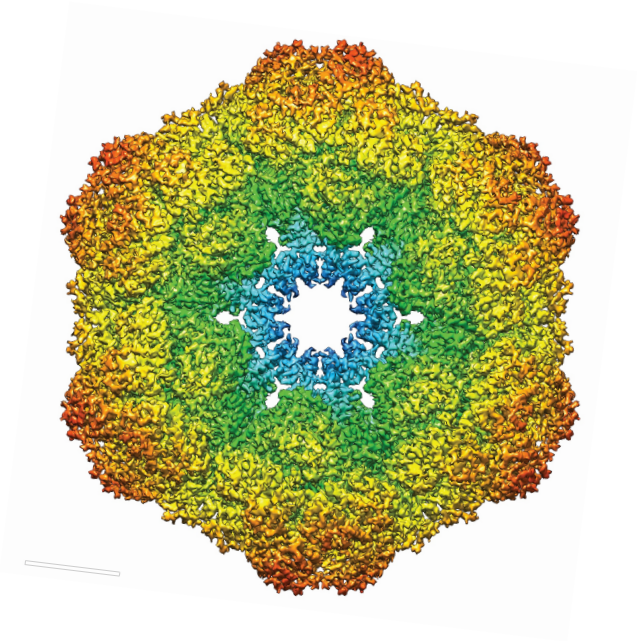
THE 7TH BRAZIL SCHOOL FOR SINGLE PARTICLE CRYO-EM

HANDS-ON



Version 8-Sep-2016

www.single-particles.org/school

1. Introduction to IMAGIC

This chapter is on how to work with the **IMAGIC** software.

An **IMAGIC** image file actually consists of two files: the header file (".hed") and the image file (".img"):

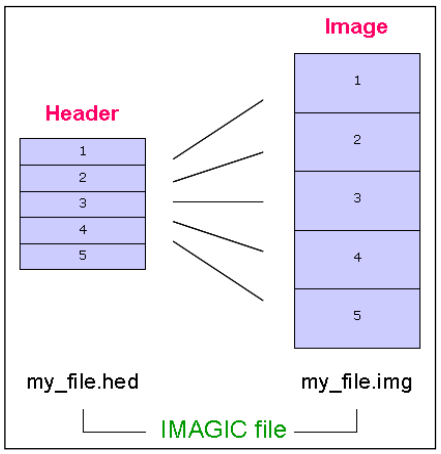


Fig. 1: IMAGIC file

The image file contains the actual image density values whereas the header file contains information about the images ("meta data") as a set of records that can be accessed through different labels. For example:

IMN	image location number (1,2,3,...)
IXLP	number of lines per image
IYLP	number of pixels per line
IZLP	number of sections if input is a 3-D volume
REF	multi-reference number
CLASSNO	class number
ALPHA	Euler alpha angle
BETA	Euler beta angle
GAMMA	Euler gamma angle
Etc…	

An additional PLT text file can be associated to an **IMAGIC** file to store further meta-data like:

coordinates of particles
contour of masks
image numbers
Euler angles
graphics (curves)
Etc…

The PLT file can contain a maximum of five numerical values per line, separated by blanks or by a comma.

A few other **IMAGIC** text (ASCII) files can be generated during processing:

a CLS file is a classification file containing classes and their members

a LIS file contains information printed during execution of a program

a LOG file output of programs when running as batch job (script)

a DAT file containing data for various purposes

DFF (deFault Files) are used to store your last answers

The **IMAGIC** coordinate system is a right-handed system with its (1,1) origin in the top-left corner of the image. The length of the lines (number of rows/columns) is **NY** and the number of lines is **NX**:

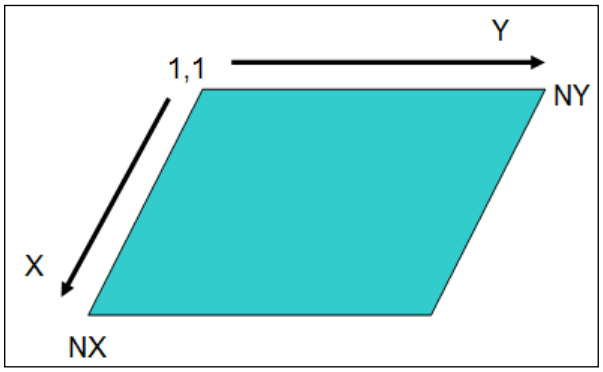


Fig. 2: IMAGIC 2-D coordinate system

The **IMAGIC** coordinates for a 3-D volume are the following:

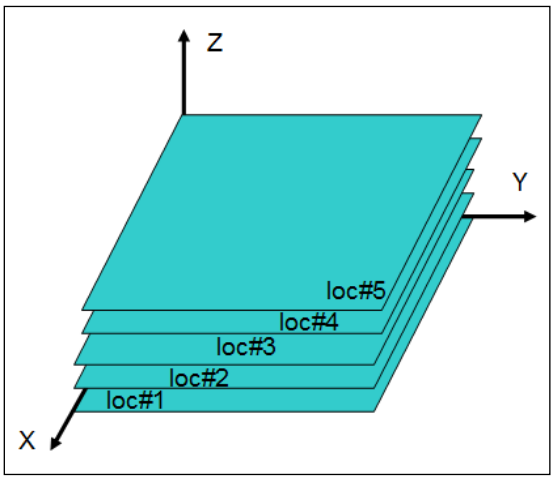


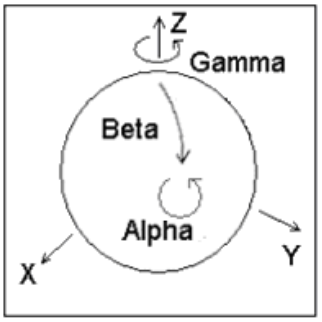
Fig. 3: IMAGIC 3-D coordinate system

Note that **Z** = **X** x **Y** as required for a right-handed co-ordinate system.

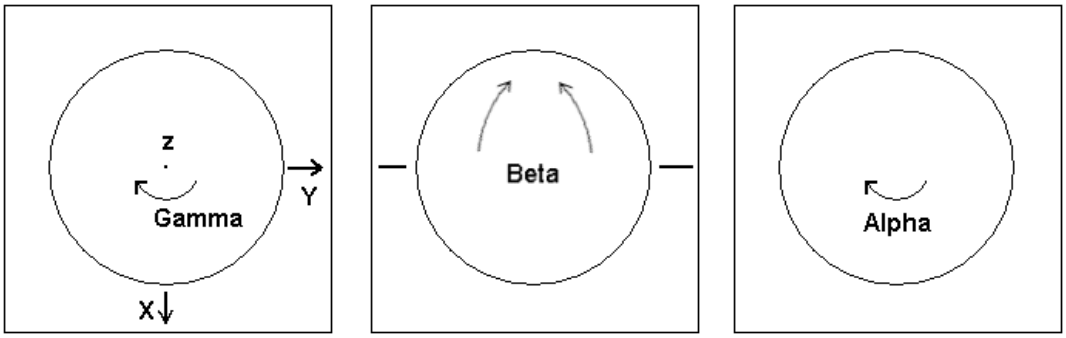
In **IMAGIC** 3-D orientations are defined by three Euler angles Alpha, Beta and Gamma.

From the perspective of an external viewer (like every IMAGIC image used/ created in commands ANGULAR-RECONSTITUTION, THREED-SURFACE, THREED-FORWARD, etc.) the Euler angles are defined as follows:

The first rotation is a rotation around the Z-axis by GAMMA, followed by a rotation BETA around the new Y-axis and a rotation ALPHA around the new Z-axis.



But normally a user does not think in this way but tries to imagine how the particle would look like "in his hands":



a) Look at the particle along the Z-axis ("north pole")

b) Rotate the particle clockwise by Gamma

c) Rotate the particle into the plane clockwise by Beta

d) Rotate the particle clockwise by Alpha

PLEASE NOTE:

The important angles to define a 3-D orientation are Beta and Gamma. Alpha is only the final in-plane rotation.

IMAGIC is started in one (or more) command window(s). The commands are interactive and are followed by specific questions. Every question also has an associated help, which can be accessed by typing "?"

IMAGIC command questions will often have a default value which appears in brackets [default]. You can use the default value by just hitting ENTER/CR.

IMAGIC remembers the last values you have entered for a specific command. These values (store in the DFF files) become the default values the next time command is used in that working directory. In general, if you do not know how to answer a question, the default values serve as an intelligent first guess.

MPI refers to parallel processing. If you notebook has multiple cores commands which are using parallel processing will ask you if you what to run the command in parallel or not. When using many images you answer will be YES. Note that the number of processors to be used should be at least the number of nodes PLUS 1:

```
Use MPI parallelisation [YES]                    : yes
Number of processors to be used                  : 3
```

Throughout this hands-on, words that appear in GREEN refer to **IMAGIC** commands. Words in red are required/suggested input values. Suggested file names are in blue.

File names are only suggested. You are free to choose whatever names you wish. However, bear in mind you will have to remember what you've chosen for the next commands.

YOUR NOTES:

1.1. Commands CREATE-IMAGE and DISPLAY

1. To start, open a command window and run **IMAGIC** by typing i or imagic.

```
my_notebook> imagic

IMAGIC-COMMAND:
```

2. Use CREATE-IMAGE to create a (test) image. First use the default options, i.e., just hit the ENTER button.

```
IMAGIC-COMMAND: create-image

** TESTIM welcomes you **

Output filename, image loc#s            : my_image
Image dimensions X,Y                    : 256,256
IMAGIC data formats you can choose      : real
Currently you can choose                : blobs      you choose
```

3. Use a separate command window to DISPLAY the image on the screen:

```
my_notebook> imagic

IMAGIC-COMMAND: display

Input image file, image loc#s           :
```

4. The first question that will appear on the screen concerns the choice of the file you wish to display. Get the test image (image), which you just created. If you have forgotten the names of the images type:

```
Input image file, image loc#s        : $dir       MS Windows
```

or

```
Input image file, image loc#s        : $ls            Linux
```

which is just an operating system call to get a list of the files you own, and look for files with the extension ".img".

Now specify the name of the image file you want to display:

```
Input image file, image loc#s      : my_image
```

DISPLAY first shows the current settings:

```
Current DISPLAY settings:

Input image FILE name              : my_image
LOCATION numbers                   : 1,1
Output DEVICE                      : XWINDOWS
DEVICE window size                 : 800,1024
SCALE factor                       : 1.0
MINX, MAXX                         : 1,256
MINY, MAXY                         : 1,256
GREYVALUES                         : 2D local survey
ERASE screen before display        : no
STARTING point (top left)          : 1,1
Display of NAME & information      : file name and location
Video lookup table (VLT)           : linear black/white
...
Parameters to be changed:
NO_CHANGES(=DISPLAY), SETTINGS, OPTIONS [NO]:
```

5. Hit the ENTER key, which means that the default NO CHANGES is used and the image will be displayed.

6. If you want to change certain settings go for the words written in capitals. For example, to change the scaling factor:

```
Parameters to be changed:
NO_CHANGES(=DISPLAY), SETTINGS, OPTIONS [NO] : scale

   Image size is: 128 x 128

Give scale factor for display               : 2
```

7. Hit ENTER to apply the changes and to go back to the DISPLAY parameter settings.

8. Then use option GREYVALUES to give different grey levels. Start with option INTERACTIVE and black, white levels -10,10 and display. Use other black, white levels and display to see how brightness and contrast of the displayed image changes. After, use option SURVEY, 2D_LOCAL.

