

# 2DX Graphical User Interface

# Beginners' User Guide

---

*for 2DX version 4.x*

Nikhil Biyani, Marcel Arheit, Sebastian Scherer, Cristina Paulino, Ricardo Righetto and  
Henning Stahlberg

E-CINA



**2DX**

2D Electron Crystallography  
Image Processing Suite

## Table of contents

<a href="#">2DX Graphical User Interface</a>	0
<a href="#">Beginners' User Guide</a>	
<a href="#">1. Introduction</a>	3
<a href="#">1.1. Background</a>	3
<a href="#">1.2. Setting up new project</a>	3
<a href="#">1.2.1. Launching 2DX GUI</a>	3
<a href="#">1.2.2. Add new project</a>	4
<a href="#">1.2.3. Project Structure</a>	4
<a href="#">1.3. Overview of 2DX GUI</a>	5
<a href="#">1.3.1. Project Library</a>	5
<a href="#">1.3.2. Edit Images</a>	5
<a href="#">1.3.3. Merge Tool</a>	6
<a href="#">1.3.4. Project Tools</a>	6
<a href="#">1.3.5. Preferences</a>	6
<a href="#">1.4. External Software &amp; Preferences</a>	6
<a href="#">1.4.1. Python</a>	6
<a href="#">1.4.2. EMAN2</a>	6
<a href="#">1.4.3. CCP4</a>	6
<a href="#">1.4.4. Standard Applications</a>	7
<a href="#">1.4.5. Optional Software</a>	7
<a href="#">1.5. Importing Images and Movies</a>	7
<a href="#">1.5.1. Preparing for the import</a>	7
<a href="#">1.5.2. Importing using Project Tools</a>	8
<a href="#">2. Image Processing of 2D Crystal Images</a>	10
<a href="#">2.1. Theoretical Background</a>	10
<a href="#">2.2. Getting started</a>	11

<a href="#">2.3. Automated Processing</a>	11
<a href="#">2.4. Manual Processing</a>	12
<a href="#">2.4.1. Determine defocus</a>	12
<a href="#">2.4.2. Determine lattice</a>	13
<a href="#">2.4.3. Determine tilt geometry</a>	14
<a href="#">2.4.4. Unbend</a>	16
<a href="#">2.4.5. Movie mode unbending</a>	18
<a href="#">2.4.6. Crystal masking</a>	20
<a href="#">2.4.7. CTF Correction</a>	23
<a href="#">2.4.8. Space Group Determination</a>	24
<a href="#">2.4.9. Map Generation</a>	25
<a href="#">2.5. Setting config as Project Default</a>	25
<a href="#">2.6. Parallel Image Processing</a>	26
<a href="#">3. 2D Merging</a>	27
<a href="#">3.1. 2D Merging in 2DX</a>	28
<a href="#">4. 3D Merging</a>	32
<a href="#">4.1. 3D Merging in 2DX</a>	33
<a href="#">5. Refining Merged Map</a>	36
<a href="#">6. Porting Project to FREALIGN</a>	37

# 1. Introduction

## 1.1. Background

2DX [Gipson et al. 2006, Gipson et al. 2007] is an initiative that aims at facilitating computer image processing in electron crystallography. Electron crystallography is our method of choice for the determination of the high-resolution structure of membrane proteins in the membrane embedded state. This is done by recording cryo-electron microscopy images of two-dimensional membrane crystals of the membrane proteins, and reconstructing the 3D structure of the proteins by computer image processing of a larger number of images.

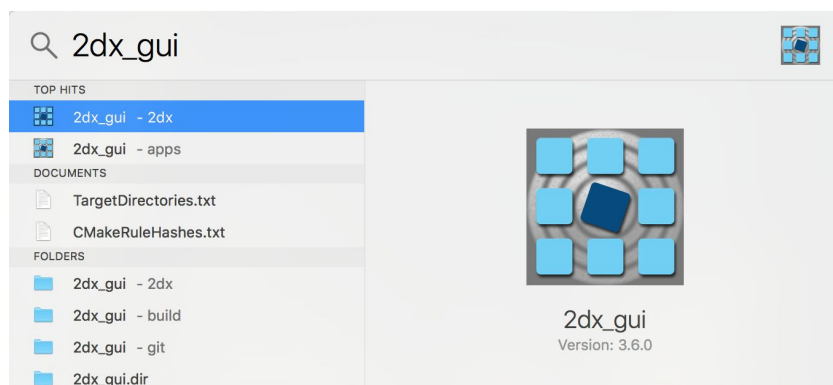
2dx utilizes the excellent MRC software that was developed and is maintained at the MRC by R. Henderson and others . We have a slightly adapted version of the MRC software code in order to interface it with the 2dx system, and we also have added programs for automation and other functions.

2DX Software Suite comprises of many softwares including a Graphical User Interface (GUI). The GUI can be launched using the prebuilt app on macOS named *2dx\_gui*. *2dx\_gui* is loaded with most of the functionalities needed from processing individual micrographs to merging the images to get a three dimensional (3D) map.

## 1.2. Setting up new project

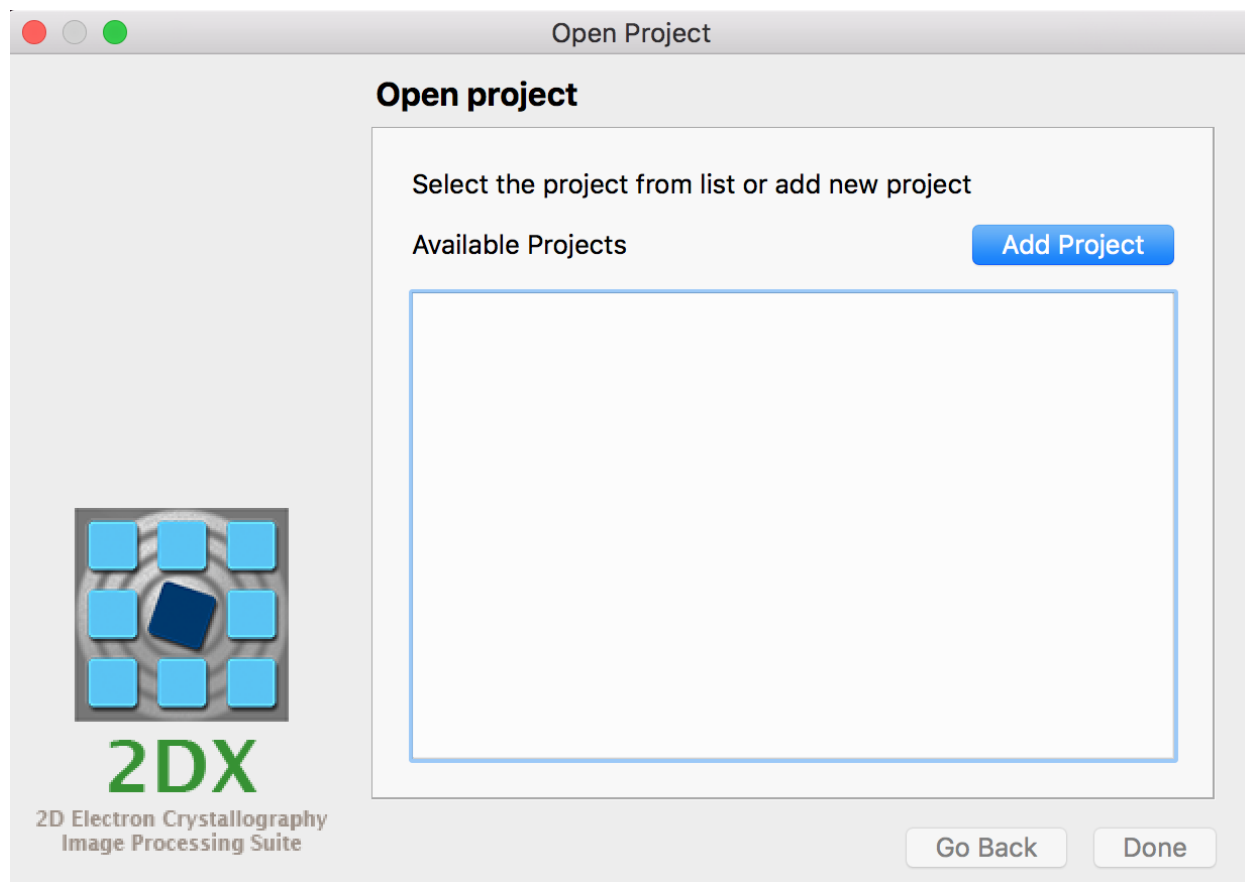
### 1.2.1. Launching 2DX GUI

Once installed, 2DX can be opened from the spotlight. Search *2dx\_gui* in the spotlight and press Enter to launch 2DX GUI.



### 1.2.2. Add new project

Launching 2DX GUI opens a dialog which can be used to manage 2DX projects. One can add a project using the “Add Project” button. A 2DX project should always be created in a separate folder. As you don't have a project right now, you should create a new folder



called "Protein\_A" under "Desktop". Select the newly generated folder and click on "Choose". You would now be asked to give a name to the project. We will call our project “Protein A”. 2DX now will look for configuration data in the selected folder. If the required data is not found the application will ask you whether a new configuration file should be created in this location. Once confirmed 2dx\_gui main window will be launched.

### 1.2.3. Project Structure

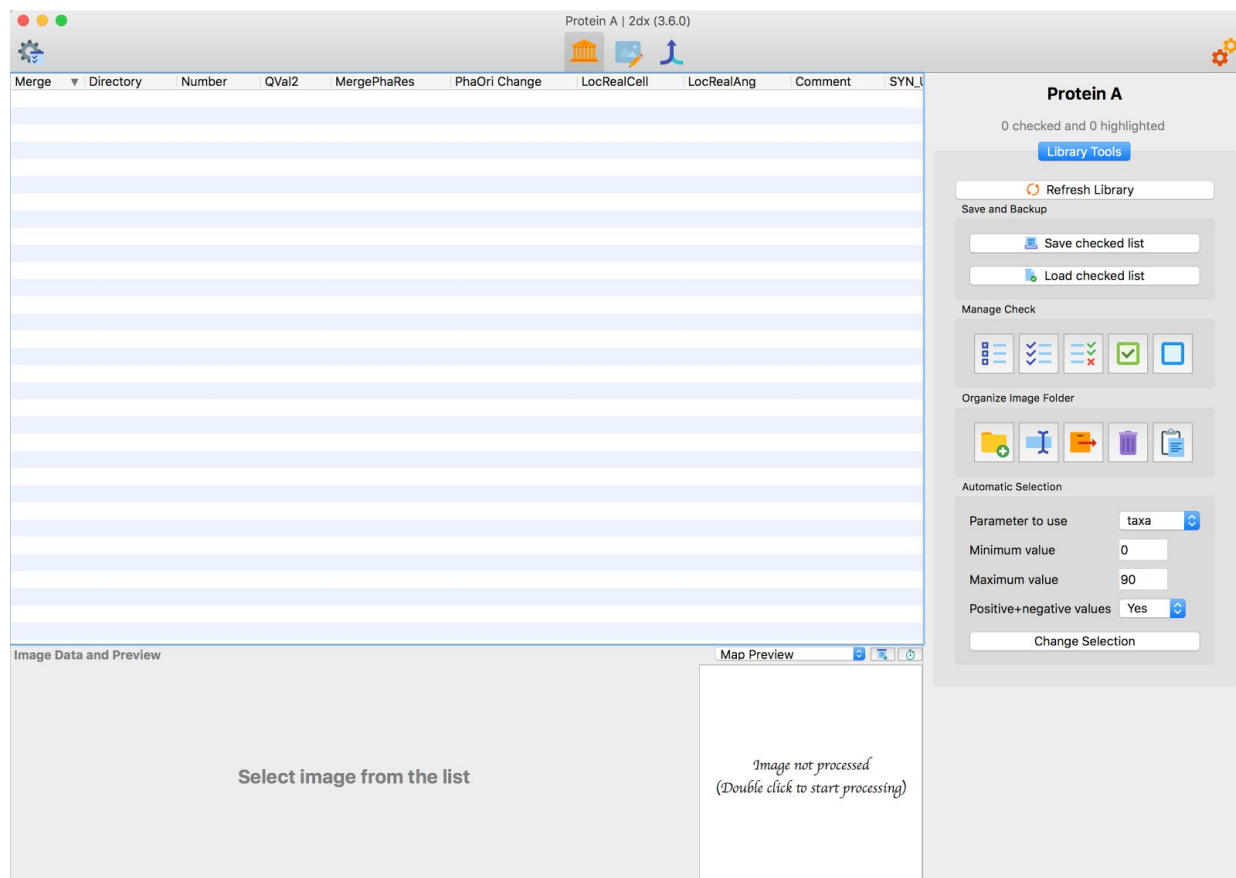
A 2dx-project directory contains the following folder and files:

- **2dx\_master.cfg** is the top-level configuration file which serves as default configuration files for all calculations performed on the entire project. Note that you should not edit these files manually as they are usually set through the graphical user interface.
- **merge-folder** contains all the files generated while merge multiple images by means of 2dx\_merge.

