFFIND4, gctf- tilt series gctf, imod - CTF estimation manual, imod - CTF estimation auto, and emantomo - ctf estimation. In this tutorial, the CTF will be estimated with all of them with the aim of showing the convenience and power of combining methods. All these CTF protocols can be found in the left side panel, tilt series -> CTF, and the input will be the imported SetOfTiltSeries.

Note 1: The goal of this section is to show how estimating the CTF with different protocols the reliability of the results can be enhanced by consensus.

Note 2: The CTF should be estimated with the raw tilt series without any kind of preprocessing. The use of dose filters, or alignment can

change the spectral properties of the tilt series, and throw inaccuracies in the CTF estimation.

The result will be a `SetOfCTFTomoSeries`, that can be visualized by clicking on Analyze results or alternatively by right-clicking on the output, in the Summary panel (low area of the main Scipion window). The viewer will open a window with the list of `SetOfCTFTomoSeries` corresponding to each tilt series. In this window the defocus, astigmatism and resolution associated to each tilt image can be visualized. Also, the window presents a summary plot with the defocus and resolution per tilt image. This plot will be of special interest to validate the estimation of the CTF. Moreover, Scipion adds some internal criteria to advise the users if the estimation is correct or not. `gctf - tilt-series gctf` `gctf` Reference: Zhang 2015

`gCTF` is the CTF estimation method of Single Particle Analysis adapted to deal with tilt series. IT can be found in tilt series -> CTF-> `gctf- tilt-series gctf` (or alternatively Ctrl+F typing `gctf`). The input will be the original `SetOfTiltSeries`.

Note: `gCTF` estimates defocus and astigmatism by default.

In Figure. 6.1, the form with the used parameters can be observed. There are two tabs of the form: Phase Shift and Streaming are not shown in Figure 6.1. However, their default values are correct, in particular Estimate phase shift? No.

Figure 6.1: Form with the used parameters in this tutorial of the protocol `gctf- tilt-series gctf`

The results of `gCTF` can be observed and analyzed in Figure 6.2.

Figure 6.2: Results of the `gCTF` protocol

`cistem - tilt-series ctffind4 cistem` Reference: Rohou 2015

Similarly to `gCTF`, `ctffind4` is the CTF estimation method of Single Particle Analysis adapted to deal with tilt series. It can be found in tilt series -> CTF-> `cistem- tilt-series ctffind4` (or alternatively Ctrl+F typing `ctffind`). The input will be the original `SetOfTiltSeries`.

In Figure. 6.3, the form with the used parameters is shown. There are two tabs of the form: Phase Shift and Streaming are not shown in Figure 6.3. However, they consider default parameters, in particular Estimate phase shift? No.

Figure 6.3: Form with the used parameters in this tutorial of the protocol `cistem- tilt-series ctffind4`

The results of `ctffind4` can be observed and analyzed in Figure 6.4. It can be shown that Scipion suggests (See status column) that the estimation of the CTF for the first two tilt series is not fully correct, this fact is in agreement with the right plot.

Figure 6.4: Results of the `ctffind4` protocol

`imod - CTF estimation auto imod CTF` Reference: Xiong 2009 Imod References: Kremer 1996, Mastronarde 2017

`imod - CTF estimation auto` can be found in tilt series -> CTF-> `imod - automatic ctf estimation (Step 1)` (or alternatively Ctrl+F typing `CTF estimation`). The input will be the original `SetOfTiltSeries`.

Note: `imod` allows to estimate only defocus and if the users wants astigmatism. This method is the same than `imod - CTF estimation manual`, but presetting the estimation parameters avoiding the estimation one by one of the CTF

In Figure. 6.5, the form with the used parameters can be observed. The input of the protocol will be the raw `SetOfTiltSeries`. `imod - CTF estimation auto` protocol presets many parameters to be set. The most important are: The expected defocus that will be set to 3000 nm (this information is known from the acquisition session, and usually contained in the imported `mdoc` files); defocus tolerance set as 200 nm around the expected defocus; Search astigmatism, set as No; Phase Shift as No; Search cut-off frequency as No; The parameters of Angle step and angle range are used to speed up the estimation and to enhance the detection of the Thon rings. Thus, an angle step of 0 means that the CTF will be estimated for each tilt angle of the tilt series. When the Angle step is greater than 0, then, the CTF is estimated in steps of the defined parameter. Intermediate estimations are then interpolated. In contrast, the angle range takes all images in an interval (of the angle range defined) centered at the tilt angle at which the CTF is going to be estimated, and the images are used to increase the signal of the Thon rings. When the CTF estimation presents drawbacks, increasing the angle range is usually a good test.