NOTE:

If you are displaying a gallery of images (aligned images, class averages, 3-D sections etc. you should always use the GREYVALUES options SURVEY, GLOBAL.

9. Play with CREATE-IMAGE again. Create REAL images, but of different sizes, and of different options.

10. DISPLAY the images. After this, create a 256,256 image of a SIEMENS star.

11. Use the COARSE-IMAGE command with factors of 6 to give coarsened images. DISPLAY them and see the effects of sampling size on your resolution. What is the size of your image now? How much detail can you see?

12. Use the BLOW-UP-IMAGE command (option BLOWUP) to blow up the coarsened images back to the original size. DISPLAY the results and compare them with the original and the coarsened images. Compare the output images of COARSE-IMAGE and BLOW-UP-IMAGE.

TIP:

You can open multiple DISPLAY windows from different command windows.

13. Use CREATE-IMAGE to create a new test image using the option BLOBS. Use MOVE-IMAGE to ROTATE, SHIFT and COMBINE (ROT&SHIFT) the image. DISPLAY the results. Remember the IMAGIC coordinate system (chapter 1).

YOUR NOTES:

1.2. Noise

You will create a sequence of test images and add noise to them.

1. Use CREATE-IMAGE to create 256 images of CHECKERS. Make the images REAL and of size 128,128. Note: To create a file with multiple images you specify the start and final location numbers, like my_image,1,256 where my_image is the file name, 1 is the start location and 256 is the final location.

```
IMAGIC-COMMAND: create-image

Output filename, image loc#s            : my_image,1,256
Image dimensions X,Y                    : 128,128
IMAGIC data formats you can choose      : real
Currently, you can choose               : checkers
Checker size                            : 16
```

2. Use ADD-NOISE to add noise to the images:

```
IMAGIC-COMMAND: add-noise

Option used                             : ADD_NOISE
Input filename, image loc#s             : my_image
Output filename, image loc#s            : my_image_noise
Mode of operation                       : noise
Mean, sigma of Gaussian noise           : 0,20
Random number seed                      : 0
```

3. DISPLAY the results.

1.3. Noise Reduction by Image Averaging

You will get some impression what image averaging means and why image processing can enhance the resolution of noisy images.

1. Use SUM-IMAGES (option SOME_SUM) to make sums of 2, 8, 64, and 256 of the images from the file you added noise to. Input file is my_image_noise. Suggested output file names are: my_sum_2, my_sum_8... and my_sum_256. In "Location number(s) wanted:" you should specify 1-2, or 1-8, or 1-64, or 1-256 accordingly:

```
IMAGIC-COMMAND: sum-images

Mode of summing                          : some_sum
Input file, NO loc#s                     : my_image_noise
Output file, ONE loc#                    : my_sum_2      etc.
Variance file, ONE loc#                  : none
Location number(s) wanted                : 1-2           etc.
Numbers wanted                           : all
```

2. DISPLAY each result and see the effects of image averaging on the signal to noise ratio.

YOUR NOTES:

1.4. MODE Commands

MODE commands allow the creation of batch/script files with a collection of commands that can be run from the command window.

1. Use command MODE-ACCUMULATE:

```
IMAGIC-COMMAND: mode-acc

IMAGIC-COMMAND (ACC.) :
```

2. Now use commands CREATE-IMAGE and SURVEY-DENSITIES. The input file for SURVEY-DENSITIES is the output file of CREATE-IMAGE.

   Remember: You can find out about a command using HELP, and about a question using ? .

   Stop accumulating commands with MODE-STOP. You can run the file with the accumulated commands in the command, e.g. bigjob.b (Linux) or bigjob.bat (MS Windows), respectively:

```
IMAGIC-COMMAND (ACC) : mode-stop

Filename for batch/script file [bigjob] : bigjob

   Command (batch/script) file bigjob.b

   is available now

   To run the job on the monitor please use

   ...
```

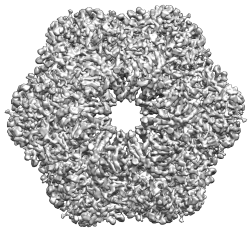
3. Use command MODE-ACCUMULATE again and accumulate command TEST-IMAGE and some other commands like ARITHM-WITH-IMAGE, SURVEY...

4. Stop accumulating commands and run the script file with MODE-SEND.

5. Use MODE-PROTOCOL to create a protocol file.

   Again use command CREATE-IMAGE with some other commands.

   Stop the protocol mode with command MODE-STOP. Edit the protocol file using a text editor of your choice.



2. The Fourier Transform

This is an exercise that will provide some basic insight into the Fourier transform (FT). Fourier transforms will be covered in the lectures; the aim of this exercise is to familiarize you with the principles of the Fourier transform and the associated **IMAGIC** commands.

Start with one-dimensional (1D) Fourier transforms and later play around with 2-D images.

2.1. Test Curves

1. Open a command window. Create a curve (a 1D image) using command CREATE-CURVE. Create an IMAGE with amplitude 1 and wavelength 0.1:

```
IMAGIC-COMMAND: create-curve

Mode of output                          : image
Output file, curve loc#s                : my_curve
Length of curve                         : 512
Curve option                            : sine
Amplitude of signal                     : 1
Wavelength of periodic signal           : 0.1
```

2. Create additional curves into the same file, sequential locations:

```
IMAGIC-COMMAND: create-curve

Mode of output                          : image
Output file, curve loc#s                : my_curve,2
Length of curve                         : 512        as before
Curve option                            : sinc       you choose
Amplitude of signal                     :            you choose
Wavelength of periodic signal           :            you choose
```

3. Open a second command window and use the command PLOT to display the curves. Like the command DISPLAY, the PLOT command first displays the current settings, which you can change by typing the names in capitals. Giving ENTER means "NO CHANGES" and the curve is displayed. You want to compare all curves so it is a good idea to fix the vertical scaling of the plot according to the chosen amplitudes with the option VERTICAL:

```
Change settings (MULT,HOR,VER,SURVEY...) [NO]: vert
Minimum, maximum for vertical scaling        : -10,10
```

Plot the curves.

4. If you want to display all curves at the same time use option MULTIPLE:

```
Change settings (MULT,HOR,VER,SURVEY...) [NO]: mult
Number of curves per plot                    :     you choose
```

2.2. Fourier Transform

1. Now calculate the Fourier transforms of the curves (curve) with the command CURVE-FORWARD-FT. The suggested output file name is my_curve_ft.

```
IMAGIC-COMMAND: curve-forw

Input file, curve loc#s                  : my_curve
Output file, curve loc#s                 : my_curve_ft
Options you can choose                   : FORWARD_FT
```

2. PLOT the Fourier transforms (May be, you want to set MULTIPLE back to 1. You can also use option VERTICAL (values like -1000,1000) to use the same vertical scaling for all curves.

2.3. Curves and their Fourier Transforms

1. There is a interactive command to create images/curves and to display the related Fourier transforms: PLAY-WITH-FOURIER-TRANSFORMS.

```
IMAGIC-COMMAND: play-with-fourier

Play with                                  : 1d_image
Mode of input                              : create
Choose curve                               : triangle
Length of curve                            : 512
Amplitude of the curve                     : 1
Width of signal                            : 0.25
```

Both, the curve and the related Fourier transform will be displayed.

You can create the next curve by typing NEXT_IMAGE:

```
How to continue                            : next
...
```

or leave the command by giving STOP_PLAYING.

```
How to continue                            : stop
```

NOTE:

In command PLAY-WITH-FOURIER-TRANSFORMS you can also use a curve/image from an input file. Use Mode of input : FILE

2. Now create various sine curves using different amplitudes 2, 4, 6, and 8. Always use wavelength 0.1:

```
IMAGIC-COMMAND: play-with-f

Play with                                  : 1d_image
Mode of input                              : create
Choose curve                               : sine
Length of curve                            : 512
Amplitude of the curve                     :              you choose
Wavelength of periodic signal              : 0.1
```

Look at the vertical scaling of the plots and notice how the amplitudes are related to the height of the peaks in the Fourier transforms.

3. Continue using SINE curves now with fixed amplitude but changing the periodicity. Use amplitude 1 and wavelength of periodic signal 0.1, 0.25 etc.

```
IMAGIC-COMMAND: play-with-f

Play with                                  : 1d_image
Mode of input                              : create
Choose curve                               : sine
Length of curve                            : 512
Amplitude of the curve                     : 1
Wavelength of periodic signal              :              you choose
```

Notice how the periodicity changes the Fourier transforms.

NOTE:

The sine (or cosine) curve in the image space corresponds to a peak in Fourier space. The amplitude and wavelength of the sine (or cosine) curve are "related" to the height and position in Fourier space.

2.4. Relationship between Image Space and Fourier Space

1. Create a new curve file (my_curve) with a number of SINE waves with various amplitudes and wavelengths. Create these into the same file, sequential locations (my_curve,1 , my_curve,2 , … my_curve,20):

```
IMAGIC-COMMAND: create-curve

Mode of output: image
Output file, curve loc#s                : my_curve
Length of curve                         : 512
Curve option                            : sine
Amplitude of the curve                  :          you choose
Wave length of periodic signal          :          you choose

IMAGIC-COMMAND: create-curve

Mode of output: image
Output file, curve loc#s                : my_curve,2
Length of curve                         : 512
Curve option                            : sine
Amplitude of the curve                  :          you choose
Wave length of periodic signal          :          you choose

IMAGIC-COMMAND: create-curve

...

IMAGIC-COMMAND: create-curve

Mode of output: image
Output file, curve loc#s                : my_curve,20
Length of curve                         : 512
Curve option                            : sine
Amplitude of the curve                  :          you choose
Wave length of periodic signal          :          you choose
```

2. Calculate the Fourier transforms of the new curves with CURVE-FORWARD-FT and store them in file (my_curve_ft).

```
IMAGIC-COMMAND: curve-forw

Input file, curve loc#s                    : my_curve
Output file, curve loc#s                   : my_curve_ft
Option used for current command            : FORWARD_FT
```

3. As before PLOT both, the curves (my_curve) and the related Fourier transforms (my_curve_ft).

4. Now, sum all curves (curve) with command SUM-CURVE:

```
IMAGIC-COMMAND: sum-curve

Choose summing option                      : total_sum
Input file, NO loc#s                       : my_curve
Output file, ONE loc#                      : my_curve_sum
Variance file, ONE loc#                    : none
```

5. PLOT the result (my_curve_sum).

NOTE:

a) Summing a huge number of sine (and cosine) curves with different amplitudes and wavelengths creates a non-periodic curve.

b) And even more: one can say that any (real) curve can be constructed by a combination of sine and cosine waves of different wavelengths and amplitudes.