- **Folders per image groups:** 2dx generates a folder for each tilt angle. Each image and all the corresponding files generated during the single image processing are stored in an individual folder in the corresponding tilt angle folder.

### 1.3. Overview of 2DX GUI

2DX GUI has five major navigation panels, all shown on the top bar. These are as follows:



#### 1.3.1. Project Library

Project Library consists of all the micrographs that were imported in the project. The micrographs are grouped by their name into “Image Folders”. You can select individual micrographs by clicking the checkbox in front of the image name. You can also select an entire folder by selecting the corresponding top-level directory. There is a menu on the left of the panel which can be used to perform a handful of operations on the project library.



#### 1.3.2. Edit Images

All the images/micrographs that were opened for editing can be accessed from this panel. Opened images are organized in different tabs like in a browser. All the images that were left open when the application was quit, will automatically be opened on next startup.



### 1.3.3. Merge Tool

Merge Tool is the panel which one would use when all the images are processed and merging is to be started. This panel can be useful for doing 2D Merging and 3D Merging and also features functionalities to import the 2DX project to FREALIGN/RELION.



### 1.3.4. Project Tools

Project Tools consists of scripts which help in organizing project. For example to import images, to process them in parallel or to repair project links one would need to go to the Project Tools panel.



### 1.3.5. Preferences

Preferences panel is used to set paths and applications that would be used during the processing. It is important that after the installation one changes these paths to make sure that everything works smoothly. This panel also features options to change the fonts.



## 1.4. External Software & Preferences

Before you start any processing, it is important to properly install dependent software and set their paths in the preferences panel. Preferences panel consists of two major categories. We here focus on the “Applications and Paths”.

### 1.4.1. Python

Python is used to run many scripts. These scripts need python (2.x) version to run properly. It is to be installed and the path can be set should be set in the preferences. Apart from this if you are running ZORRO, installation of Anaconda python packages is necessary. For this one needs to provide the Anaconda python executable path.

### 1.4.2. EMAN2

EMAN2 is the successor to EMAN1. It is a broadly based greyscale scientific image processing suite with a primary focus on processing data from transmission electron microscopes. The software can be downloaded from <http://blake.bcm.edu/emanwiki/EMAN2>. After the installation set the installation directory path in the Preferences panel of 2DX.

### 1.4.3. CCP4

CCP4 exists to produce and support a world-leading, integrated suite of programs that allows researchers to determine macromolecular structures by X-ray crystallography, and

other biophysical techniques. CCP4 aims to develop and support the development of cutting edge approaches to experimental determination and analysis of protein structure, and integrate these approaches into the suite. CCP4 is needed to manipulate maps in 2DX. Download and install CCP4 from <http://www.ccp4.ac.uk/download/index.php>.

#### 1.4.4. Standard Applications

2DX also uses some standard applications such as Text Editor, Web Browser. Most of these are inbuilt in the OS. One can set the command to these applications in the Preferences.

#### 1.4.5. Optional Software

Softwares like Spider are not essential but recommended as some of the script will run only if spider is installed.

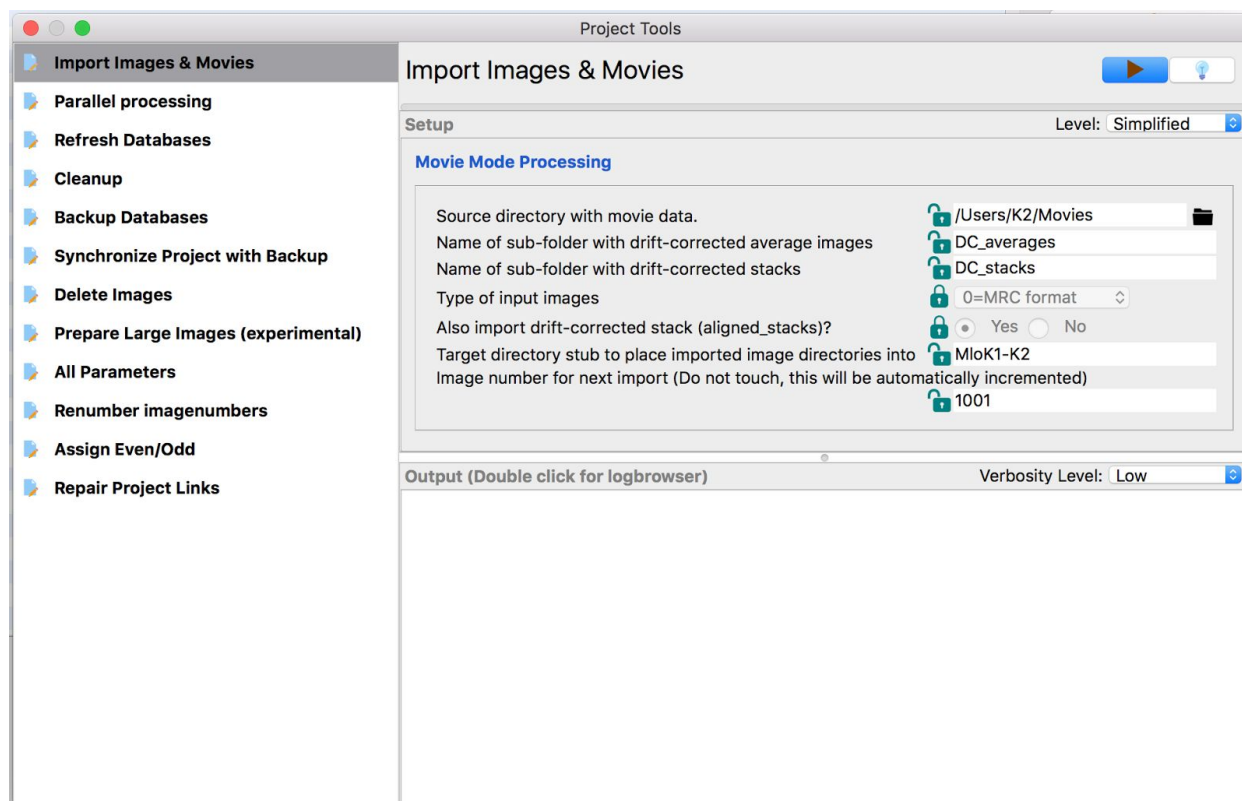
### 1.5. Importing Images and Movies

#### 1.5.1. Preparing for the import

Before you start to import make sure that the movies and averages are stored in a particular way expected by 2DX. Create a main folder containing a group of images. This folder should have two subfolders one containing the movies and the other containing the averages. The correspondence of the average to the movie would be done using same file names. So the corresponding average and movie should have same name. The input images and movies can be in "mrc" or "tiff" format. Such different main folders should be arranged for each group of images. Although grouping can also be done later.



## 1.5.2. Importing using Project Tools



Open the “Project Tools” panel from the top bar places left corner. This opens a dialog containing various scripts and their parameters. Select the script named “Import Images and Movies” from the scripts panel on the left side as shown in the image. Fill in the parameters appropriately using the folders created in the last step.

**Tip (Extra information on parameters):** Hovering on the parameter labels gives more information about each parameter using a tool tip.

**Tip (Help with scripts):** The “bulb” button on the top right near the run button gives more information about the script.

**Tip (Using advanced settings):** In some cases there are some advanced parameters available which can be accessed from the drop down menu just above the parameters panel.

Once all the parameters are set, the import can be started using the play button on the right hand side of the script label. The progress of the import is shown from the progress bar. The output is shown in the bottom container.



**Tip (Changing output verbosity):** The verbosity level of the output can be changed with the drop down menu of the top of the output container. There are four levels starting from Silent to Highest verbosity.

After the images are imported close the “Project Tools” panel. In “Project Library” panel “Refresh Library” to update the project library. You would see that the library now would have images listed.

**Tip (Changing columns in project library):** The default parameter selection can be adapted to your needs by simply right-clicking on the parameter header. You can choose your parameter selection in the appearing selection list.

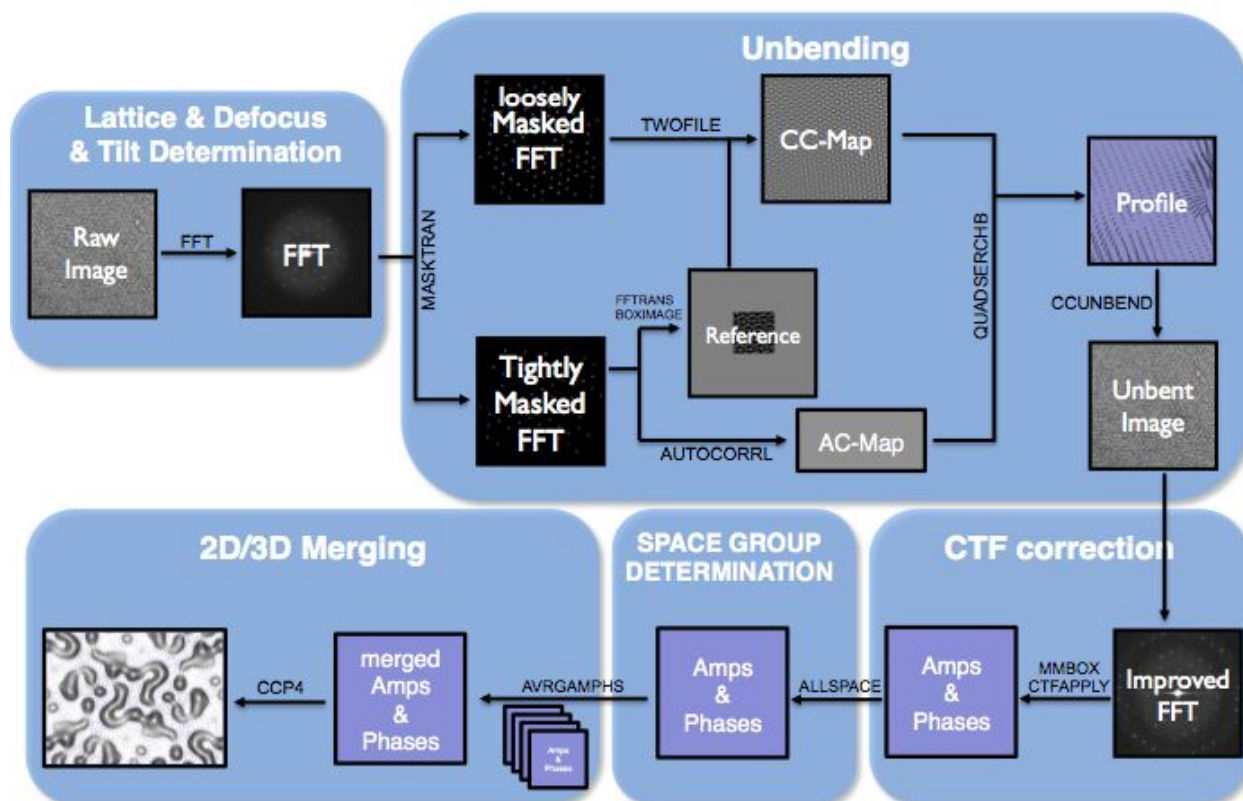


## 2. Image Processing of 2D Crystal Images

Electron crystallography of membrane proteins uses cryo-transmission electron microscopy to image frozen-hydrated/sugar embedded 2D crystals. The processing of recorded images exploits the periodic arrangement of the structures in the images to extract the amplitudes and phases of diffraction spots in Fourier space. However, image imperfections require a crystal unbending procedure to be applied to the image before evaluation in Fourier space. Here we describe the image processing of 2D crystal the 2DX software system.

### 2.1. Theoretical Background

The processing of 2D crystal images involves several steps. These steps are the extraction of the lattice, the identification and correction of the crystalline deformation as well as image distortions introduced by the imaging system. This workflow is listed below and simplified in the block diagram.