Note: These parameters work with this tilt series, other tilt series can require different parameters to estimate the CTF.

Figure 6.5: Form with the used parameters in this tutorial of the protocol `imod- automatic CTF estimation`

The results `imod - CTF estimation auto` can be observed and analyzed in Figure 6.6. In this case, we can observe that the `TiltSeries 048` presents a status - Failed, in agreement with `cistem`. However the other estimations are in agreement with `gCTF` and `cistem`.

Figure 6.6: Results of the `imod - CTF estimation auto` protocol

`imod - CTF estimation manual imod CTF` Reference: Xiong 2009 Imod References: Kremer 1996, Mastronarde 2017

This manual estimation will not be carried out in this tutorial, next guidelines are only informative.

`imod - CTF estimation manual` can be found in tilt series -> CTF-> `imod - CTF estimation manual` (or alternatively Ctrl+F typing `CTF estimation`). The input will be the original imported `SetOfTiltSeries`, and the protocol will be fulfilled with exactly the same parameter than

imod - CTF estimation auto. The difference with that protocol is that in this case, when the execute button is clicked, a window with the list of tilt series is shown. Then by selecting and double-clicking on any of them an interactive window of imod will be open.

Figure 6.7: Imod - CTF estimation manual form with the used parameters

emantomo - ctf estimation emantomo Reference: Chen 2019

emantomo - ctf estimation can be found in tilt series -> CTF->emantomo - ctf estimation (or alternatively Ctrl+F typing ctf). The input will be the original SetOfTiltSeries.

Note 1: emantomo - ctf estimation only estimates defocus. Note 2: emantomo - ctf estimation considers um as defocus unit

In Figure. 6.8, the form with the used parameters can be observed. The important ones are the Defocus range, which defines the interval of defocus to be analyzed, with the corresponding defocus steps. In this case the defocus search will be in the interval 2um-7 um in steps of 0.02 um. Similarly occurs with the Phase shift, it will be estimated in the interval 0.0-1.0 in steps of 1.0 degrees. Finally, the Tile size defines the box size to compute the Fourier transform, powers of 2 are recommended, 256 or 512.

Figure 6.8: For with the used parameters in this tutorial of the protocol emantomo - ctf estimation

The results of emantomo can be observed and analyzed in Figure 6.9. In this case all estimated CTF are in the same defocus range (around 25000A), and Scipion suggests that they are well estimated.

Figure 6.9: Results of the emantomo - ctf estimation protocol

Filtering tilt series by CTF estimation

Up to this step the CTF has been estimated with 4 different software packages that make use of different algorithms. Thus, they provide similar, but slightly different estimations. This section will teach how to create a subset. However, such subset will be created based on the most reliable SetOfTiltSeries from the CTF estimations. To do that, we will compare the CTF estimation of all methods. As it can be observed, the estimations of gCTF and emantomo seem to be the most reliable, due to the range of estimated defocus. This fact is supported by the Scipion criterion that sets an ok status to their CTF estimations. Despite, gCTF and emantomo are able to estimate the CTF of all tilt series, the tilt series 048 presents drawbacks for imod and ctffind4. Thus, we will remove this tilt series with a double teaching objective: 1) speed up the reconstruction, 2) show how to create subsets.

The creation of a subset of SetOfCTFTomoSeries can be carried out in the CTF viewer. It only requires marking the unwanted tilt series (see Figure 7.1) and clicking on Generate subsets. Then, in the Summary of the Scipion main windows two output of CTFTomoSeries should appear, the original one, and the created subset.