6. Calculate the Fourier transform (my_curve_sum_ft) of the new curve (my_curve_sum) using command CURVE-FORWARD-FT. PLOT the Fourier transform (my_curve_sum_ft).

NOTE:

The sum of sine (or cosine) curves in the image space relates to the sum of the sine (or cosine) peaks in Fourier space.

7. Now calculate a reverse Fourier transform with command CURVE-REVERSE-FT. Input is the Fourier transformed curve (my_curve_sum_ft).

8. PLOT the reverse Fourier transform. Note that it is the same as the original curve (curve) before the Fourier transformation.

NOTE:

The information in a curve/image ("Image Space" or "Real space") and in the Fourier transform ("Fourier Space") is equivalent. The curves/images in both spaces contain all curve/image information.

This means that in image processing it is possible to go from one space to the other without losing any image information!

NOTE:

Mode of Fourier transforms: FORWARD means going from a curve/image ("Image Space" or "Real space") to its Fourier transform ("Fourier Space"). The transformation going from the Fourier transform to the curve/image is called REVERSE (sometimes also called "inverse").

YOUR NOTES:

2.5. Fourier Space and Filtering

1. Let us once again create sine curves: one with a long wavelength and another with a short wavelength (0.5 and 0.002) using CREATE-CURVE. Remember to save these two curves to the same file using the same file name and different location numbers, like in my_curve and my_curve,2 .

2. Calculate the Fourier transforms with CURVE-FORWARD-FT and compare the results using command PLOT (use option MULTIPLE).

NOTE:

(a) The first sine curve shows "large" details, which in Fourier space are represented by densities close to the centre of the Fourier transform.

(b) The second sine curve shows "small" details, which in Fourier space are located far away from the centre.

(c) Usually the very "large" details (density ramps, for example) and the very "small" details (mostly noise) are hiding the motif, which you are interested in.

(d) As seen in your two test curves Fourier space offers a nice possibility to remove this unwanted information: filter the Fourier transform close to the centre ("low frequencies") and at the borders ("high frequencies").

3. Create a new curve (my_curve) with CREATE-CURVE, with BLOCKWAVE for curve option, 500 for the amplitude and 0.25 for the wave length

4. Next create a ramp in location #2 (my_curve,2). Run CREATE-CURVE with option RAMP, and use 1 for the amplitude and 3 for the inclination.

5. Now add the block-wave and the ramp curves with SUM-CURVE using summing option TOTAL_SUM. Do not calculate a standard deviation (give none). Use the output name my_curve_ramp.

6. PLOT the sum (my_curve_ramp). Notice how the signal (block-wave) is disturbed by the ramp.

7. Remove these "large" unwanted details (low frequencies) with a high-pass filter in Fourier space using command CURVE-FILTER and option HIGH_PASS and low-frequency cut-off 0.01.

```
IMAGIC-COMMAND: curve-filt

Input file, curve loc#s                    : my_curve_ramp
Output file, curve loc#                    : my_curve_ramp_hp
Filter option                              : high_pass
Low frequency cut-off                      : 0.01
Remaining LF transmission                  : 0,0
```

8. PLOT the high-pass filtered curve (my_curve_ramp_hp).

NOTE:

Large details can be suppressed by high-pass filtering in Fourier space. But, be aware that the low-frequency component of the original curve can also be affected.

Errors/artefacts can occur at the edges.

9. Next create Gaussian noise in location #3 (my_curve,3). Run CREATE-CURVE with option NOISE, and use 0, 40 for the MEAN and SIGMA.

10. Add the noise to the block-wave signal with CURVE-SUM.

```
IMAGIC-COMMAND: sum-curve

Choose summing option                      : some_sum
Input file, NO loc#s                       : my_curve
Output file, curve loc#                    : my_curve_noise
Output standard deviation file             : none
Location number(s) wanted                  : 1;3
```

11. PLOT the sum (my_curve_noise). Notice that the signal is disturbed by noise.

12. Remove these "small" unwanted details (high frequencies) by low-pass filtering in Fourier space with command CURVE-FILT using option LOW_PASS and high-frequency cut-off of 0.1.

```
IMAGIC-COMMAND: curve-filt

Input file, curve loc#s                : cmy_curve_noise
Output file, curve loc#                : my_curve_noise_lp
Filter option                          : low_pass
High frequency cut-off                 : 0.1
```

13. PLOT the low-pass filtered curve (my_curve_noise_lp).

> NOTE:
>
> Noise (small details / high frequencies) can be removed by a (Fourier space) low-pass filter. But, of course, also fine details of the original curve are affected.

14. Finally, create a curve with disturbing low (ramp) and high frequencies (noise). Add all three curves (my_curve) with CURVE-SUM, using the option TOTAL_SUM (to get my_curve_ramp_noise).

15. PLOT the curve (curve_ramp_noise) to see how the curve is disturbed by the ramp and by Gaussian noise.

16. To remove both unwanted information call CURVE-FILTER again now using a BAND_PASS filter, which is a combination of a high-pass and a low-pass filter:

```
IMAGIC-COMMAND: curve-filt

Input file, curve loc#s              : my_curve_ramp_noise
Output file, curve loc#              : my_curve_ramp_noise_bp
Option to choose                     : band
Low frequency cut-off                : 0.01
Remaining LF transmission            : 0
High frequency cut-off               : 0.1
```

17. As usual PLOT the filtered curve (my_curve_ramp_noise_bp) and see how a band-pass can remove unwanted information.

TIP:

During image processing, it is a good idea to have your own naming convention, so that in a list of files you can easily understand what each file is from its name. For example in this case "sine" is your input file containing a sine wave and "sine_mask" is your output file with the masked sine wave.

YOUR NOTES:

2.6. 2-D Images and Fourier Transforms - First Steps

We now want to play with 2D images and their Fourier transforms.

1. Start by using PLAY-WITH-FOURIER-TRANSFORMS using 2D images (option 2D_IMAGE). First, create a SINE-wave image (my_sine). When asked for a PERIODICITY choose 0.1:

```
IMAGIC-COMMAND: play-with-f

Play with                                  : 2d_image
Mode of input                              : create
Choose curve                               : my_sine
Image dimensions X,Y                       : 512,512
Wavelength of periodic signal              : 0.1
Direction of wave                          : horizontal
Mask radius, drop-off (0: no mask)         : 0          no mask
Grey values to scale display (0: auto)     : 0          automatic
```

PLAY-WITH-FOURIER-TRANSFORMS will display the create images in the first display window and the related Fourier transform in the second display window.

2. You should now simply see two "points" in the Fourier transform of the input image. This is the Fourier space representation of a sine wave, with the periodicity you have specified (0.1 is suggested above). When "in" Fourier space, information about higher frequencies (i.e. when the wave length is small) is given further towards the edge of the Fourier transform image, whilst information about the lower frequencies is given towards the centre.

3. As you are examining a 2-D image, the sine waves also have a direction. In this case the sine wave travels horizontally. In the Fourier transform, if you were to join the points with a line it would also go horizontally.

4. To demonstrate this effect continue with NEXT_IMAGE and create the same image again but now using option VERTICAL. Note that how the direction has changed in both, the image and the related Fourier transform.

5. Finally continue with NEXT_IMAGE now using the direction option ANGLE. Use 45 (degrees). As before you should find that the direction of the frequency space points has rotated by 45 degrees.

```
IMAGIC-COMMAND: play-with-four

Play with                                  : 2d_image
Mode of input                              : create
Choose curve                               : sine
Image dimensions X,Y                       : 512,512
Wavelength of periodic signal              : 0.1
Direction of wave                          : angle
Rotatation angle                           : 45
Mask radius, drop-off (0: no mask)         : 0          no mask
Grey values to scale display (0: auto)     : 0          automatic
```

6. However when using angles, which are not multiples of 45 (let's say: 30) there will no longer be just points. This is because the rotated sine waves are no longer continuous, and also have interpolation effects from the rotation. In other words the images are no longer pure sine waves

7. Use command PLAY-WITH-FOURIER-TRANSFORMS to play around with other images. Learn how the related Fourier transforms look like.

8. You can use command CREATE-IMAGE to create 2-D image files (my_image), which you can sum with command SUM-IMAGES (my_image_sum). Use command PLAY-WITH-FOURIER-TRANSFORMS to visualise images and Fourier transforms.

NOTE:

Like in the 1D case for curves, any (real) 2-D image can be seen as a combination of sine and cosine waves of different frequencies, and in different directions. A Fourier transform decomposes a real image into its constituent sine / cosine waves. Only sine waves that fit perfectly on the sampling grid, have perfect diffraction peaks in Fourier space.

2.7. 2-D Images and Fourier Transforms - Masks

1. The motif in your images is normally close to the centre of the frame and you are normally not interested in the information near the frame edge. To minimize the influence of background you may want to mask out the edges.

2. The effect of masking can also be visualised using PLAY-WITH-FOURIER-TRANSFORMS. Create a SINE image rotated by 30 degrees. First use no mask. To better visualise the result adapt the display grey-value scale:

```
IMAGIC-COMMAND: play-with-f

Play with                                  : 2d_image
Mode of input                              : create
Choose curve                               : sine
Image dimensions X,Y                       : 512,512
Wavelength of periodic signal              : 0.1
Direction of wave                          : angle
Rotatation angle                           : 30
Mask radius, drop-off (0: no mask)         : 0          no mask
Grey values to scale display (0: auto)     : 1,8
```

3. Subsequently mask out the centre of the image with a soft drop off circular mask. You should see the peaks more clearly now.

```
IMAGIC-COMMAND: play-with-f

Play with                                  : 2d_image
Mode of input                              : create
Choose curve                               : sine
Image dimensions X,Y                       : 512,512
Wavelength of periodic signal              : 0.1
Direction of wave                          : angle
Rotatation angle                           : 30
Mask radius, drop-off (0: no mask)         : 0.7,0.1    soft mask
Grey values to scale display (0: auto)     : 1,8        as before
```

NOTE:

You may have noticed that after applying the soft-circle, as well as the points becoming clearer, they also become larger. This demonstrates a very important image space / Fourier space relationship. A multiplication in image space (the application of a soft-circle is effectively a multiplication) leads to a convolution in Fourier space. This relationship occurs in both directions i.e. if you were to multiply the Fourier space image by a circular mask, you would get a convolution in image space (this is what filtering is), and similarly if you were to convolute in one space, you will get a multiplication in the other.

2.8. 2-D Images and Fourier Filters

1. As was done for the 1-D curves you can also use filters in Fourier space to remove unwanted information in 2-D images.