1. Defining basic processing parameters
2. Calculating the Fourier transform of the image

3. Measuring the defocus in different locations in the image
4. Calculating potential specimen tilt from the defocus gradient
5. Determining the 2D crystal lattice
6. Calculating potential specimen tilt from distortions of the 2D crystal lattice
7. Determining a spotlist of significant Fourier reflections
8. Determine lattice distortion vectors and perform a first image unbending (Unbend I)
9. Iteratively refine the lattice distortion vectors, do refined unbending (Unbend II)
10. Movie mode unbending
11. Extract amplitude and phase values for each Fourier reflection from the unbent image
12. Correct the list of amplitudes and phases for the microscopes CTF
13. Determine space group
14. Calculate a final projection map from this image

These steps will now be discussed in detail, using the 2dx software package as an example.

## 2.2. Getting started

Starting in “*Project Library*” panel double-click the image that you want to process. Choose a non-tilted good quality image for the beginning. This will open a tab for the selected image in the “*Edit Images*” panel. Each image tab consists of different containers displaying data, processing parameters and image-processing routines also called as scripts. There are two major categories of these scripts which can be toggled by the button bar on the left side of the tab. These two categories include “*Standard Scripts*” and “*Custom Scripts*”. The image processing workflow of a 2D crystal is in the order of the scripts in the *Standard Scripts*. *Custom Scripts* contains other utility scripts.

When the tab is launched it automatically runs the *Init Program* script. The script checks the presence of all required external dependencies and the consistency of the image directory. Next you have to launch the *Init Files and Parameters* script, which prepares the input files for the later processing steps. The image parameters that certainly have to be altered are:

- the Magnification
- the Pixel Size on Recording Medium
- CS and kV value of the used microscope.

## 2.3. Automated Processing

The image processing workflow of a 2D crystal is in the order of the scripts in the *Standard Scripts*. To evaluate the periodicity of the crystal, the image is transformed into Fourier space, where the lattice is then determined determined in reciprocal space. This lattice is then transformed into real space, and is used to identify the aberrations from the ideal

lattice by cross correlating with an idealized reference map. The Contrast Transfer Function (CTF) is then used to correct distortions originated from the imaging. Since it is possible to process the image in automated manner, select all the scripts in the *Standard Scripts* and press the *Play* button in the control panel. The scripts are executed consecutively. The progress bar depicts, the progress of the current script and the Log panel displays the text output of the running script. You can change the verbosity level of the output in the *Log panel* with the buttons in the left corner of the panel. When all scripts are completed click on the last script of the workflow "*Generate Map*" to see its outcome. The result of the automatic unbending and CTF correction can be examined by double-clicking the *Non-symmetrized Map* in the Images panel. This will display  $2 \times 2$  unit cells.

## 2.4. Manual Processing

Since the automatic processing is not perfect you may want to improve your result manually. You can do this at any time and will still profit from the prior executed steps. In this section we will walk you through each step.

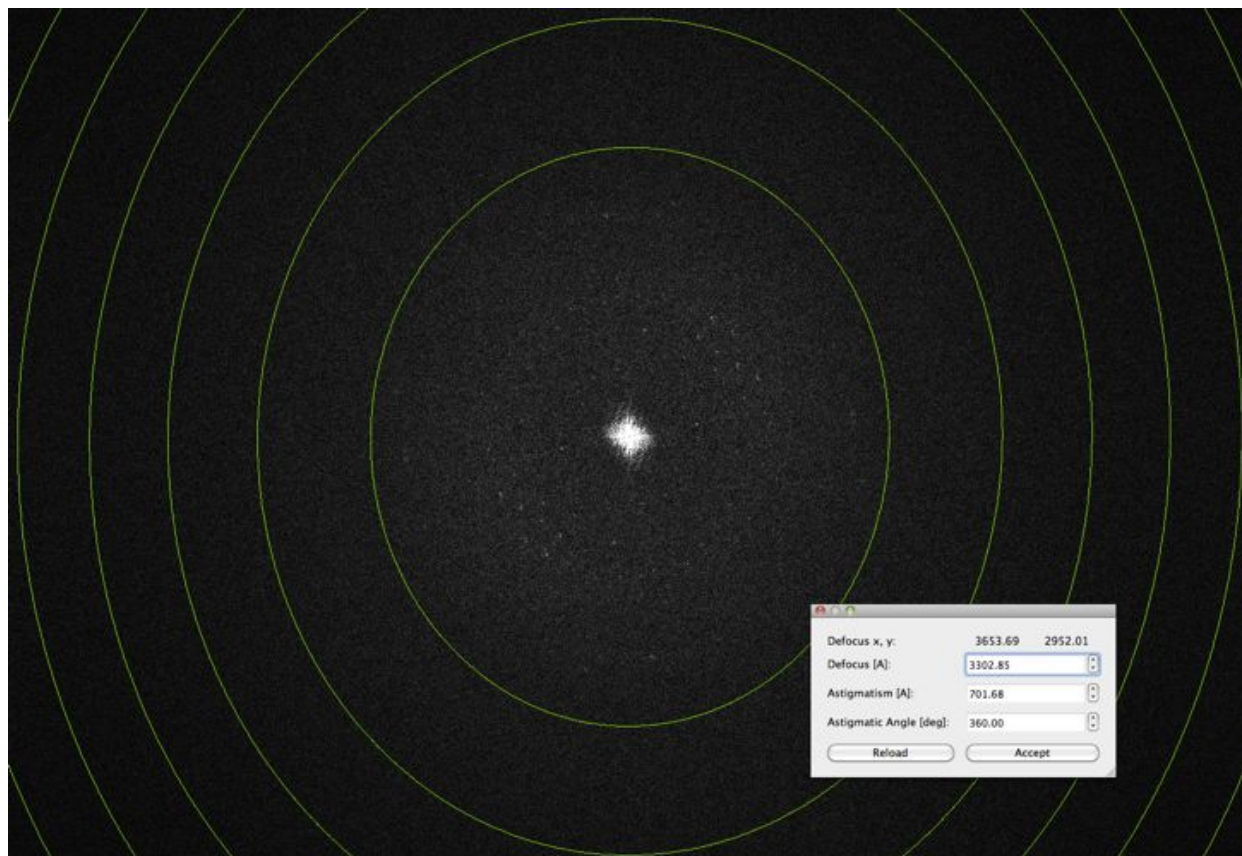
### 2.4.1. Determine defocus

The first step is to determine the defocus and astigmatism for the later CTF correction. To do so run the *Calculate FFT* and the *Get Defocus* script and double click on *FFT of Downsampled Image* in the Images container. This will open up a down sampled Fourier transformation of the original image in the full screen viewer. You can zoom in with the period key and out with coma key. Press "C" in the full screen browser or select *View CTF* in the Navigator menu to see the Thon rings. You can adjust the defocus and astigmatism in the dialog window.

If you have trouble seeing the Thon rings you may want to change the Brightness and the contrast in the full screen viewer. You can do so by selecting *Adjust Contrast/Brightness* dialog from the Navigator menu. Press *Accept* if you are happy with the parameters, or use the shortcuts "B" and "N".

If you still have problems seeing the Thon rings you can create a periodogram. To do enter the *Advanced view* in the Parameter panel of the *Calculate FFT* script by clicking the right button in the upper left corner of the panel. This will give you the full parameters list. Select *Yes* for *Generate periodogram?* and run the script again. The periodogram will show up in the Images panel and in this image you might see the Thon rings more clearly and can select them more easily. The selection is made in the same matter as before.

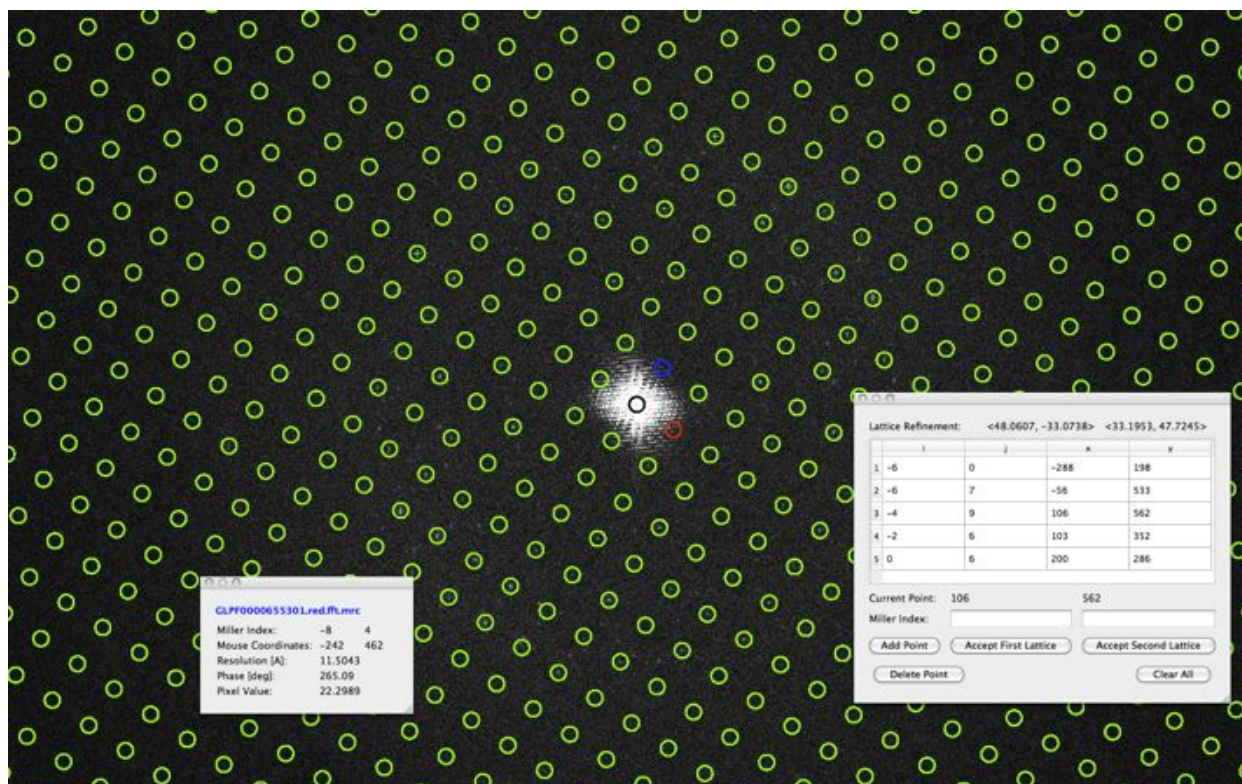
The defocus and astigmatism values will be used later in the *Correct CTF* script.



### 2.4.2. Determine lattice

The next step is to determine the crystal lattice. You can run the *Get Lattice & Tilt* script in order to find automatically your lattice. Open *FFT of Downsampled Image* in the full screen browser. You can display the lattice by pressing the "L" key. By pressing the D key or selecting *Display Coordinate Info* from the *Navigator* menu, you will get a dialog window that displays the amplitude, phase, the Miller index etc. of the pixel that the mouse pointer hovers over. To define the lattice de novo or to refine the lattice you have to enter the *Lattice Refinement Mode* through the *Lattice Refinement in the Navigator* menu.

Now you are able to add spots to your lattice. These would ideally be spots with a high resolution. With a double click the spot with the maximum intensity value in the surrounding area of your mouse pointer is selected. By pressing the Enter key or the *Add Point* in the *Lattice Refinement* dialog window the spot is added to the table. The Miller index is chosen according to the existing lattice. In case this is wrong you can alter it in the dialog window. You can activate the *Display Parameters* dialog window in the *Navigator* window. Here you can change the *Maximum Value Search Range* that is used when double clicking. Or you can change the method to use a *Gauss Fit*. To get a closer look at the area you can make a right click.