Figure 7.1: How to create a subset of CTFTomoSeries with the CTF viewer. The results of this figure correspond to gCTF estimation.

Due to the number of CTFTomoSeries of the created subset is lesser than the number of SetOfTiltSeries (3 SetOfTiltSeries and 2 CTFTomoSeries) there will be a mismatch between these two sets. Therefore, it is necessary to create a subset of the SetOfTiltSeries with the corresponding TiltSeries with CTF estimated in the subset of CTFTomoSeries.

The subset of the TiltSeries will be created for the aligned ones (output non-interpolated of imod - fiducial alignment). To do that, we open the output (non-interpolated) by right-clicking (in the summary box) with DataViewer, see Figure 7.2. This should open a window (see Figure) with a table containing all information related to each tilt series. To create the subset, the tilt series 048 will be unmarked, then just click on the red button +TiltSeries. The result will be a new building block in the ScipionThree that contains only two tilt series the TS_293 and TS_320.

Figure 7.2: Data viewer with the NON-interpolated tiltseries. The _tsId column can be in a different order in the table of the user.

CTF correction Once the subset with the good tilt series has been created, then a CTF correction can be carried out.

Note 1: Scipion has a standard CTF model, when the CTF is estimated with any CTF method, the output is stored in the Scipion standard. Thus, to correct the CTF Scipion converts the standard to the corresponding package, in this case imod - CTF correction. This avoids the mismatching between units and parameters of different software packages.

Note 2: When an image processing with SubTomogram Averaging (STA) is performed, the CTF correction is needed to later correct the local CTF of each subtomogram, but the CTF correction of the Tilt series images (this section, imod - CTF correction) is not so common with some exception like small proteins. In contrast, dealing with cellular environments (no STA), the CTF should be always corrected.

Some of the CTF protocols do estimate the astigmatism, like gctf, or ctffind4, others do not like emantomo, and in the case of imod, the user can choose if the astigmatism is calculated or not.

Note: If the user wants to correct the astigmatism, it must be rotated to adapt the astigmatism to the imod model.

Rotate astigmatism Scipion has its own CTF standard model to allow the conversion between protocols, some methods like imod - CTF correction or nova - novaCTF reconstruction expect a given orientation of the astigmatism. Thus, it is necessary for these protocols an intermediate step to rotate the astigmatism and adapt the output of the CTF to these protocols. The protocol xmiptomo - astigmatism

rotation undertakes exactly that task. It can be found in the left side panel Tilt Series->CTF-> xmiptomo - astigmatism rotation, (or with the hotkeys Ctrl+F searching by astigmatism). The form of the protocol can be found in Figure X, and its input will be the SetOfCTFTomoSeries of gCTF (because we decided to use the gCTF estimation). The input and the output of xmiptomo - astigmatism rotation will both be SetOfCTFTomoSeries, and therefore the output can also be opened with the CTF viewer by right-clicking on the output, or just clicking on Analyze Results button.

Figure 8.1: Form of xmiptomo - astigmatism rotation with the used parameters

imod - CTF correction imod CTF Reference: Xiong 2009 Imod References: Kremer 1996, Mastronarde 2017

Once the astigmatism has been rotated, the CTF correction can be performed. To do that imod - CTF correction protocol will be used. It can be found in the left-side panel, Tilt-Series->CTF->imod - CTF correction, or alternatively (Ctrl+F searching CTF correction). The input of this protocol is a SetOfCTFTomoSeries (the output of any CTF estimation protocol, like gctf, ctffind4, imod, or emantomo) and the SetOfTiltSeries to be corrected. In this tutorial the input will be the two created subsets: the one with the good set of tilt series and the other with the good set of CTF. Thus, imod - CTF correction form with the used parameters is shown in Figure 8.2.