2. Create a new test-image showing a RECTANGLE and add some noise to it:

```
IMAGIC-COMMAND: create-im

Putput filename                    : my_rectangle
Image dimension                    : 256,256
...

IMAGIC-COMMAND: add-noise

Mode of operation                  : ADD_NOISE
Input file                         : my_ectangle
Output file                        : rmy_rectangle_noise
Mwan, sigma of Gaussian noise      : 0,5
Random number seed                 : 0              your choice
```

3. Apply low-pass filters to the images (my_rectangle_noise) with the command LOW-PASS-FILTER:

```
IMAGIC-COMMAND: low-pass

Mode of operation                    : LOWPASS
Input file                           : my_rectangle_noise
Output file                          : my_rectangle_noise_lp
High frequency cut-off               : 0.2              your choice
```

Play with different values for "High frequency cut-off" and always DISPLAY both, the original image (my_rectangle_noise) and its low-pass filtered version (my_rectangle_noise_lp).

4. Also apply high-pass filters onto the images (my_rectangle_noise):

```
IMAGIC-COMMAND: high-pass

Mode of operation                    : HIGHPASS
Input file                           : my_rectangle_noise
Output file                          : my_rectangle_noise_hp
Low frequency cut-off                : 0.2              your choice
Remaining transmission               : 0
```

Play with different values for the "Low frequency cut-off" and DISPLAY both, the original image (my_rectangle_noise) and its high-pass filtered version (my_rectangle_noise_hp).

NOTE:

(a) "Large" details are represented by low frequencies.

(b) "Small" details are represented by high frequencies

(c) Usually the very "large" details (density ramps, for example) and the very "small" details (mostly noise) are hiding the motif, which you are interested in.

(d) Fourier filters offers a nice possibility to remove this unwanted information: filter the Fourier transform close to the centre ("low frequencies") and at the borders ("high frequencies"). Such a filter is called a BAND-PASS FILTER.

5. **Now apply band-pass filters to a "real" image. In the data directory Dataset/Wormhemoglobin/Particles_5 on the Brazil School network drives you can find an IMAGIC image file with five "worm hemoglobin" particles h_test. Copy h_test.hed and h_test.img to your working directory.**

6. Apply the band-pass filter with command BAND-PASS-FILTER:

```
IMAGIC-COMMAND: band-pass

Mode of operation                    : BANDPASS
Input file                           : h_test
Output file                          : h_test_bp
Low frequency cut-off                : 0.2          your choice
Remaining transmission               : 0
High frequency cut-off               : 0.8          your choice
```

7. Open a second terminal window and DISPLAY the output (h_test_bp) to check the result. Do NOT exit DISPLAY but call option WATCHDOG.

8. Now play with different values of "Low frequency cut-off" and "High-frequency cut-off). Always use the same output file (h_test_bp). DISPLAY/WATCHDOG will automatically display the new images.

   Use "extreme" band-pass parameters so that only high frequencies (0.2,0,0.9, for example) or only low frequencies (0.05,0.005,0.1, for example) are retained.

9. Next, adjust the low and high frequency cut-offs to the particle size. Remember, a band-pass filter is combination of low- and a high-pass filter:

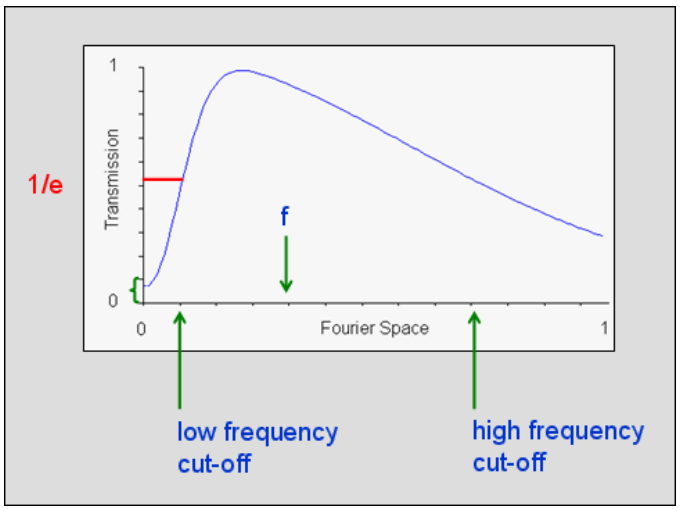


Fig. 4: Band-Pass Filter

Say that the image is scanned such that each pixel is of size

$$\text{pixel size}$$

The best resolution, which can (theoretically) be achieved for this sampling is given by the right edge of the Fourier transform image. It is the so-called *Nyquist* frequency

$$2\ x\ \text{pixel size}$$

Which corresponds to the maximum spatial frequency

$$\frac{1}{2\ x\ \text{pixel size}}$$

Remember that the centre of the transform is zero spatial frequency.

Any cut-off value asked by **IMAGIC** filtering commands is a fraction f between 0 and 1 and corresponds to a spatial frequency

$$\frac{\text{f}}{2\ x\ \text{pixel size}}$$

The low-frequency cut-off: to remove all those low frequencies, which contain information larger than the size of your particle. These could be density ramps or other low-frequency information coming from the background of the images. You can adapt the low-frequency cut-off ("large patterns") to the size of the particle

$$\frac{2\ x\ \text{pixel size}}{\text{particle size}}$$

High frequency cut-off: to remove high frequencies containing mostly noise and little signal, thus increasing the overall SNR (signal to noise ratio) of the images. Adapt the high frequency cut-off ("small patterns, noise") to the expected resolution:

$$\frac{2\ x\ \text{pixel size}}{\text{expected resolution}}$$

The size of the test particle is 200 Angstrom and the pixel size is 5.43 Angstrom. Expecting a resolution of 15 Angstrom you get:

$$\text{LF cut-off} = \frac{2\ x\ \text{pixel size}}{\text{particle size}} = \frac{2 \cdot 4.4}{200} = 0.044$$

$$\text{HF cut-off} = \frac{2\ x\ \text{pixel size}}{\text{exp. resolution}} = \frac{2 \cdot 4.4}{14} = 0.63$$

So you can use:

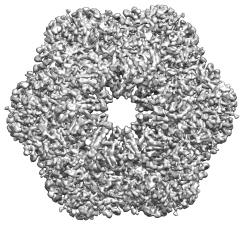
LF cut - off: 0.05
HF cut - off: 0.7

The **IMAGIC** filter "cut-off" parameters are very gradual Gaussian drop-off values and do not correspond to sharp masks in Fourier space!

NOTE:

2 x pixel size is the Nyquist frequency which is the theoretical limit to the resolution that can be achieved.

YOUR NOTES:



3. The Data Set: Worm Hemoglobin

Hemoglobin (Hb) is the iron-containing oxygen-transport metalloprotein present in the red blood cells of vertebrates. In earth worms (Lumbricus terrestris), the hemoglobin (sometime spelt as haemoglobin; also known as erythrocruorin) is extracellular, freely dissolved in the blood as a 3.6 MDa dodecameric assembly.

Point-group symmetry: Dodecameric assembly D6 (622)

Data collection:

Micrographs were collected as 7-frame movies on an FEI Titan KRIOS with a Cs corrector and a X-FEG operated at 300 kV.

Spherical aberration:	0.02 mm
Focal distance:	3.4 mm
Objective aperture:	120 micrometre
Pixel Size:	1.11 Å (coarse 2: 2.22 Å; coarse 4: 4.44 Å)
Size of a single micrograph:	4096 x 4096

YOUR NOTES:

On the Brazil School server you will find a **whgb_dataset_2016** directory with the following sub-directories:

Directory "**00_whgb_test_micrographs**" containing:

1 stack of two test micrographs in **IMAGIC** format (two from the 500 micrographs described below)

Directory "**01_whgb_micrographs_mrc**" containing:

15 micrographs aligned movie sums in **MRC** format

Directory "**02_whgb_micrographs_imagic**" containing:

70 raw micrographs (10 movies of 7 frames each) in a single IMAGIC file with 70 locations (full 4096x4096 pixels)

Directory "**03_whgb_micrographs_preprocessing**" containing:

3500 pre-processed (including camera correction; anisotropic magnification correction) and 4-times coarsened ("C4") micrographs (500 movies)

Directory "**04_whgb_micrographs_moviealigned**" containing:

500 camera corrected, anisotropic magnification corrected, aligned 4-times coarsened micrograph movie-sums

Directory "**05_whgb_micrographs_CTF_correction**" containing:

Files associated with the automatic CTF correction

Directory "**06_whgb_particle_picking**" containing:

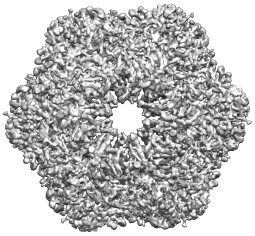
Files associated with particle picking

Directory "**07_whgb_particle_classification**" containing:

Files associated with particle picking

Directory "**08_whgb_first_3D_reconstruction**" containing:

Files associated with the first 3D reconstruction



4. Image Analysis: Short Overview

The hands-on processing of the worm hemoglobin dataset:

- Playing with the CTF
- Prepare and CTF correct the Micrographs
- Particle Picking
- Extract/box and pre-treat the Single Particle Images
- Alignment-by-Classification
- MSA-Classification
- Angular Reconstitution
- 3-D Reconstruction
- Multi-Reference Alignment
- Re-Projections and Iterative Refinements
- Fourier Shell Correlation
- Advanced topics (movie-alignment etc.)
- More…

5. Micrographs

The raw data will first be organized as a "stack" of images, which will be treated as your initial, raw data. To achieve this, the micrographs will be appended together into one single file.

Before you begin appending the micrographs you should look at the original micrographs to get an idea of how to convert your micrographs to an **IMAGIC** stack file.

1. **You will find 15 of the MRC-format images in the data directory whgb_dataset_2016/01_whgb_micrographs_mrc of the Brazil School server. The 15 MRC micrographs are actually aligned movie-sums of size 1024x1024 pixels.**

2. Copy the 15 MRC images to your own working directory. We suggest you to use the same directory names on your computer as on the server (or those suggested in this manual).

3. To be able to process the micrographs in **IMAGIC** you need to convert them into **IMAGIC** format. First you need to write the file names of all micrographs, which you want to use into a text file. Create this text file (filenames.txt) with your text editor and add the micrograph file names

   whgb_msums_c4_001.mrc
   …
   whgb_msums_c4_015.mrc

   one name per line.

4. You may later want to refine the results in **Frealign**. That program requires a micrograph identification number, which we can already write into the header of the **IMAGIC** files. To be able to do this you should create an additional text file (filenumbers.txt) with the micrograph numbers:

   001
   …
   015

   one number per line. (You can also insert this numer, and many other general parameters) at a later processing stage using one of the many options of the HEADERS-INFORMATION command).