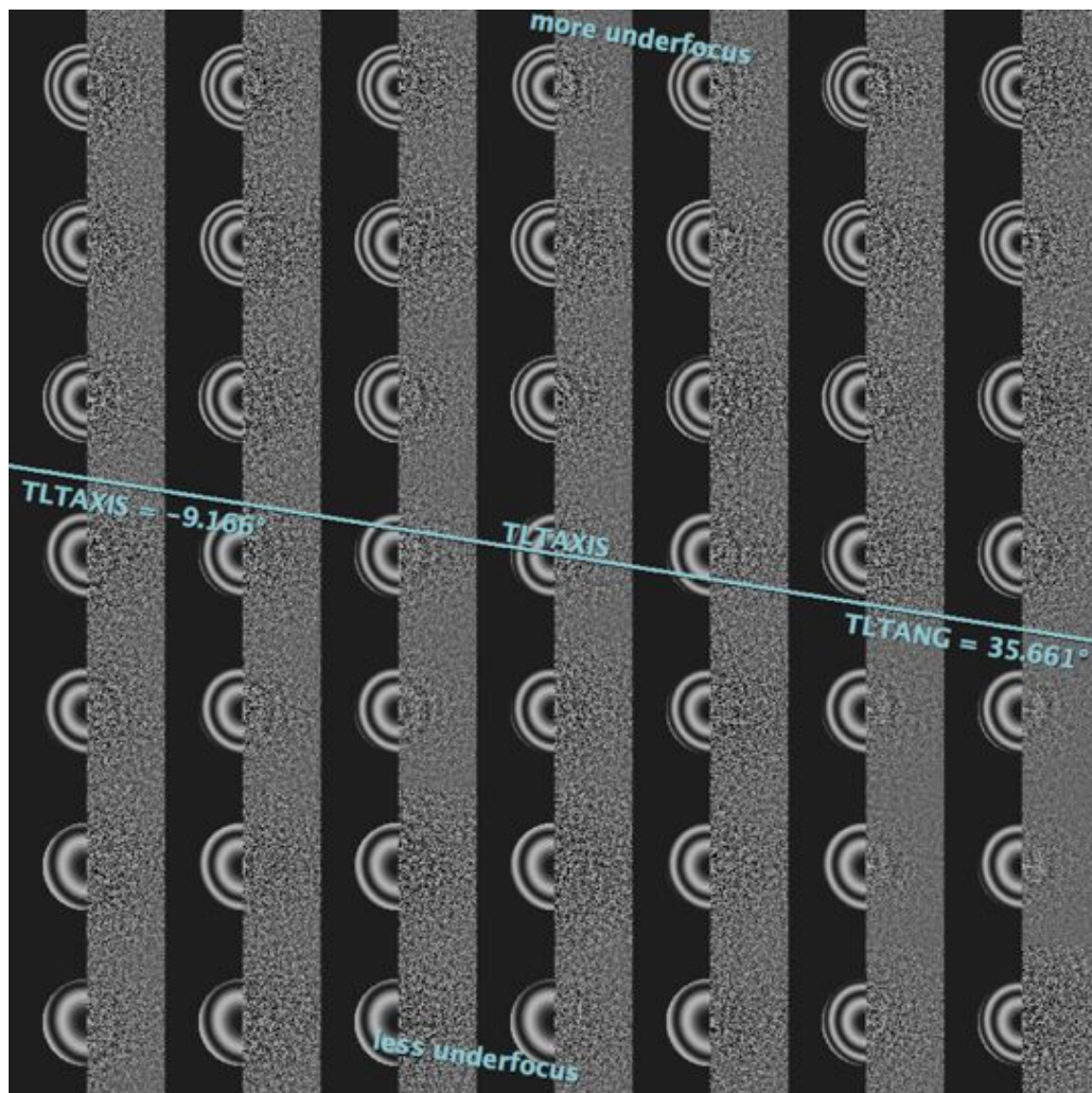
After having added spots press the *Accept First Lattice* button and you will see how the lattice is shifted to match the picked spots.

Ideally you will have only one lattice. But usually there is a second weaker lattice showing in Fourier space. This lattice can be defined and refined in the same manner as with the first lattice. You can view this lattice by pressing the *S* key. The second lattice could later be used to filter out spots that overlap in the process of unbending.

You can calculate the unit cell dimensions which is defined by this lattice by running the *Evaluate Lattice* script from the *Custom Scripts* panel.

### 2.4.3. Determine tilt geometry

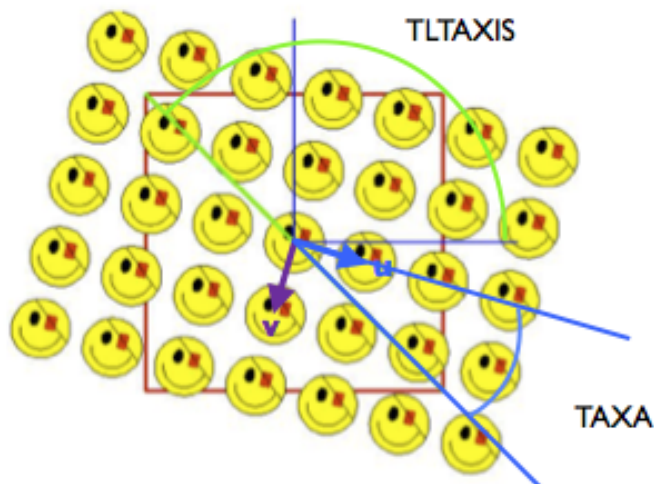
If you are processing a non-tilted image you should set *Determine Tilt Geometry* to *No* in the *Tilt Geometry Section* of the *Processing Data* panel. In case you are processing a tilted image there are two ways to determine the tilt geometry of your image. Firstly, you can determine the tilt geometry from the defocus gradient of your Fourier transform. This will be executed together with the *Get Defocus & Tilt* script. This way you determine the tilt geometry of the carbon film support, which is described by the variables TLAXIS (tilt axis of the microscope) and TLTANG (angle between tilt axis and the grid). These two variables are valid for the sample support, and have nothing to do with the crystal. In fact, they could also be determined in absence of any crystal. Please note, that for smaller tilt angles, the defocus gradient is the safer method.



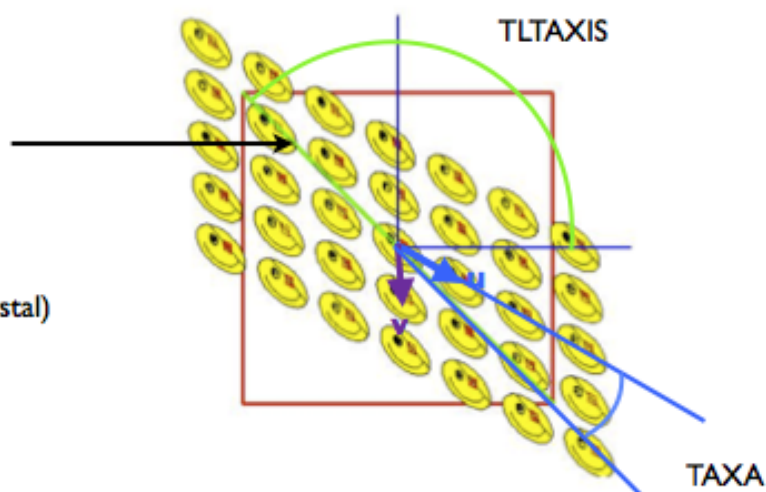
Secondly, you can determine tilt geometry based on the lattice distortion. For this purpose you have first to define the lattice, for example after running the *Get Lattice & Tilt* script. In addition to the TLTAXIS and the TLTANG it will also calculate TLAXA, which describes the orientation of the crystal with respect to the tilt axis. This lattice based tilt geometry is only reliable for larger tilt angles, which should be greater than 25 degrees. In that case, the *Get Lattice & Tilt* script will use the now determined lattice orientation to translate the defocus-based tilt geometry into the sample tilt geometry.



TLTANGL=0  
(top view of the crystal)



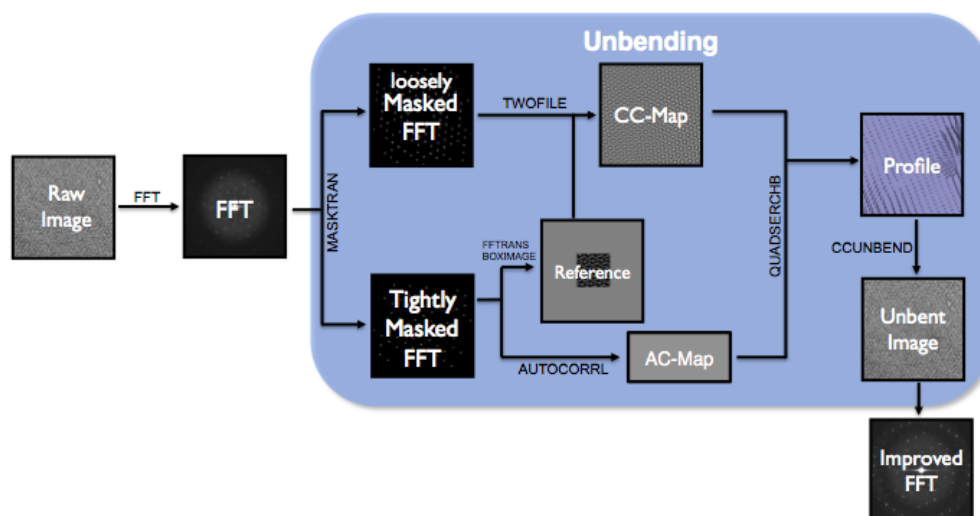
TLTANGL=45  
(top view of the crystal)



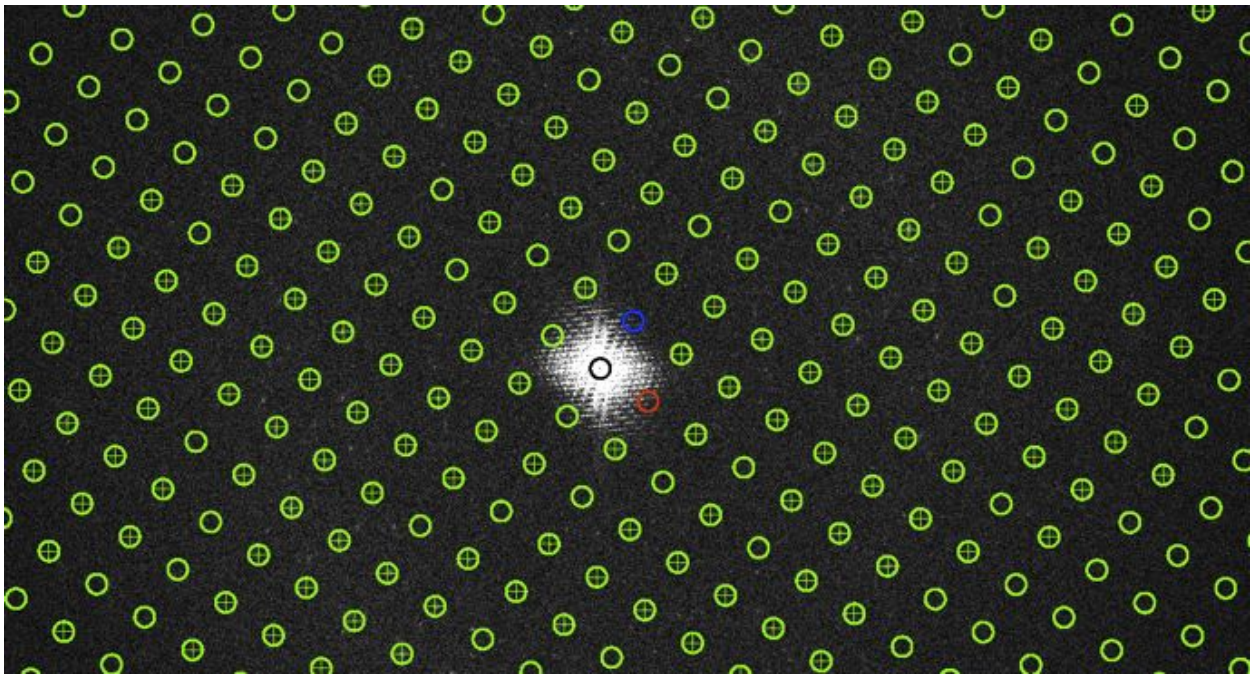
#### 2.4.4. Unbend

This chapter describes how you can correct for translational crystal distortion. Therefore you compare (= cross-correlating) a tightly masked and thus "perfect" transform (= reference) with a loosely masked transform that still contains all the information about irregularities in the lattice. However, to get the best results one first needs to determine the xy-coordinates that will serve to centre the reference area. The cross-correlation is done most conveniently in reciprocal space (twofile) because in this case it amounts to a multiplication of the loosely masked transform of the image with the complex conjugate of the transform of the reference area. Backtransformation of the crosscorrelation map is the first step to translate its information into real space disorder parameters. Consecutively, all unit cells can be searched for the exact position (= offset) of the object and the degree of similarity with the reference area (quadserchb). In practical terms this is achieved by first

correlating the object (or part of it) with itself. This so called autocorrelation map is calculated from a small part of the reference area and its centre appears as a more or less symmetrical peak whose shape depends on the actual distribution of Fourier terms in the image (autocorrl ). Finding the best fit of this central peak and the cross-correlation peaks for the individual unit cells provides the length and orientation of the distortion vectors and goodness (=height) of correlation. It should be mentioned that quadserchb only determines the translational offset for the cross-correlation peak but does not take into account any rotational disorder. Once the distortion vectors are known, the original image can be corrected by re-interpolating its optical densities to bring the unit cells onto a "perfect" lattice (ccunbend ). The resulting unbent image is Fourier transformed to give an FFT with sharper diffraction peaks. The unbent FFT is evaluated (mmbox ), yielding a list of amplitudes and phases for each lattice spot, together with an IQ value for each spot, which express the signal-to-noise ratio. The weighted sum of all IQ spots is depicted in the QVal, which can be used to judge the quality of the results of image processing.