Figure 8.2: imod - CTF correction form with the used parameters

Tomogram Reconstruction with CTF correction Tomogram reconstruction using CTF corrected Tilt series Imod References: Kremer 1996, Mastronarde 2017

This section will be the same that the one in section 5 - Tomogram Reconstruction, but substituting the input SetOfTiltSeries by the output of imod - CTF correction, it means the SetOfTiltSeries with CTF corrected. As we did in the mentioned section, to reconstruct the tomograms imod - tomogram reconstruction can be used (also tomo3d - reconstruction), it is located in Tomogram->reconstruction->imod - tomogram reconstruction. The used parameters can be found in Figure 9.1, and they are exactly the same than the used in section 5. Tomogram Reconstruction, with the exception of apply a sirt filter in the advanced options. This time, this option has been enabled and set a sirt-filter of 25 iterations to show how the use of this filter significantly increases the contrast of the reconstruction. In Figure 9.1, the form of the imod - tomogram reconstruction with a sirt-filter of 25 iterations is shown. A comparison between this reconstruction and the ones obtained in section 5 . Tomogram Reconstruction with imod - tomogram reconstruction without applying the sirt filter and with the pure SIRT algorithm of tomo3d - reconstruction, can be observed in Figure 9.1.

Figure 9.1: imod - tomogram reconstruction parameter for reconstructing the tomogram with CTF correction using a SIRT like filter

Figure 9.2: imod - tomogram reconstruction form using a sirt filter considering as input the CTF corrected tilt series

Tomogram reconstruction using local CTF novaCTF reference: Turanova 2017

Alternatively to the Tomogram reconstruction using CTF corrected Tilt series of the previous section, it is possible to reconstruct the tomogram considering that each region of the tomogram will present a different defocus, it is applying a local CTF, or a local defocus. The protocol nova - Tomogram CTF defocus carries out exactly this task.

The tomogram reconstruction with the CTF corrected tilt series can be

Figure 9.3: nova - Tomogram CTF defocus

Finally the whole workflow after this tutorial should look like the one in Figure 9.4

Figure 9.4: Final state of the workflow References JM De la Rosa-Trevín, A Quintana, L Del Cano, et al. Scipion: A software framework toward integration, reproducibility and validation in 3D electron microscopy, *Journal of Structural Biology*, 195,1, 93-99 (2016). A. Burt, C.K. Cassidy, P. Ames, P. et al. Complete structure of the chemosensory array core signalling unit in an E. coli minicell strain. *Nat Commun* 11, 743 (2020). B. Turoňová B, M. Sikora, C. Schürmann, et. al. In situ structural analysis of SARS-CoV-2 spike reveals flexibility mediated by three hinges, *Science* 370 203-208 (2020) J.R. Kremer, D.N. Mastronarde, J.R. McIntosh, Computer Visualization of Three-Dimensional Image Data Using IMOD, *Journal of Structural Biology*, 116, 1, 71-76 (1996) D.N. Mastronarde, S.R. Held, Automated tilt series alignment and tomographic reconstruction in IMOD, *Journal of Structural Biology*, 197, 2, 102-113 (2017) JI Agulleiro, JJ Fernandez. Fast tomographic reconstruction on multicore computers. *Bioinformatics* 27:582-583, (2011). JI Agulleiro, JJ Fernandez. Tomo3D 2.0--exploitation of advanced vector extensions (AVX) for 3D reconstruction. *Journal of Structural Biology* 189:147-152, (2015). K. Zhang. Gctf: Real-time CTF determination and correction, *Journal of Structural Biology*, 193, 1, (2016) A. Rohou, N. Grigorieff, CTFFIND4: Fast and accurate defocus estimation from electron micrographs, *Journal of Structural Biology*, 192, 2, (2015) M. Chen, J.M. Bell, X. Shi, X. et al. A complete data processing workflow for cryo-ET and subtomogram averaging. *Nat Methods* 16, 1161-1168 (2019) Q. Xiong, M.K. Morpew, C.L. Schwartz, CTF Determination and Correction for Low Dose Tomographic Tilt Series, *Journal of Structural Biology*, 168(3) 378-387 (2009). B. Turoňová, F.K.M. Schur, W. Wan, and J.A.G. Briggs, Efficient 3D-CTF correction for cryo-electron tomography using NovaCTF improves subtomogram averaging resolution to 3.4 Å, *Journal of Structural Biology*, 199, 3, 187-195, 2017