5. Convert the micrographs with the command IMPORT-EXPORT (same command as EM2EM). Remember that the pixel size of these micrographs is 4.44 Å:

```
IMAGIC-COMMAND: import-ex

Convert 2D images or 3D volumes          : 2d
Data format of the input to be converted: mrc
MRC format                               : mrc_2000/2014
Type of input file                       : set_of_many_files
Are the input images movie frames        : no
Export to which data format              : imagic
How to get import file names             : file_of_filenames
File of input file names                 : filenames.txt
Output image file                        : micrograph_play
Pixel size (in Angstrom)                 : 4.44
Use standard em2em coordinate conversion: yes
In case of conflicts, which preference   : threshold_dens
Set some additional output header values: yes          we want to
                                                       store the
                                                       EM data
How to get the defocus values            : no_defocus
                                                       not yet
                                                       determined
How to get the EM parameters             : interactive
Microscope acceleration voltage          : 300
Focal distance of objective              : 3.4
Spherical aberration                     : 0.02
Objective aperture (micrometer)          : 120
How to get the micrograph numbers        : file_of_numbers
Text file with numbers                   : filenumbers.txt
```

NOTE:

Remember that every **IMAGIC** command provides detailed help. You can access it by typing HELP <command>. Also every command question provides help, which can be accessed by typing a ? next to it.

NOTE: You currently only imported 15 micrographs to get used to EM2EM.

6. **Now continue the practical with all 500 micrographs provided (movie sums). Copy the files whgb_c4_msums.hed and whgb_c4_msums.img that you can find on the data directory whgb_dataset_2016/04_whgb_moviealigned_images of the Brazil School server to your working directory.**

7. To check if the copy was done correctly you use the command HEADER option HOWMANY:

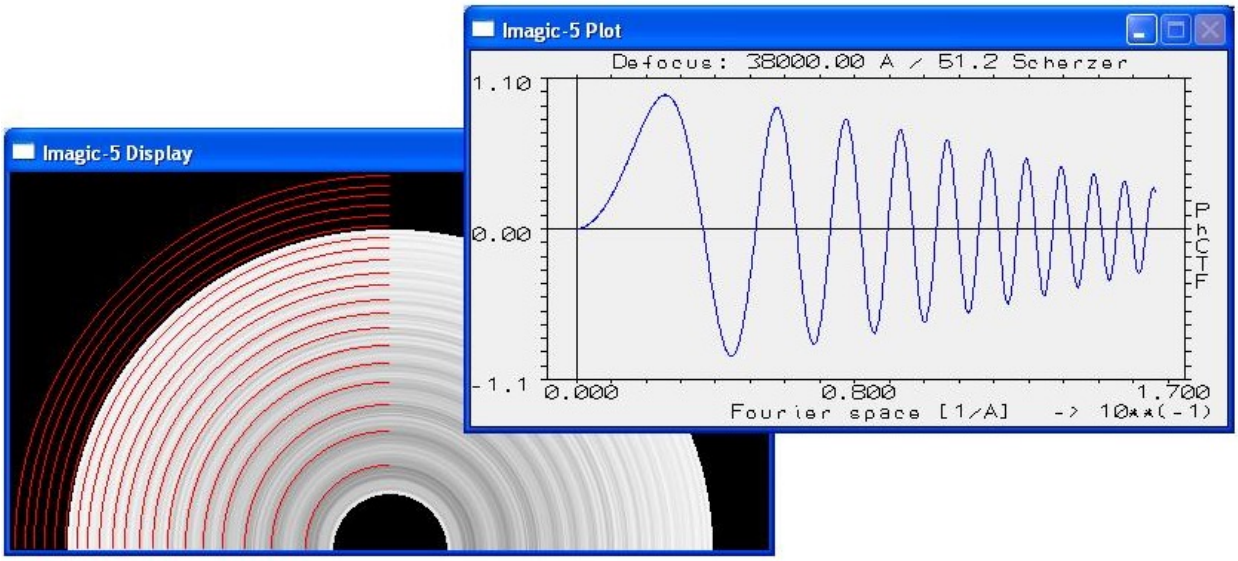
```
IMAGIC-COMMAND: header

Specify option                        : howmany
Input (header) file, image loc#s      : whgb_c4_msums
```

The command HEADER should tell you that there are 500 images, which have a size of 1024x1024.

8. Suppress extreme low frequencies by applying a band-pass filter. Use command PREPARE-IMAGE. One can also use this command to remove extreme density values in the micrographs (such as dead pixel in the CCF camera, dust on a scanned micrograph etc.). Do not use a mask:

```
IMAGIC-COMMAND: prep-im

Mode of operation                     : PREPARE_IMAGES
Input file, image loc#s               : whgb_c4_msums
Output file, image loc#s              : whgb_c4_prep
Low frequency cut off                 : 0.02
Remaining low-freq. transm.           : 0
High frequency cut off                : 0.9     "0" means no cut-off
Mask radius, drop-off                 : 0       "0" means no mask
Desired new sigma                     : 10      "0" means keep sigma
Remove (dust) outliers                : yes
Outliers off beyond which sigma       : 4.5
Invert the image densities            : yes     "yes" for vitreous-ice data
```

6. Contrast Transfer Function (CTF)



As discussed in the lectures an electron microscope unfortunately does not image all frequencies equally. This exercise is meant to play around with the command TRANSFER, which is an interactive program to calculate the (rotationally symmetric) CTF according to chosen microscope parameters.

6.1. Playing around with EM Parameters and their Influence on the CTF

1. Call command TRANSFER. Note that this command is an interactive command with many parameters like in commands PLOT and DISPLAY, which you already know. You can use the keywords written in capitals to change important parameters. TRANSFER allows you to change the various parameter settings until you type CR/ENTER, which means NO CHANGES, i.e. go ahead and display the CTF curve:

```
IMAGIC-COMMAND: transfer
```

TRANSFER displays the settings:

```
Current TRANSFER settings:
==========================

Desired TRANSFER function           : Phase CTF
Acceleration VOLTAGE                : 200          kV
Relativistic WAVE length in Angstrom: 0.025045     Angstrom
CHROMATIC aberration                : None
SPHERICAL aberration constant       : 2.2          mm
FOCAL length of objective           : 1.6          mm
APERTURE of objective lens          : 50.0         micro m
Coherent illumination SOURCE/ANGLE  : 0.0
DEFOCUS value                       : 890.7438965  Angstrom
GENERAL defocus values              : 1.2          Scherzer
OBJECT size defocus envelope        : Off
LENGTH of transfer function         : 640          pixel
PIXEL size in curve                 : 1.0          Angstrom
-------------------------------------------------------------
MODE of operation                   : Calculation of CTF
Output DESTINATION for plot(s) is   : IMAGIC plot

Change options (VOLT,DESTIN.,MODE, etc. ...) [NO] :
```

2. First play around with different pixel sizes:

```
Change options (VOLT,DESTIN.,MODE, etc. ...) [NO] : pixel

Pixel size measured in Angstrom                    : 2.22
```

3. Always give CR/ENTER to display the CTF curve.

4. Next change the DEFOCUS value to 10000 Angstrom:

```
Change options (VOLT,DESTIN.,MODE, etc. ...) [NO] : defocus

Defocus value                                      : 10000
```

5. Now give CR/ENTER to display the CTF curve.

6. Now you can play around with other defocus values (200, 500, 3000, 30000...) and notice their influence on the CTF.

NOTE:

In Scherzer focus (GENERAL defocus value = 1.0 Scherzer) you have good image contrast over a large range of frequencies but, unfortunately, you have very little image contrast in the low frequencies and, as a result, you cannot recognize your particles. When using large defocus values, lower frequencies are transferred better, improving the visibility of the particles. But, unfortunately, you now get more frequencies, which are not imaged at all (the "zeroes") and even worse, some frequencies are imaged with reversed contrast.

7. Also play around with other parameters (VOLTAGE etc.) and examine the related CTF curves.

8. For a low VOLTAGE parameter also define a CHROMATIC aberration. Examine the related CTF curves.

NOTE:

A large amount of chromatic aberration creates an envelope function, which is imposed onto the CTF so that the very high frequencies are not transferred any more. Even CTF correction cannot restore these higher frequencies.

YOUR NOTES:

6.2. Interactive CTF Correction

Before using the automatic CTF estimation/correction procedures it is a good idea to interactively try to correct the CTF for a few micrographs.

Copy the files test_micrographs that you can find in the data directory whgb_dataset_2016/00_whgb_test_micrographs of the Brazil School server to your working directory. The file contains two micrographs (coarsened by a factor of 2 - the pixel size is 2.22 Å).

Use command TRANSFER.

1. First specify the important EM parameters. In this case:

```
Change options                          : voltage
Acceleration voltage in kV              : 300
...
Change options                          : spherical
Spherical aberration in mm              : 0.02
...
Change options                          : focal
Focal distance                          : 3.4
...
Change options                          : pixel
Pixel size measured in Angstroms        : 4.44
...
```

2. To estimate the CTF call option FIND_CTF:

```
Change options                    : mode
Dimension of the data set         : 2d
Choose mode of operation          : find
Input file, image loc#s           : test_micrographs,one
                                          location number
Default filter parameters         : yes
```

TRANSFER displays: the CTF curve;

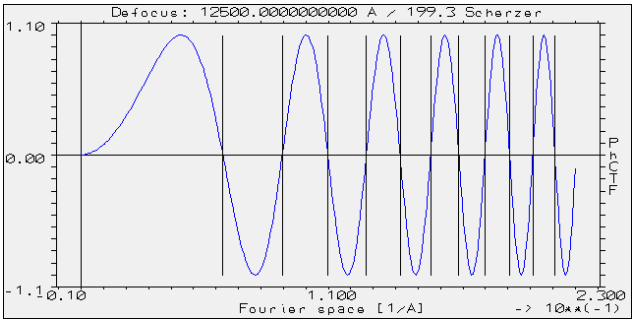


Fig. 5: CTF plot in TRANSFER

the profile of the rotational power spectrum;

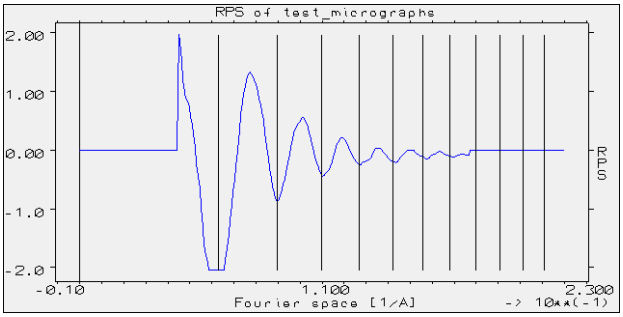


Fig. 6: RPS plot in TRANSFER

as well as the rotational power image.

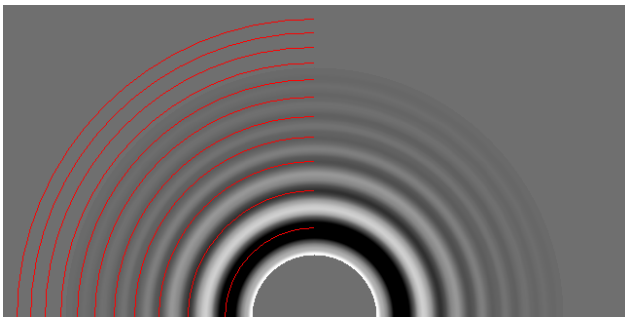
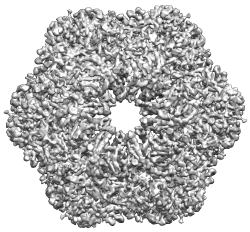


Fig. 7: RPS image in TRANSFER

In the CTF curve vertical lines mark the zeroes. The positions of these zeroes are also shown in the rotational power spectrum profile (vertical lines) and the rotational power spectrum image (red lines).