The unbending is split into two rounds as depicted in the *Standard Scripts* panel by the *Unbend I* and *Unbend II* scripts. The needed reference for each unbending step is created from a spot list, which holds spots in Fourier space that lie on the lattice (scripts *Get Spotlist for Unbend I* and *Get Spotlist complete*). You can manually refine this spot list in the full screen view of the Fourier transformation. Enter *Spot Selection Mode* via *Spot Selection* in the Navigator Menu. With the *P* key you can identify the spots automatically selected. You may want to display the lattices when refining the spots. By clicking on a lattice spot you select it and by clicking it again you undo the selection. In this manner select now the spots that have a high signal to noise ratio. Save the spot list under *Spot List* in the *Navigator* menu.



From this outline it becomes clear that the quality of the reference area is the main factor to the success of the "unbending" procedure because any disorder present in the reference will remain uncorrected. Accordingly, improvements in the quality of the reference area by successive passes of processing make the determination of the lattice distortions and consequently, the correction of the image more accurate. In most cases the result will not get any better after 2-3 passes of image filtering and lattice unbending. However, a further improvement can be obtained once a 3D model is available, because in this case a "perfect" reference area can be created *de novo* by calculating a back projection of the model for the "precise" imaging conditions (maketran). Further parameters you can play around with are the radius of the loosely masked Fourier (*mask*) and the diameter of the reference (*box*).

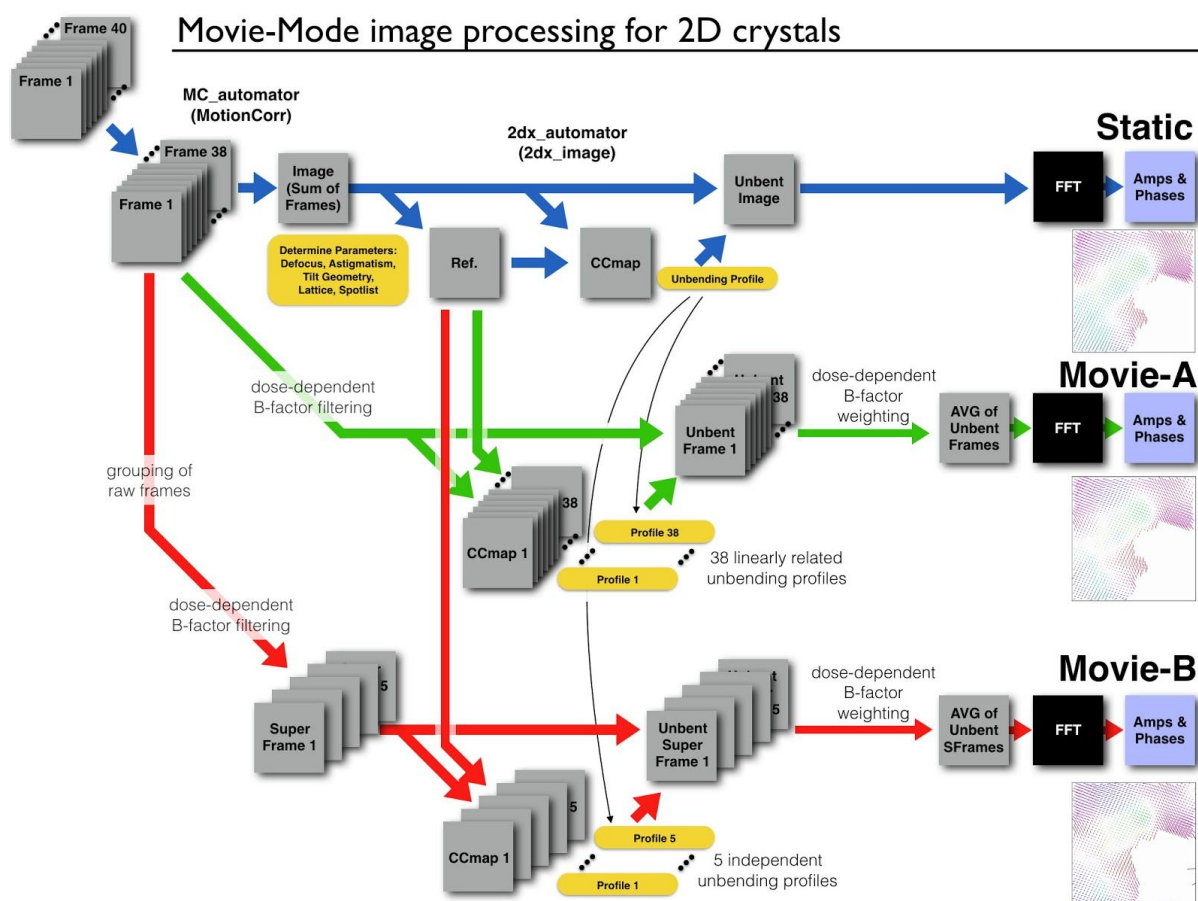
#### 2.4.5. Movie mode unbending

As the 2D crystals have to be imaged under sample tilt, the specimen drift during data recording is immense. The carbon film will move upwards in the microscope, which for a tilted sample means a lateral movement. If in addition the 2D crystal is not firmly attached to the carbon film, it will move erratically in varying directions in the image.

Movie data are recorded with a direct electron detector in dose-fractionation mode. A raw movie is first to be whole-frame drift-corrected. This can be done by a program such as MotionCorr, MotionCor2, Unblur, or with our in-house program ZORRO. ZORRO is being integrated into 2dx, which should allow to load a raw, non-drift-corrected movie into 2dx, and whole-frame drift-correct it with ZORRO directly from within 2dx.

Once, drift-corrected movies are available in 2dx, they still need to be unbent with the so-called Movie-Mode unbending. Here, the individual unit cells of a 2D crystal are each unbent independently, so that varying drift within a 2D crystal can be corrected.

2dx offers two different algorithms for movie-mode unbending, termed Movie-A and Movie-B.



A raw movie (top left, here with 40 frames) is first to be whole-frame drift-corrected (in this example using MotionCorr, leading to 38 remaining frames). This drift-corrected movie is then averaged (Image, Sum of Frames), which is then conventionally unbent, resulting in a final map called “Static”, top right.

The first movie-mode algorithm and script, termed Movie-A, then tries to unbend each of the 38 frames individually. This, however, usually fails, since the single frames don't have enough signal-to-noise ratio (SNR) to allow the unbending. Help comes from using as unbending reference the reference from the Static unbending. Another help comes from the algorithm, unbending the frames consecutively: Starting with the unbending profile from the Static run and applying that to a middle frame, the algorithm “crawls” from one frame to the next frame, learning from the previous frame and only allowing smooth and small variations in the unbending profile from one frame to the next one. These constraints on the

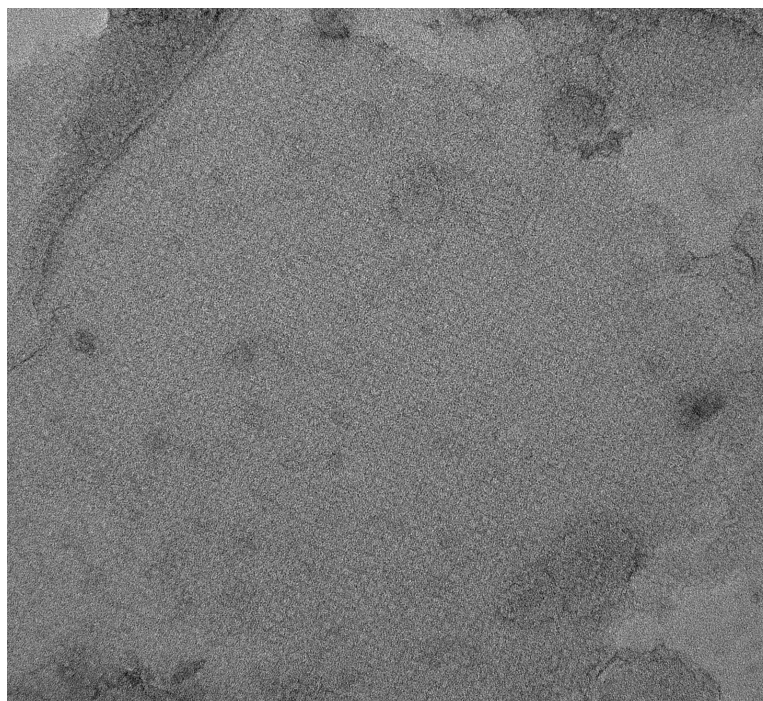
unbending profiles mean that a sudden jump during the movie recording cannot be accommodated. But this still allows to deal with the very low SNR of individual frames. 2dx will ask for several parameters in the Movie-A script, the most important being movie-smoothing (“Movie-Mode drift smoothing over how many neighboring lattice nodes”). This allows you to define, how many of the neighboring unit cell drift vectors should be averaged to replace a specific unit cell drift vector, if that unit cell’s drift vector deviates from that of the neighbors by more than a given `movie_drift_threshold` (“Movie-Mode drift smoothing threshold”).

Movie-mode also applies a dose-dependent B-factor filter to the movie series: Frames are resolution limited with a B-factor filter (i.e., with a Gaussian fall-off of amplitudes at higher resolution), whereby the B-factors are adjusted so that the first frames are used with their full high resolution, while later (burned) frames are used only with their low resolution. These B-factor filtered frames are then unbent. After unbending, the frames are summed up in Fourier space, and the resulting final Fourier sum is inversely B-factor filtered. This means that high-resolution terms are now up-weighted, so that in the end all resolution frequencies have the original intensity restored.

The second movie-mode algorithm and script, termed Movie-B, groups a certain number of the original (whole-frame drift-corrected) frames together, forming so-called “super-frames”. In the above example, the 38 frames are grouped into 5 super-frames, each consisting of 8 original frames. So, frame 1-8 are averaged into super-frame 1, frame 9-16 into super-frame 2, frame 17-24 into 3, frame 25-32 into 4, and frame 33-38 into super-frame 5. These 5 super-frames then show enough SNR, so that they can be unbent independently, but still exploiting the high-contrast Reference from the Static unbending run. Also in Movie-B, dose-dependent B-factor filtering before and inverse B-factor adjustment after unbending is applied.

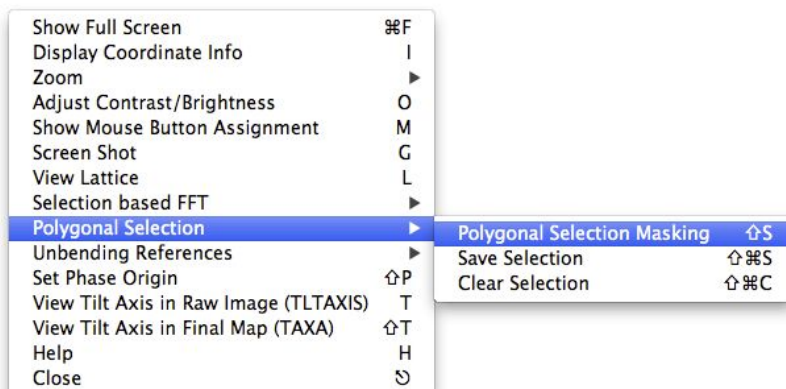
#### 2.4.6. Crystal masking

In real world applications the crystal almost never fills the entire image.



Therefore it does not make sense to use regions where the crystal is not regular or where we don't have any crystalline structure. 2DX offers a tool with which you can manually mask an image. The following walkthrough explains how you can mask the crystal.