3. Play around with various defocus values until the zeroes in the CTF curve (lower curve) and the zeroes/Thon rings in the micrograph images (images above) are the same.

Defocus values found:



7. (Automatic) CTF Correction

Coming back to the stack of micrograph images (whgb_c4_prep), you will CTF correct the full stack rather than the individual micrographs.

7.1. Calculate pre-treated Amplitude Images

1. Calculate the amplitude images (whgb_c4_ampl) of the micrographs (whgb_c4_prep). Before the amplitudes are calculated the micrographs will be masked and once more band-pass filtered. These are the first filter parameters, which you are asked to specify. The amplitude images itself will also be masked and band-pass filtered (especially the background has to be removed by reducing the low frequencies). Having applied this filter the Thon rings should be better visible. The command to do all this is CREATE-PRETREATED-AMPLITUDES. Output will be the pre-treated amplitudes (whgb_c4_ampl):

```
IMAGIC-COMMAND: create-pre-ampl

Option used                          : AMP_PRETREATED
Input file, image loc#s              : whgb_c4_prep
Output file, image loc#s             : whgb_c4_ampl
  Before the calculation of the amplitudes the images
  will be band-pass filtered to remove low frequencies.
  Please specify (0,0: no filter):
Low frequency cut off                : 0.2      High: data is coarsened
Remaining low-freq. transm.          : 0
High frequency cut off               : 0.99
  The image will be masked by a soft circle.
  Please specify:
Mask radius, drop-off (0: no mask) : 0.99,0.05
Apply which arithmetic operation     : nothing
```

```
  Finally also the amplitudes will be band-pass filtered
  to better visualize the Thon rings. Please specify the
  band-pass for the amplitudes (0,0: no filter):
Low frequency cut off                 : 0.02      very small
Remaining low-freq. transm.           : 0.02      NEVER use 0
High frequency cut off                : 0.5       no high frequencies
Cut out the central part              : no
Coarsen the final amplitude images    : yes
Coarse factor                         : 2
```

2. Now it is necessary to check if the filter parameters were chosen correctly. First we average all pre-treated amplitudes (micrograph_ampl) with the command SUM-IMAGE:

```
IMAGIC-COMMAND: sum-image

Mode of summing                       : total
Input image file, No loc#s            : whgb_c4_ampl
Output image file, ONE loc#           : whgb_c4_ampl_sum
Variance file, ONE loc#               : none
```

3. DISPLAY this sum (whgb_c4_ampl_sum):

   You can (but you don't have to) adjust the DISPLAY using option GREYVALUES to better visualise the Thon rings.

   Generate a PROFILE of the central line:

```
Change options (VOLT,DESTIN.,MODE, etc. ...) : profile
Use cursor to position profile               : no
Starting point (IMAGE coordinates X,Y)       : 257,257
End point      (IMAGE coordinates X,Y)       : 257,512
...
Change options (VOLT,DESTIN.,MODE, etc. ...) : no
```

   If the CTF curve does not converge to zero, the low frequencies are not yet reduced enough and the parameter in CREATE-PRETREATED-AMPLITUDES should be enhanced:

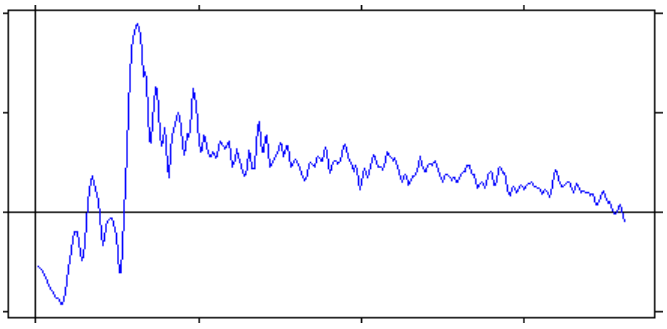


Fig. 8a: Profile in DISPLAY

If the CTF curve approaches zero for high frequencies the band-pass parameters were chosen correctly:

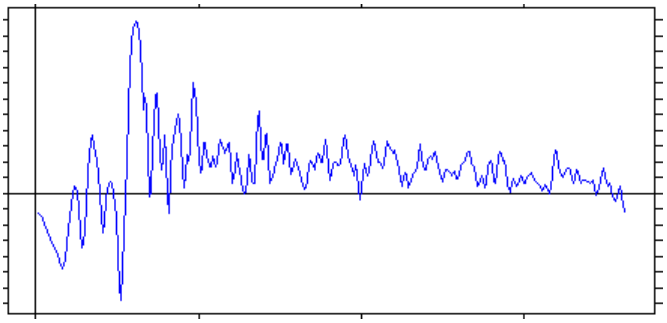


Fig. 8b: Profile in DISPLAY

NOTE:

If the checks are done, do not forget to reset your DISPLAY parameters GREYVALUE to SURVEY and 2D_IMAGE or GLOBAL.

YOUR NOTES:

7.2. Estimate CTF using MSA and Classification

Next, the pre-treated amplitude images will be treated by multivariate statistical analysis (MSA) and classification. In contrast to the individual amplitude images the class averages will show the Thon rings much better which are needed to find the defocus values.

1. Certain areas are not of interest and should not be taken into consideration for MSA and classification. Create a mask for MSA:

```
IMAGIC-COMMAND: msa-mask

Option used                        : MSAMASK
Mode of mask                       : total_sum
Input file, image loc#s            : whgb_c4_ampl
Output filename, image loc#s       : whgb_c4_ampl_msamask
Additional mask                    : ring_mask
Ring mask radius1, radius2         : 0.45,0.95
```

2. DISPLAY the MSA mask (whgb_c4_ampl_msamask) and check if the ring mask correctly masks out the unwanted inner and outer parts. If not redo the command MSA-MASK using other radii.

3. Run MSA-RUN on the amplitude images (whgb_c4_ampl):

```
IMAGIC-COMMAND: msa-run

Choose mode of operation           : fresh
MSA distances                      : modulation
Input (= output) file              : whgb_c4_ampl
Input MSA mask file                : whgb_c4_ampl_msamask
Eigenimages output file            : whgb_c4_ampl_eigen
Use default answers for            : yes
Number of iterations               : 50
Number of eigenimages              : 15
Rootname for results file          : whgb_c4_ampl_msa
```

NOTE:

MSA-RUN is a CPU intensive command when using large data sets (even when running in MPI parallel). So it can be a good idea to use commands MODE-ACCUMULATE (and later MODE-STOP) to create a batch job and run it over night or during lectures.

4. Classify the MSA treated amplitude patches (whgb_c4_ampl) with command MSA-CLASSIFY. "Active eigenimages" are the location numbers of the eigenimages showing Thon rings:

```
IMAGIC-COMMAND: msa-classify

Input to be classified                : images
Classification option                 : hac
Input (=output) header file           : whgb_c4_ampl
Percentage of images to be ignored    : 0
Active eigenimages                    : 10     the last eigenimage must
                                               still contain Thon rings
Use default classification options    : yes
What number of classes do you wish    : 50     your choice
Rootname for output files             : whgb_c4_ampl_classify
```

5. Run MSA-SUM to generate the class averages for every class):

```
IMAGIC-COMMAND: msa-sum

Input images to be summed             : whgb_c4_ampl
Rootname of MSA-CLASSIFY results      : whgb_c4_ampl_classify
Output class averages                 : whgb_c4_ampl_classums
Down weight small classes             : no
Fraction of worst members to ignore   : 0
Mode of summing statistics            : none
```

6. Mask the class averages with the command MASK-IMAGE:

```
IMAGIC-COMMAND: mask-image

Option used                    : MASK_IMAGE
Mode of mask                   : cross-ring
Input file                     : whgb_c4_ampl_classums
Output file                    : whgb_c4_ampl_classums_masked
Inner, outer ring radius       : 0.3,0.95
Cross (half-) width            : 1
```

7. Now command CTF-FIND will estimate the CTF of all masked class-averages (whgh_c4_ampl_classums_masked):

```
IMAGIC-COMMAND: ctf-find

Input amplitude image file, loc#s  :
                                 whgb_c4_ampl_classums_masked
Output CTF "check" file, loc#s     : whgb_c4_ampl_half_half
Show orrelation area in half-half  : yes
Scale theor./experim. spectrum     : 0.6
PLT output file with defocus values: whgb_c4_ampl_defocus
All EM data in input header        : no
...
Pixel size                         : 4.4
Inner and outer correlation radius : 0.3,0.96
                                          inner radius of the MSA mask
Defocus search range               : 900,20000
Step size for search               : 700
Maximum astigmatism level expected : 300
Use partial coherence              : no
Generic envelope function halfwidth: 0.5
Full output                        : no
```

NOTE:

Like the MSA-RUN command CTF-FIND can be a CPU intensive and time-consuming command (even when running in MPI parallel). So think about running this command in batch mode and to create a script/batch job using commands MODE-ACCUMULATE and later MODE-STOP.

Important settings of command CTF-FIND are explained here:

- Output "found" CTF (whgb_c4_ampl_half_half): Each image in this output file will contain a) in the left half: the input amplitude image, and b) in the right half: the "estimated" CTF. These "half_half" images should be used to check the accuracy of the CTF estimation. The Thon rings of both half should fit.
- PLT output file (whgb_c4_ampl_defocus.plt): This text file will contain the estimated CTF parameters, such as defocus #1, defocus #2 and the direction of astigmatism (defocus angle). This file can be opened with a text editor to view the results of the fitting.
- Inner and outer correlation radius: A normalized cross correlation is used to compare the filtered experimental amplitude image to the theoretical CTFs. The centre and periphery of the amplitude image do not contain rings and are not important in the estimation. Therefore, the cross-correlation is only computed over a ring area specified by two radii. You can play with these parameters to obtain the best estimation - or simply try the suggested values.
- Defocus range and step size: Here you can set the parameters for the initial brute force search. The first parameter is the start of the search, the second is the end of the search and the third is the step size over which the search is conducted. You can play with these parameters to obtain the best estimation - or simply try the suggested values.

YOUR NOTES:

8. The CTF parameters determined by CTF-FIND are stored in the PLT output file (whgb_c4_ampl_defocus.plt), in the headers of the output images (whgb_c4_ampl_half_half) or in the input=output MSA class averages of the amplitude images (whgb_c4_ampl_classums_masked) but not in the headers of the micrograph images (whgb_c4_prep), which are to be CTF corrected.

   Call the command HEADER to take over the CTF/defocus parameters:

```
IMAGIC-COMMAND: headers

Options available              : takeover
Takeover options available     : class_sum_defocus
Input MSA-SUM or CTF file      : whgb_c4_ampl_classums_masked
Input classification file      : whgb_c4_ampl_classify
Input=output (header) file     : whgb_c4_prep
```

9. Before doing the CTF correction you first have to evaluate the CTF estimation done in CTF-FIND by comparing the amplitude images against the estimated theoretical amplitude images:

   DISPLAY the "half_half" output images (patches_ampl_half_half). The left half of each image shows the amplitude class averages (whgb_c4_ampl_classums_masked), the right half the estimated CTF. Compare how well the zeros match between them. If needed you can adjust the DISPLAY parameter GREVVALUES to "0.0,0.3" to better visualise the Thon rings.

   Use DISPLAY option SELECT to store the locations of the "good" class averages (for which the zeros in the "half_half" images match). Do NOT select "bad" class averages, for which the zeros do not match or which do not show any Thon rings. The selected locations are stored in a PLT file (good_classes.plt):

```
Change settings (MULT,HOR,VER,SURVEY...)[NO]: select
Output (PLT) file for loc#s                 : good_classes
...
Change settings (MULT,HOR,VER,SURVEY...)[NO]: no
```

   Select a "good" image by clicking into the image on the screen. To cancel this selection, click into the image once more. A red border indicates that the image is selected; a black box indicates that the selection was

cancelled. To get the next series of images click into the green NEXT button in the upper left corner of the display window. To stop option SELECT click into the red STOP button.

You can also write down the "good" amplitude images and use the INTERACTIVE option in the subsequent command EXCLUSIVE-COPY.

"GOOD" CLASS AVERAGES:

10. Extract the class averages of the good micrographs with command EXTRACT-IMAGE. Input is the file with the micrographs:

```
IMAGIC-COMMAND: msa-ext

Mode of operation                  : extract
MSA'd input images, NO loc#s       : whgb_c4_prep
Input classification (CLS) file    : whgb_c4_ampl_classify
Select relative to all classes     : no
Where to get the wanted classes    : plt
PLT file containing class numbers : good_classes
Output file name, NO loc#s         : whgb_c4_prep_selected
Sort output images                 ; no
Fraction of worst to ignore        : 0
Also store class-sum images        : no
```

NOTE:

Now the next step would be to CTF correct (phase flip) th "good" micrographs (whgb_c4_prep_selected) using th defocus information stored in the headers of these "good micrographs ("All defocus and EM values in headers: yes")

Here we do NOT use these defocus values. This is nc because we believe you did not do the CTF determinatio correctly ☺ but to make sure that all participants start th subsequent image processing with the same CTF correcte micrographs.

The defocus values, which we want to use are stored in a PLT file, which you can find on the network drive:

Copy the PLT file whgb_c4_defocus.plt in the data directory whgb_dataset_2016/05a_whgb_micrographs_MSA_CTF_find of the Brazil School server to your working directory.

11. Use CTF-FLIP to calculate the CTF corrected (phase flipped) micrographs (whgb_c4_flip)

```
IMAGIC-COMMAND: ctf-flip

Original images/patches NO loc#s       : whgb_c4_prep
Output file name, NO loc#s             : whgb_c4_flip
All defocus and EM values in headers   : no
Where to find the defocus parameters   : plt
PLT file containing defocus parameters : whgb_c4_defocus
Where to get the EM parameters         : header
Aperture of the objective              ! 120
Full output                            : no
```

NOTE:

You can alternatively estimate the defocus parameters with CTFFIND3 or CTFFIND4 (Linux or Mac OS X).

CTFFIND3 / **CTFFIND4** are programs provided by the Grigorieff lab (http://grigoriefflab.janelia.org/ctf) and is licensed under the terms of the GNU Public License version 3 (GPLv3). The programs are not part of **IMAGIC** but **CTFFIND3** can be used within the **IMAGIC** environment with command CTFFIND3, if the **CTFFIND3** program is installed in the FREALIGN directory of **IMAGIC**.

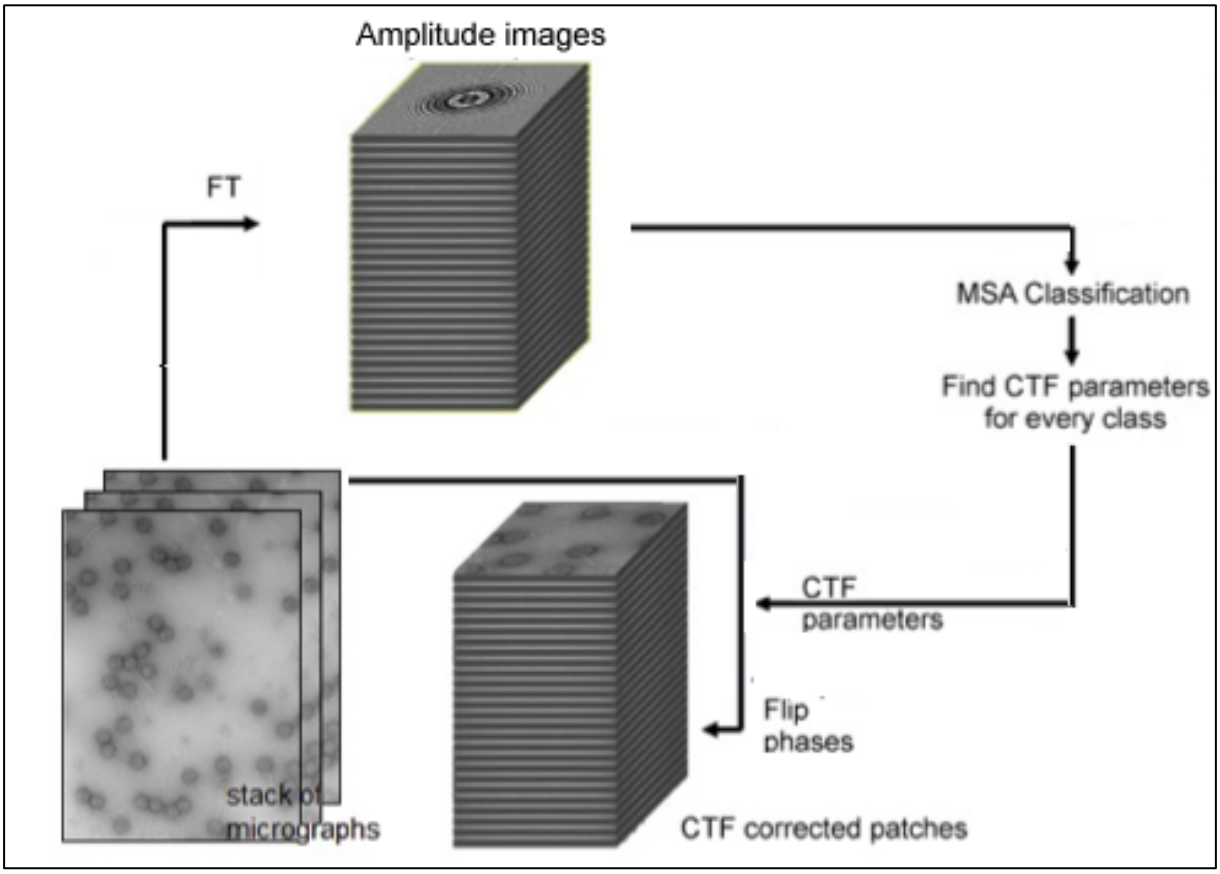
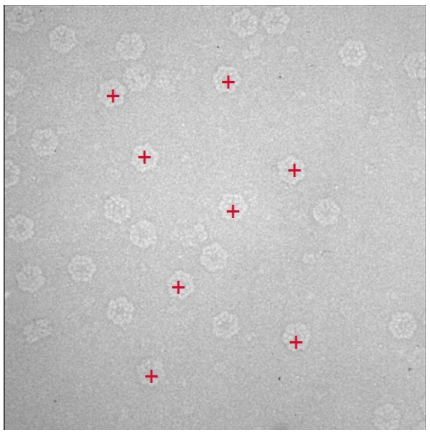


Fig. 9: Automatic CTF correction using MSA and Classification

8. Particle Picking



After CTF correction you are ready to select the particles from the stack of the CTF-corrected micrographs.

You will find 500 CTF corrected micrographs whgb_c4_flip (movie sums) on the data directory whgb_dataset_2016/06_whgb_particle_picking of the Brazil School server.

8.1. Modulation Picking

Particle picking is an essential step in image processing. When working with low-contrast images of small proteins, and/or images taken close-to-focus, in which the particles are not clearly seen, it is important to avoid any bias in the data. Therefore, it is strongly recommended to do initial particle selection using the reference-free VARIANCE/MODULATION picking (as was explained in the lectures).

Particles picked using this approach usually contain a lot of junk (ice, carbon foil, clumped particles), which can be sorted out by looking at the statistics of the images and removing the outliers, followed by MSA and classification.

As the result, one obtains low-resolution class-averages, which correspond to different views of the molecule (side, top, intermediate), which can be used as references for correlation (CCF_MATCHING) picking (chapter 8.3.)

8.2. Initial Interactive Picking

To save time, we will use INTERACTIVE picking in this practical. Results of this picking will allow you to create references for correlation (CCF_MATCHING) picking without introducing (too much) bias (chapter 8.3).

1. First low-pass filter the micrographs with the command LOW-PASS-FILTER, reducing the high-resolution details (avoid over-fitting):

```
IMAGIC-COMMAND: low-pass

Mode of operation                          : LOW_PASS_FILTER
Input file, loc#s                          : whgb_c4_flip
Output file, loc#s                         : whgb_c4_flip_lp
High frequency cut off                     : 0.3
```

2. Use command DISPLAY to check the result (whgb_c4_flip_lp). Then choose one of the micrographs with option LOCATION

```
Parameters to be changed:
NO_CHANGES(=DISPLAY), SETTINGS, OPTIONS [NO] : loc
Input location numbers (all: 0,0)            : 1     you select
```

   and press ENTER/CR to display it.

3. After having displayed the wanted location select 3-6 different particle views using the option COORD:

```
Parameters to be changed:
NO_CHANGES(=DISPLAY), SETTINGS, OPTIONS [NO] : coord
Store values in a (PLT) file                 : yes
Output coordinate (PLT) file                 :
                                    whgb_c4_particles_inter
```

   Press ENTER/CR to see the selection window. Use the mouse to select the particles and follow the instructions in DISPLAY to pick several different views.