1. There are multiple ways how you can mask your image. You can mask the crystal automatically by selecting the option *Do automatic masking of 2D crystal* in the Parameter panel of *Unbend II*. Alternatively, you can mask your crystal manually. Open the original image from the result panel of the initialization script and mask the image there. For many applications this does not work, as the crystal is not visible by eye. Thus we advise to mask in the file *XCF Map for Manual Merging* which can be found in the result panel of *Unbend II*. This file shows you the cross-correlation profile generated during unbending the crystal. The height of the peak corresponds directly to the local quality of the crystal. In regions with no crystalline structures we will not find any peaks. As usual the file is opened in the fullscreen mode by a simple double click.
2. In order to be able to mask the image you have to activate the *Polygon Selection Mode* by right-clicking in the image and selecting "Polygon Selection > Polygon Selection Masking".



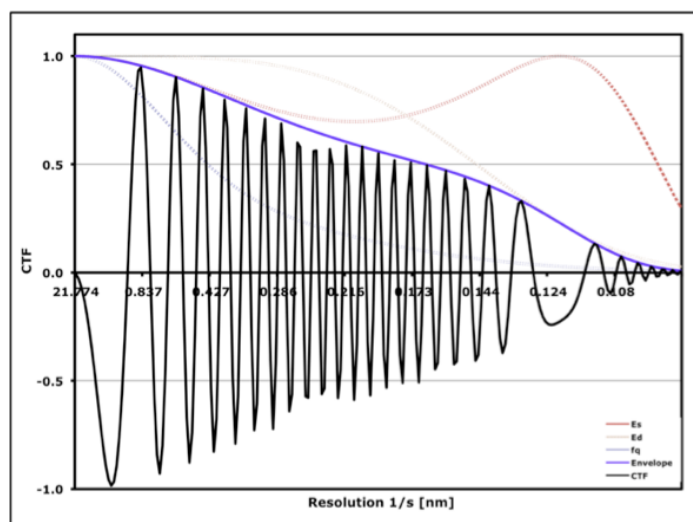
3. Now you can mask your crystal by defining the borders with a bunch of left mouse clicks. A double-click will automatically close the selected polygon. Note that you can delete the current selection by selecting "Polygon Selection > Clear Selection" from the menu appearing when right-clicking on the image.



4. Once you are happy with your selection you can save the current selection by selecting "Polygon Selection > Save Selection" from the menu appearing after another right-click.
5. In order to apply your polygon selection you have to run the custom script *Mask Crystal from Polygon*. This script will replace all the pixels outside the polygon with the average grey value inside the polygon. In the future calculation this new image will be used as input image.
6. After masking the crystal from the saved polygon you have to save the current configuration by means of clicking on the save button in the header of the 2DX graphical user interface. Afterwards you have to rerun all the standard scripts, except *Get Defocus & Tilt* and *Get Lattice & Tilt* as neither the defocus nor the lattice are affected by the masking. This will generate a density map uniquely from the data points inside the selected polygon.

### 2.4.7. CTF Correction

The next step consists in an initial correction of the phase data for the effect of the contrast transfer function (ctfapply). The CTF is a modulation of the scattered waves by the objective lens, which results in periodical contrast reversals across the image. In Fourier space this corresponds to a phase shift by  $180^\circ$  where the CTF is negative. Because this modulation will be different for each image (due to their different amounts of defocus) the phases must be corrected to allow the combination of data from several images. The visual manifestation of the CTF is a characteristic pattern of alternating light and dark bands (= Thon rings) in the diffuse diffraction patterns of amorphous materials. The dark areas correspond to frequencies that are not or only poorly transferred and hence do not contribute to the formation of the image. Vice versa, good transfer is achieved in the bright areas. In order to correct for CTF in your images run *Correct CTF* script in the *Standard Scripts* panel. In addition, this scripts gives you an IQ-Plot.





### 2.4.8. Space Group Determination

CTF-corrected data can always be combined in "plane group p1", i.e. assuming no symmetry at all. However, if the specimen does exhibit a certain symmetry, data handling and refinement becomes a lot easier because symmetry imposes constraints on the molecular transform. For instance, a simple twofold axis of symmetry means that the densities within the unit cell are related by a 180° rotation about this axis. Consequently, the density distribution will appear symmetrical in non-tilted image data (= projection density map). To encode this in reciprocal space requires all the projection terms to be symmetric about the origin, i.e. they are cosine waves that can only be shifted by 0 or 180° with respect to the phase origin. The introduction of redundancy is a further advantage of symmetry. For instance the terms (1 0 0), (0 -1 0) and (-1 1 0) are all different if no symmetry is present. However, if the specimen displays three- or six-fold symmetry these terms have identical phases and amplitudes, i.e. they are symmetry related, describing the same structural feature. In other words while the data of a single image only contribute a single measurement for each reflection in the first case, three independent measurements of the "unique" reflection (= 1 0 0) are obtained for three- or six-fold symmetry. In addition, the actual phase values allow discriminating further between three- (p3) and six-fold (p6) symmetry. Since the latter has intrinsic twofold symmetry all phases would be expected to adopt 0 or 180° values. However, this is not true in p3 where like in p1 no phase constraints are applicable. The agreement between symmetry related reflections is independent of the CTF as long as the image has no significant astigmatism because they have the same distance from the transform origin. In summary, the relationships between the phases obtained from images of non-tilted crystals allow an unambiguous determination of the specimen's symmetry because each of the 17 plane groups has a unique set of constraints.

To determine the space group of your crystal run the *Get Spacegroup & Phase Origin* script in the *Custom Scripts* panel. It will calculate the internal phase residuals for all space groups and thereby choose the optimal space group and the corresponding phase origin. The space group and phase origin are then in 2DX as entries for *Symmetry and Phase Origin*. To find the right space group it is better to set the *Resolution limitation for phase origin search* to a lower resolution than with what you process your image. In case you know the space group of your crystal you can also use this script to just find the according phase origin for the image. Note that only projection images of non-tilted crystals show symmetry.

Optionally you can set your phase origin manually. This way you could process all your single images to a similar phase origin, making it later easier to find a common phase origin in 2D merging. For this purpose run the *Set PhaseOrigin Manually* script in the *Custom Script* panel. Open the *Old p<X>-symmetrized Map* in the *Image* panel. Position your mouse cursor on the wished phase origin and press shift "P". The needed phase origin changes will be transferred to the *Parameter* panel. To apply the changes you have to rerun the script once again.

```
#####2dx_allspace#####
Symmetries to test = ALL
Stepsize and Phase Search Array Size = 3 , 121
IQ Max = 5, Resolution Max = 10
```

SPACEGROUP	Phs.Res. (#) v.other spots (90 random)	Phs.Res. (#) v.theoretical (45 random)	OX	OY	TX	TY	Target
1 p1	21.1	170	15.3	170			
2 p2	33.7!	85	16.8	170	-21.1	123.9	30.6
3b p12_b	73.2	65	41.4	6	-146.6	-180.0	21.6
3a p12_a	74.4	67	16.6	10	-180.0	124.1	21.8
4b p121_b	25.9!	65	27.7	6	-110.7	-141.0	21.6
4a p121_a	11.2*	67	10.2	10	-153.0	-145.7	21.8
5b c12_b	73.2	65	41.4	6	-146.6	-180.0	21.6
5a c12_a	74.4	67	16.6	10	-180.0	124.1	21.8
6 p222	62.0	217	16.8	170	-20.9	-56.1	24.8
7b p2221b	54.7	217	37.4	170	123.4	34.3	24.8
7a p2221a	61.4	217	39.7	170	69.1	-146.5	24.8
8 p22121	24.5*	217	17.0	170	-20.7	-55.7	24.8
9 c222	62.0	217	16.8	170	-20.9	-56.1	24.8
10 p4	68.5	189	39.5	170	-111.9	33.3	25.4
11 p422	75.6	415	41.1	170	130.8	123.6	23.1
12 p4212	66.0	415	17.0	170	-21.1	124.3	23.1
13 p3	63.9	74	--	--	78.0	122.7	21.1
14 p312	72.4	195	29.7	18	80.9	-3.3	21.6
15 p321	66.7	199	39.8	26	77.8	123.2	21.8
16 p6	59.3	233	16.8	170	158.8	-56.1	24.6
17 p622	72.8	479	43.9	170	-165.6	-120.7	22.8

```
-----
* = acceptable
! = should be considered
^ = possibility

OX,OY = best phase origin for this symmetry
TX,TY = best beam tilt for this symmetry
Target = target resid. based on statistics, taking Friedel weight into account

==== Best SpaceGroup is p22121 =====
==== 2dx allspace - normal end. =====
#####2dx_allspace finished.#####
```

### 2.4.9. Map Generation

Final step of 2D image processing is the generation of the projection map. You can create an unsymmetrized map by running the *Generate Map* script in the *Standard Scripts* panel. Alternatively you can generate a symmetrized map by applying the selected space group. For this run *Generate Map* script in the *Custom Scripts* panel. The latter will calculate a table of phase residuals during the symmetrization, from which you can judge the overall quality and resolution of your image.

Please note that you have to set the parameter *Invert contrast to the final map again* to *No* in case you are processing a negative stained image, and to *Yes* in case of a cryo image.

## 2.5. Setting config as Project Default

Once you are happy with processing one image you can set the parameters from this image to all the images. This can be done in following steps.

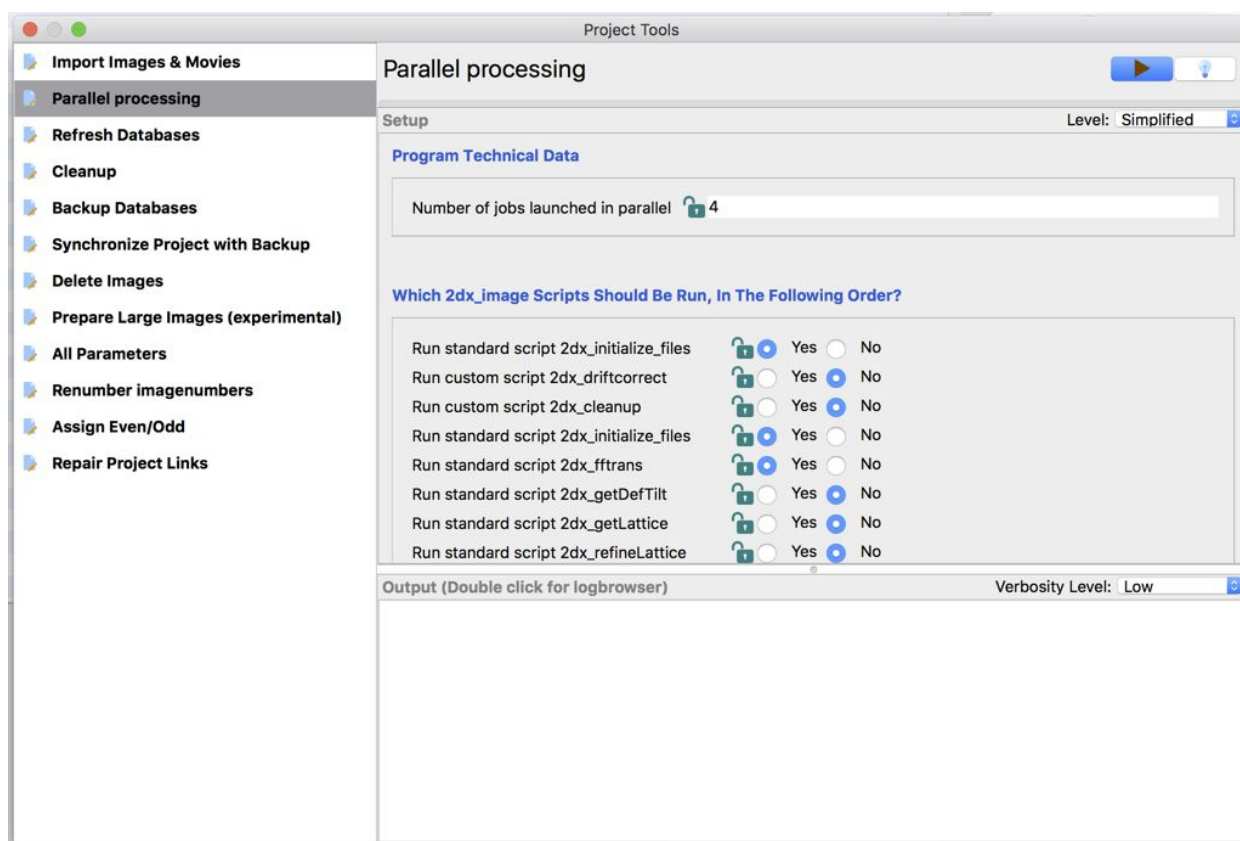
1. To save this parameters as default for the project new imported images click the “*Save as Project Default*” button on the top of the tab located on the right of the

script title. So all the new imported images would have this new config file as default.

2. But this would change only the project master config file. To copy this config to others we can use “*Reset Image Config*” script from “Project Tools”. First select all the images which you want to reset and then run the script from “Project Tools”.

## 2.6. Parallel Image Processing

It is possible to run the automated processing of images in parallel. This can be done using the “*Parallel processing*” script in the “*Project Tools*” panel which can be activated using the top left button in the main 2dx\_gui.



Set the number of parallel processes and select all the scripts which needs to run in the setup panel. After the setup hit the *Play* button. The progress can be monitored by the progress bar and in the output panel.



### 3. 2D Merging

In the previous sections we have seen how to process individual images of 2D crystals. From a single micrograph we generate a list of spots containing their amplitudes and phases in Fourier space. This section describes how to merge the projection maps from each micrograph of a non-tilted specimen in order to produce one common projection map. This is similar to creating class averages in single particle electron microscopy, where projections of particles with the same orientation are averaged together.

In contrast to single particle classification in the context of 2D crystals we already know the projection direction from the electron microscope, all under the assumption that the crystals are perfectly flat. The central section theorem gives us the mathematical backbone to consider each image created by a transmission electron microscope as a projection of the crystal. Thus all images of non-tilted 2D crystals represent a projection of the crystal in  $z$  direction. In the previous section we have seen how filtering the Fourier transform of 2D crystal results equals averaging the proteins in real space. The task of 2D merging is to average the Fourier components of the images of untitled crystals together.

To be able to average the densities gained from each image one has to align the images first. They have to be in register with each other which is best done by aligning them to a common reference. As we see later in the practical part the reference is just one of the images chosen by the user. The alignment can also be understood as setting a common origin.

We again make use of the Fourier theory for this task. For single image processing we only considered at the amplitudes of the Fourier transform and the lattice it spans. This lattice always had a common origin and approximately the same dimensions, only the orientation varies across the images. This is due to that the magnitude of the Fourier Transform does not change even when the image is shifted. Therefore averaging the amplitudes is just averaging the diffraction spots with the same miller indices. The phases of the Fourier transform though determines the origin of the image in real space. Thus aligning all the images results in shifting the phases of each image to match with the reference.

The last step of 2D merging is averaging. The simplest way would be to take the mean of the values for each unique reflection depicted by a Miller index. But the program `AVRGAMPHS` written by Richard Henderson weights each reflection based on its IQ value. Hence each single diffraction spot of every image contributes to the average according to its signal-to noise ratio. For the user this has the advantage that even images of bad quality can be merged without making the resulting 2D map worse, additionally one can still benefit from the few spots that have a good IQ value. The final merged 2D map is then the inverse Fourier transform of the list of averaged amplitude and phases.

### 3.1. 2D Merging in 2DX

Having processed all the images of non-tilted specimen we are ready to merge them to an average projection map. We will use the *Merge Tool* panel to do so.

1. Open the *Merge Tool* panel
2. Select the *2D merge scripts* panel from the left bar.
3. We want to align the images to a common phase origin. Therefore we have to select one image as reference. Usually the image with the highest QVal and/or lowest phase residual error is used. The QVal can be read from the "QVal2" column in the project panel. To make it easier one can also sort the images by the this attribute by clicking on the title of the column.
4. When choosing the reference make sure that the phase origin of the image was chosen accordingly the present space group. If the image suffers from a wrong phase origin you should set the phase origin manually.
5. The appropriate reference should be selected with the tick mark from the "Project Library" panel. Make sure no other images are selected.
6. Before creating the reference one should to choose the resolution the data on which the alignment process can rely on. This resolution is specified by the parameter *Resolution of the merged dataset for the reference*. We set this value initially to 15 Å.
7. Make sure the *Symmetry* is set correctly in the processing data panel because the selected symmetry will be applied to the reference, i.e. correction of symmetry-restricted phase values and removal of symmetry-forbidden reflections.
8. Now run the script *Merge Selected Images*. This script merges the data of the selected image into a reference file. Several output files are created as listed in the Images panel. The essential ones are the merge.apf, which is the list of merged amplitude and phases, and the MTZ file, which is then used as reference named *MTZ: Merged full reciproc. space 2D data for reference*. Double clicking the MTZ file will show the content in your web browser. The file mainly consist of a reflection list as already seen in the APF file. Every reflection is depicted by index (H,K,L), where the L is set to 0 in the 2D merging, followed by amplitude and phase, which have been CTF corrected and symmetrized. The MTZ file also contains information about the crystal unit cell dimensions, the symmetry, the resolution range, etc.

Now that we have created a reference we can start aligning the non-tilted image data.

1. Uncheck the image you have selected to be the reference and select all other images recorded from non-tilted specimen that you want to align.

**Tip (Automatically selecting the images):** The images from the "Project Library" can be automatically selected

Automatic Selection

Parameter to use	taxa
Minimum value	0
Maximum value	90
Positive+negative values	Yes

Change Selection

using the “*Automatic Selection*” tool located at the bottom right of the “Project Tools” panel. Choose the parameter to be used, provide the minimum and maximum values of the range. Finally hit “*Change Selection*” once you are done. This will automatically select all the images with selected parameter in provided range.

2. With the script *Refine Once* you can now align the selected images to the reference. All you need to do is to specify the range over which the phase origin search should be performed. Unless you have already manually selected the phase origin in all of the projection maps, we advise you to initially cover the entire crystal unit cell i.e. a search range of  $360^\circ$  phase angle. The search range is defined by the parameters *Stepsize of the phase origin search* and the *Number of steps in the phase origin search* in the *Merging* section. In a first phase origin search you should use a coarse step size of  $6^\circ$  which with a number of 60 steps will go from  $-180^\circ$  to  $174^\circ$ . In later stages you should decrease the step size to a smaller value in order to perform a local refinement of the phase origin.
3. Run the *Refine PhaseOrigins Once* script. This will refine the phase origin for all selected images with the help of the MRC program *origtilt*. The last phase origin change for every image will be listed as *phaori\_last\_change* in the Results panel as well as in the *PhaOri Change* column.
4. Now we visually inspect the phase origin change for each projection map by running the *Generate Image Maps* script. Therewith the projection map are updated with the refined phase origin and is reflected in the Image Preview panel. You can scroll through the images in the Images panel or the Project panel.

**Tip (Limiting library to selected images):** If you want to see only the selected images in the Project Library panel you can select the option “*Show checked images only*” from the “*Library Tools*” located on the right of the panel. The same menu option can be used to switch back to view all images.



5. Run *Refine PhaseOrigins Once* again with the current step size until the phase origin of all refined images does not improve anymore. Depending on the step size it might happen that the optimal phase origin is in-between the two considered phase origins and will jump back and forth. This is also a sign that you have reached the optimum for this step size and should decrease it for the next merge and refinement runs.
6. Besides the updated parameters in the results panel, the script also produces output files in the images panel. If you are curious how *origtilt* is called to refine every single image you should have a look at *CSH: refinement script*. The output of *origtilt* might be more interesting and is in the text file listed as *LOG: origtilt B output*. This file should be considered when evaluating the refinement process. It contains the list of reflections for all refined images. The most important value in this file though is

the refined phase origin. It is followed by a table of resolution ranges with the associated phase residual for the reflections in that range. This allows the verification up to what resolution the refinement is trustworthy. One thing to keep in mind is that for the coarse refinement we limited the resolution of the reference. After the resolution table in *LOG: origiltk B output* a cross-correlation map is plotted through a matrix where each entry represents the normalized cross-correlation value between the refined image and the reference. It should contain a clear peak which should end up in the center after the last rounds of refinement.

7. After the first cycle of refinement we should update our reference with the refined data. Therefore include the image you initially used as a reference by adding it to the selection. Now run *Merge Selected Images* again to create a new reference.
8. If you set the *Symmetry* in the Processing Data panel and the symmetry is above space group P2 you will get a table named "Phase Residual In Resolution Ranges". You should consider the phase residuals as measure to what resolution you can trust your data and adapt the *Resolution of the merged dataset for the reference* accordingly. The resolution ranges of the table list the phase residual up to the *Upper Resolution limit plus 1Å* further.
9. With the new reference at hand we can perform a finer refinement of the phase origin. Hence the *Stepsize of the phase origin search* should be decreased to e.g. 0.5Å. In any case make sure that the step size together with the number of steps at least span the range of the previous step size (in our case 6Å).
10. Now run *Refine Once* again for the refinement until the phase origin for each image does not change anymore i.e. *PhaOri Change* will be zero. The cross-correlation maps in *LOG: origiltk B output* should now show sharp and centered peaks.
11. Until now we have always executed the merge and refine process separately. But when you are at the point where the images merged to a reference are the same set as the images being refined i.e. the selection does not change, you can also use the *Merge & Refine* script and do merging and refining iteratively for the selected number of rounds.
12. When using *Merge & Refine* script one should have a look at the parameters *phaori\_last\_change* and *MergePhaseResidual* in the Results panel. As we have mentioned before the phase origin change should decrease with each iteration and ideally end up being zero. The measure of how good the refined phase origin is quantified *MergePhaseResidual*. This value will also decrease in the process of refinement. If the value remains around 90° then the alignment is not working.
13. For more in-depth knowledge about your merged data set there is the option *List Reflections into logfile (LIST)*. If this is enabled a separate log-file is created called *LOG: reflections after origiltk*. This file is the complete list of reflections from every merged image. This means that for every H,K index the available values from the contributing images is listed. The values associated with each reflection besides the H,K index are the amplitude, phase, image number, IQ, film weight, background

amplitude and CTF. Appending these values are dashed lines followed by the IQ value again. The number of dashed lines is inverse proportional to the IQ value i.e. a reflection with an IQ of 1 has the most dashed lines appended. This helps to identify the good reflections even in a large list. The reflections with IQ values of 1 to 4, even have the amplitude and phases repeated at the end. This should make it easier to detect reflections with data differing from the amplitude and phases of the same reflections stemming from other images. This deviation can be a sign for a wrong CTF correction in that image, therefore you should have a look if the defocus was detected correctly in this image.

14. If you did not add all the non-tilted images yet or have gained new images from the microscope you should use the existing reference to align the novel images in the same fashion by jumping back to step 1.

Once all the data of the non-tilted images are aligned the last step is to merge them all together to one.

1. As described above the final alignment should also be visually inspected by running the *Generate Image Maps* script again and then scrolling through the images.
2. Running the *Final Merge* script is similar to the previous merge steps, except that we will not use the result as a reference for further alignment, rather as resulting average 2D map. Hence it is of great importance to chose the *Upper Resolution Limit (RESMAX)* carefully. One should only trust the data up to a resolution where the phase residual are low. If you are looking at the table of phase residuals a good estimate if phase residual reflects meaningful data can be given by the following relation

$$phase\ residual \leq 90 - (90/\sqrt{n})$$

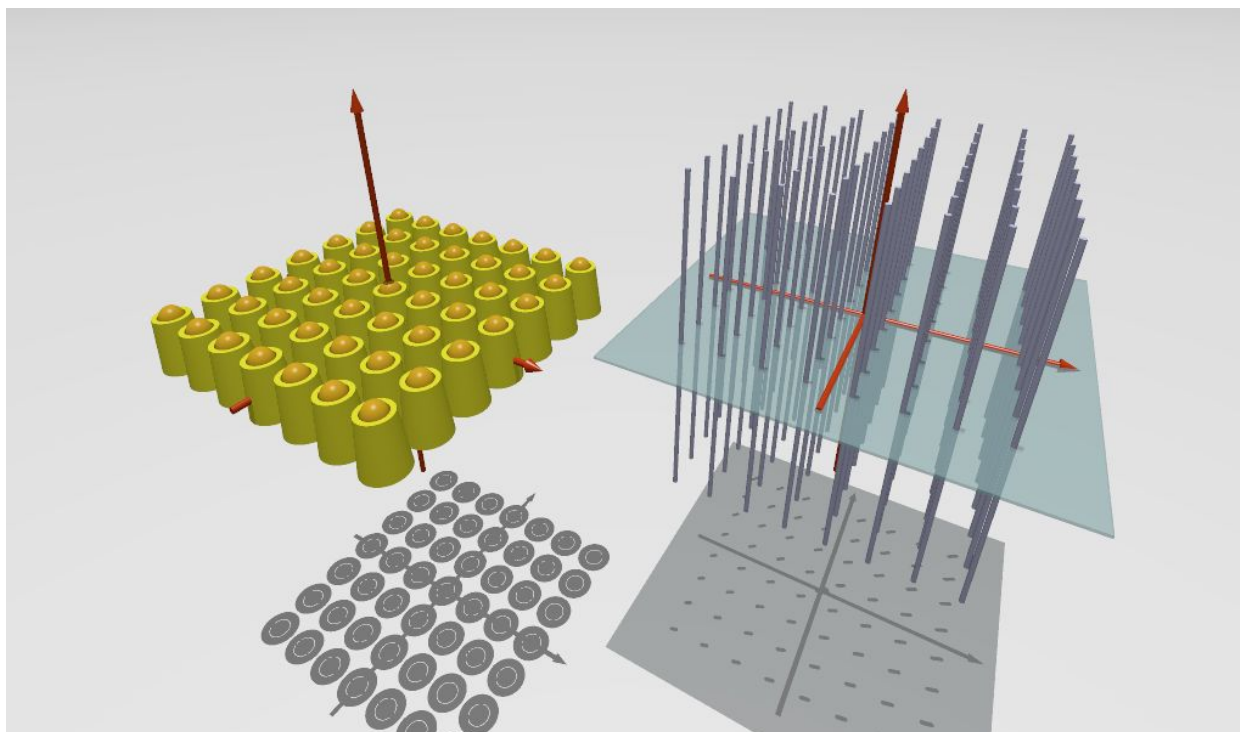
where  $n$  is the number of spots in a given resolution range. Final merge also creates two kind of 2D maps: one in form of an MRC image and the other in form of a contour plot. For our example data set where we set the symmetry to P4 and the files the images are named *p4-symmetrized final 2D map* and *PS: p4-symmetrized final 2D map plot*.

**Tip (Saving and Loading selections):** Selection of checked images can be saved using the button “Save checked list” in the right toolbar in the “Project Panel”. These selections can be later loaded using the button “Load checked list”.



## 4. 3D Merging

The general goal of electron transmission imaging techniques such as Electron Crystallography, Tomography or Single Particle reconstruction is to gain a 3D computer model by integrating sets of 2D data recorded by an electron microscope. The common mathematical tool that all these techniques share is the Central Section Theorem. The theorem states that the projection along the  $z$  direction of a three dimensional volume contains the same information as the central slice in the reciprocal space ( $z^* = 0$ ) of the 3D-Fourier transform of the volume. This means that we can gain a full sampling of the Fourier Transform of the object by the projections along all spatial directions. Accordingly, the 3D reconstruction become just a Inverse Fourier transform. The theory of Electron Optics allows to some approximation the use of images recorded by a Transfer Electron Microscope (TEM) as a projection of the sample. This means that with sufficient images of samples imaged in different orientation to the electron beam, one can simulate the sample's three dimensional density distribution.



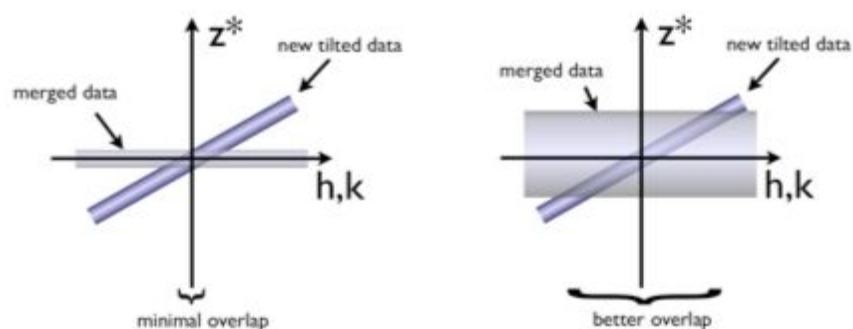
We have seen that the Fourier transform can be used to harvest the repeating signal of the 2D crystal. But since we are dealing with 2D crystals the periodicity is limited to the  $x,y$ -plane. The crystal is ideally only formed by one layer in the vertical direction. This results in continuously distributed Fourier component in vertical direction as illustrated. The diffraction spots we have seen in 2D with the index  $(h, k)$  now become so called lattice lines

$(h, k, z^*)$  where  $h$  and  $k$  still correspond to the same Miller index, but  $z^*$  defines the position along the lattice line.

Projections of tilted 2D crystals now produce diffraction spots, where each reflection corresponds to the lattice line  $h, k$  and height  $z^*$ . The height corresponds to the intersection of the tilted plane with the lattice line  $h, k$  and thanks to the central section theorem this can be calculated from the tilt geometry of the crystal. So each recorded image produces a set of amplitude and phases with the coordinates  $(h, k, z^*)$ . The goal is now to densely sample the lattice lines by integrating as much data as possible with varying orientation and tilt angle. This allows us to sample the Fourier transform of our crystal to a certain degree, but due to instrumental restrictions of the TEM, where can not record an image of a any specimen tilted higher than  $\pm 70^\circ$  there will always be a missing cone.

### 4.1. 3D Merging in 2DX

3D merging in 2dx in follows the same workflow as for 2D merging. The problem is still merging data from different images into one common data set. Therefore finding the common phase origin is the task at hand. But in contrast to 2D data where each reflection  $h, k$  and height  $z^* = 0$ , the amplitude and phases of a tilted sample are usually associated with a reflection at a height that is not zero. These measurements belong to a tilted plane in the 3D Fourier space, where the tilt geometry is defined by the tilt angle (*TANGL*) and the orientation of the lattice on that plane (*TAXA*). If we would now try to refine the phase origin of the tilted image data with non-tilted dataset, then the data points available for comparison would only be the one that lie on the tilt axis. Those measurements are too few to give a sufficient alignment of the tilted data. To deal with the alignment problem of the first tilted data we need to increase the range in  $z^*$  direction, so that a wider region for allowed comparisons between the non-tilted and the reference data exists as illustrated.



In the following steps we will show how 2DX can merge data from 2D crystals with different tilt geometry to a 3D reconstruction of your protein structure.

**Tip (Backup and Synchronize):** We advise you to always backup your project before you start with 3D Merging, using the script *Synchronize with Backup* from “*Project Tools*”. This script allows you to submit your project data to a different directory.

**Tip (Automatically selecting the images):** Also save the merged map from 2D merging with help of the *Copy Merged Dataset* custom script. This script allows you to copy the last merging result to any of the given 10 registers.

1. Select the *3D merge scripts* panel from the left bar.
2. Select all processed images that belong to the lowest tilt group ( $TANGL < 30$ ), except the non-tilt data. Preferably, the selection should include ten or more projections with reliable measurements. The projections of sample with low tilt is determined by defocus gradient which can be unreliable to some extent, therefore it is essential that the handedness is correct.
3. We have mentioned the critical step of aligning the first tilted dataset to a 2D merged map. So before running *Refine PhaseOrigins Once* limit the selected tilted data set to 15 Å with the parameter *RESMAX*. In this first 3D refinement step it is indispensable to have sufficient reference points ( $>20$ ) for the phase origin determination. This can be reached by increasing vertical  $z^*$  tolerance along a lattice line, which is defined by the processing parameter *3D: zstarwin* in the 3D Merging section. In the beginning of 3D merging set this parameter to 0.2 or something higher. This measure depends on the vertical thickness of your crystal defined by the parameter *ALAT*. When refining the highest tilt data *zstarwin* should be decreased to  $\frac{1}{2*ALAT}$ . So for a 200 Å sample we eventually set *zstarwin* to 0.0025, just to give you a feeling for the value range.
4. As in 2D merging we start out with a step size of 6.0 (*Stepsize of the phase origin search*) and 60 steps (*Number of steps in the phase origin search*).
5. Run *Refine PhaseOrigins Once* until the phase origins do not essentially change anymore.
6. Examine the refinement of the tilted data to the non-tilted by looking at the *origtilt* output *LOG: origtilt B output*. You should be familiar with this file from 2D merging, but in contrast to 2D you find that the refined reflection now have non-zero  $z^*$  values. Check the cross-correlation maps and if there is a distinct peak in the center. In that file you will also find a line like this "ORIGIN REFINEMENT DONE BETWEEN 48 OF THE NEW REFLECTIONS", make sure the number of listed comparisons are above 20. Also have a look at the phase residual.
7. To visually inspect the refinement result run the *Generate Image Maps* script and scroll through the phase shifted projection maps.
8. Select the images for which the phase origin could be determined and the images which were used for the 2D reference. If you saved the selection file, just use that file to load those selection again.
9. Merge the selected images all to one common 3D reference through *Merge Selected Images*, but with a *Resolution of the merged dataset for the reference* set to 15 Å.

10. Now refine the alignment of the tilted images through the *Refine PhaseOrigins Once* script, but with a smaller *Stepsize of the phase origin search* while reducing *zstarwin*.
11. After the refinement have a look at the phase residual (*MergePhaseRes*) for every refined image. If the phase residual is high it could have its origin in the CTF correction with a false defocus. This can be detected by verifying the merged data set. Because the data from most images will be complete up to 15 Å one can profit from the redundancy in the number of reflections to identify errors in the CTF correction. The defocus in the images, where the errors were detected has to be adjusted before proceeding with the refinement.
12. If the cause of the bad phase residual was not the CTF correction, check if refining the tilt geometry can improve the alignment. This would be done in another *Merge & Refine (Iterative)* run while having the parameter *Refine Tilt Geometry (Only in 3D mode)* set to Yes. This should only be done if a thorough reference already exist. You will discover that if the reference is not sufficient most commonly the tilt refinement will cause the tilt angle of all the refined images to increase.
13. If you find that your image data has a resolution of 5 Å or higher you should refine the beam tilt at this stage. This can be achieved by setting enabling the option *Refine Beam Tilt* in the *Merging Refinement* section for the refinement.
14. Now you can include images of higher tilt and align them to your 3D reference with the *Refine PhaseOrigins Once* script. The procedure of the refinement is done as before, so you should redo all the steps from item 7 to here. Remember that you can now decrease the parameter *zstarwin*.
15. The benefit of data stemming from sample tilted more than 25° is that the tilt geometry is more trustworthy, since it was determined by lattice distortions through emtilt. This means when you have merged a fair amount of higher tilt data you can in turn refine the tilt geometry of the lower tilt data.
16. Once all image data has been included and all refinements have been done you can decide on the final resolution cut-off for the individual images (Upper Resolution Limit) and run *Final Merge*.
17. An important file that appears in the Images panel during merging is *PS: TLTPLOT* file. This shows you the completeness of your 3D data set and allows you to determine at what tilt angle more data is needed for an even sampling of the 3D Fourier space.
18. The final step in the 3D reconstruction of your structure is the calculation of the 3D map. This can be achieved with the *Generate Merged Map* script. This creates a volume in CCP4 format named *MAP: Final 3D Volume*. You can examine the result in Chimera by double-clicking it.
19. Once a reliable 3D-model has been generated on can re-process the already used images with the *(Re-)Process all images* script. Thereby one can improve the data

from the individual image data by unbending with a reference created through back projection from the model. We call this procedure synthetic unbending.



## 5. Refining Merged Map

The quality of maps generated using 2D electron crystallography is limited owing to various factors including: (i) the problem of missing cone which occurs due to the fact that micrographs can be recorded until a certain tilt angle typically up to 60 degrees; (ii) the images of tilted samples have much lower quality than those of non-tilted ones; (iii) the current methods such as Lattice line fitting used to generate 3D data lack reliability. An iterative method applying known constraints in real and reciprocal space is devised in 2DX. These constraints exploit the fact that the densities mimic protein structures.

In 2DX we can refine the maps using the script “*Refine Merged Map (3D only)*”. There are several parameters in the section “3D-Volume Refinement” which should be properly set to get meaningful refined structure.

*Expected membrane height*: If the protein you work on is a membrane protein, then one can specify the height (in Å). The densities will then be restricted to this height.

*Low pass filter resolution for mask*: In each iteration, the current density is first low-passed to create a blurry version of the current map. This is used as a mask to filter out noise and signal.

*Higher and lower density thresholds*: These thresholds suggest the density limits assumed for the noise. Everything below the lower threshold will be deleted. The densities which lie between higher and lower threshold are linearly scaled down.

*Factor to scale the optimum energy of recovered cone*: Due to masking and noise removal the amplitudes of the remaining reflections can go higher. This parameter can be used to scale the energy in each iteration.

*Amplitude cutoff*: The amplitude value below which all the reflections would be ignored. This can be changed to speed up the refinement.

After you run the script, there would be many files generated. Look at the masks and final volumes. Analyze if because of the masking the volumes being cut are reasonable. The aim should be to remove as much noise as possible keeping as much signal as we can. Try different parameters if you think the masking is not reasonable.







## 6. Porting Project to FREALIGN