4. Use the command CUT-IMAGE with the option APERIODIC to extract your selected particles. The coordinates file is the PLT file which you generated in DISPLAY (whgb_c4_particles_inter.plt)

```
IMAGIC-COMMAND: cut-im

Mode of operation                    : aperiodic
Input file, image loc#s              : whgb_c4_flip_lp
Output file, image loc#s             : whgb_c4_ref_0
Output image dimensions X,Y          : 128,128
Coordinates (plt) file               : whgb_c4_particles_inter
```

5. Check the extracted particles (whgb_c4_ref_0) with the command DISPLAY.

6. To reduce the influence of the neighbouring particles and background impose a soft circular mask:

```
IMAGIC-COMMAND: mask-im

Mode of operation                    : MASK_IMAGE
Mode of mask                         : soft_circular
Input file, image loc#s              : whgb_c4_ref_0
Output file, image loc#s             : whgb_c4_ref_0_m
Mask radius, drop-off (0: no mask): 0.65,0.05
```

7. Check the particles in DISPLAY. The individual particles, which you picked from the micrographs, will not be perfectly centred. To centre the particles use the command CENTER-IMAGES:

```
IMAGIC-COMMAND: cent-im

Input file, image loc#s              : whgb_c4_ref_0_m
Output file, image loc#s             : whgb_c4_ref_0_cent
Mode of operation                    : self
Correlation function wanted          : ccf
Maximal shift                        : 10
How many centring iteartions         : 3
```

```
Reduce box size (central cut)      : no
Use MPI parallelisation            : no
```

8. DISPLAY the centred particles (whgb_c4_ref_0_cent). If the particles are not well centred re-do command CENTER-IMAGES and try out other options (TOTSUM, SELF_ROTATE…).

8.3. Prepare References for Correlation Picking

Now, we are ready to create the references for correlation picking.

1. To not introduce any bias create rotational averages of the centred particles. Use command AVERAGE-ROTATIONAL:

```
IMAGIC-COMMAND: av-rot

Input file, image loc#s            : whgb_c4_ref_0_cent
Mode of output                     : image
Output file, image loc#s           : whgb_c4_ref_0_avrot
```

2. Update the circular soft mask:

```
IMAGIC-COMMAND: mask-im

Mode of operation              : MASK_IMAGE
Mode of mask                   : soft_circular
Input file, image loc#s        : whgb_c4_ref_0_avrot
Output file, image loc#s       : whgb_c4_pick_ref
Mask radius, drop-off          : 0.57,0.05
Use MPI parallelisation        : no
```

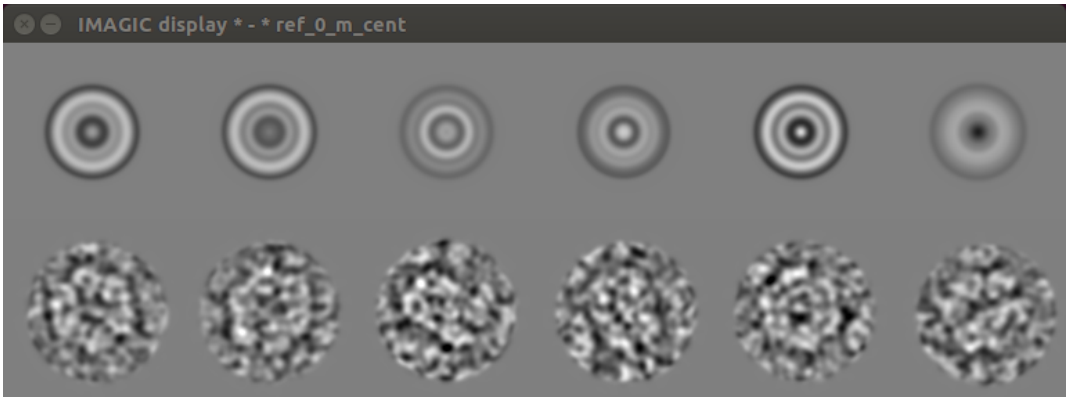


Fig. 10: Centred and rotational averaged images (upper row) of selected particle images (bottom row).

8.4. Correlation Picking

1. Run PICK-PARTICLES with option CCF_MATCHING:

```
IMAGIC-COMMAND: pick

Mode of particle detection          : ccf
Input raw images file, loc#s        : whgb_c4_flip
Store pick functions to file        : yes
Output cross correlation file       : whgb_c4_ccf_pick
Output (PLT) file with peaks        : whgb_c4_ccf_pick_coord
Extract found particles             : no
Input reference file                : whgb_c4_pick_ref
Ref. already rotationelly symmetric: yes
Max. number of particles per loc    : 120
Max. overall number expected        : 0
Minimum distance between peaks      : 64
Minimum distance X,Y from edges     : 91,91
Full output of all peak parameters : yes
```

TIP:

In general:

Try first picking only one micrograph.

DISPLAY this micrograph (option LOCATION) together with the coordinates from picking (option PLOT).

Once the parameters have been optimized pick from the whole stack.

2. Extract the particles from the CTF flipped micrographs with the command CUT-IMAGE:

```
IMAGIC-COMMAND: cut-im

Mode of operation                    : aperiodic
Input file, image loc#s              : whgb_c4_flip
Output file, image loc#s             : whgb_c4_part
Output image dimensions X,Y          : 128,128
Coordinates (plt) file               : whgb_c4_ccf_pick_coord
```

3. DISPLAY and check the extracted particle images (whgb_c4_part).

YOUR NOTES:

8.5. Extract "good" Images after Correlation Picking

PICK-PARTICLES usually also picks some unwanted objects, which you should remove.

1. To calculate the image statistics (average density, minimum, maximum, sigma) call command SURVEY-DENSITIES. Use option UPDATE_HEADER to store the results in the image headers.

```
IMAGIC-COMMAND: survey

Mode of survey                          : 2d_local
Mode of output                          : update_header
Input file                              : whgb_c4_part
```

2. First, generate a histogram of the cross-correlation coefficients (CCC) with the command HEADERS:

```
IMAGIC-COMMAND: headers

Specify option                          : histogram
Histograms from which images            : all
Histogram option                        : ccc
Number of bins for histogram            : 56
Width of histogram                      : 79
Input file                              : whgb_c4_part
```

3. PICK-PARTICLES dumps the picked particles in decreasing order of the correlation coefficient (CCC). The histogram curve shown by the command HEADERS option HISTOGRAM therefore has a bump-like structure.

   The first images usually contain edges and relate to the upper part of the histogram curve. The last images normally contain ice particles and relate to the lower part of the histogram curve. The "bad" images can therefore be easily removed. Check the histogram to find out the locations numbers (first column of numbers – in reverse order) where to cut the histogram:

```
34506     0     0   7.00E-01 |
34503     3     3   1.55E+00 |
34499     7     4   1.98E+00 |
34493    13     6   2.40E+00 |
34471    35    22   2.83E+00 |
34425    81    46   3.25E+00 |
34297   209   128   3.68E+00 |*
33968   538   329   4.10E+00 |***
33342  1164   626   4.53E+00 |******
32326  2180  1016   4.95E+00 |**********
30815  3691  1511   5.38E+00 |***************
28724  5782  2091   5.80E+00 |*********************
25781  8725  2943   6.23E+00 |******************************
22399 12107  3382   6.65E+00 |**********************************
18401 16105  3998   7.08E+00 |****************************************
14482 20024  3919   7.50E+00 |****************************************
10942 23564  3540   7.93E+00 |************************************
 8071 26435  2871   8.35E+00 |*****************************
 5919 28587  2152   8.78E+00 |*********************
 4285 30221  1634   9.20E+00 |****************
 3211 31295  1074   9.63E+00 |**********
 2477 32029   734   1.01E+01 |*******
 1982 32524   495   1.05E+01 |*****
 1602 32904   380   1.09E+01 |****
 1334 33172   268   1.13E+01 |***
 1142 33364   192   1.18E+01 |**
  990 33516   152   1.22E+01 |*
  865 33641   125   1.26E+01 |*
  727 33779   138   1.30E+01 |*
  632 33874    95   1.35E+01 |*
  534 33972    98   1.39E+01 |*
  453 34053    81   1.43E+01 |*
  388 34118    65   1.47E+01 |*
  325 34181    63   1.52E+01 |*
  283 34223    42   1.56E+01 |
    0 34506   283   1.64E+01 |***
```

4. DISPLAY the related images and find out the start and end location numbers of the "good" images.

YOUR NOTES:

First good location:

Last good location:

5. Extract the good images with the command EXTRACT-IMAGES:

```
IMAGIC-COMMAND: extract-im

What should be copied              : 2d_images
Exclusive copy operations          : EXTRACT
Input file                         : whgb_c4_part
Output file                        : whgb_c4_part_sort_1
Source of image locations          : interactive
Location numbers wanted            : 1-...              your choice
Numbers wanted                     : all
```

NOTE:

You can use other parameters (sigma in image densities, average density etc.), to exclude "bad" images.

If time is restricted you can skip the following SORT and EXCLUDE parts and continue with pre-treatment (chapter 9.).

6. If wanted use the SIGMA parameter to exclude "bad" images

```
IMAGIC-COMMAND: sort-im

What should be copied              : 2d_images
Exclusive copy operations          : SORT
Input file                         : whgb_c4_part_sort_1
Output file                        : whgb_c4_part_sort_2
Source of SORT values              : header
Criteria for SORT                  : sigma
Sort UP or DOWN                    : up
How many sorted images wanted      : 0                  0 : all
Numbers wanted                     : all
```

Again look at the histogram in HEADERS option HISTOGRAM now using SIGMA

```
IMAGIC-COMMAND: headers

Specify option                        : histogram
Histograms from which images          : all
Histogram option                      : sigma
Number of bins for histogram          : 56
Width of histogram                    : 79
Input file                            : whgb_c4_part_sort_2
```

and check if there are bad images. As before, use DISPLAY to select.

```
8724     1     1   7.37E+00 |
8721     4     3   7.62E+00 |
8717     8     4   7.75E+00 |
8706    19    11   7.87E+00 |*
8690    35    16   7.99E+00 |*
8676    49    14   8.12E+00 |*
8655    70    21   8.24E+00 |*
8606   119    49   8.37E+00 |**
8544   181    62   8.49E+00 |***
8452   273    92   8.61E+00 |****
8299   426   153   8.74E+00 |*******
8079   646   220   8.86E+00 |**********
7786   939   293   8.98E+00 |*************
7347  1378   439   9.11E+00 |********************
6803  1922   544   9.23E+00 |*************************
6107  2618   696   9.36E+00 |*******************************
5297  3428   810   9.48E+00 |************************************
4381  4344   916   9.60E+00 |*****************************************
3507  5218   874   9.73E+00 |****************************************
2682  6043   825   9.85E+00 |************************************
1935  6790   747   9.98E+00 |*********************************
1392  7333   543   1.01E+01 |************************
 916  7809   476   1.02E+01 |**********************
 603  8122   313   1.03E+01 |**************
 360  8365   243   1.05E+01 |***********
 224  8501   136   1.06E+01 |******
 147  8578    77   1.07E+01 |****
 100  8625    47   1.08E+01 |**
  76  8649    24   1.10E+01 |*
  53  8672    23   1.11E+01 |*
  36  8689    17   1.12E+01 |*
  25  8700    11   1.13E+01 |*
  18  8707     7   1.15E+01 |
  11  8714     7   1.16E+01 |
   7  8718     4   1.17E+01 |
   0  8725     7   1.20E+01 |
```

YOUR NOTES:

First good location:

Last good location:

As before extract the good images with EXCLUSIVE-COPY.

```
IMAGIC-COMMAND: extract-im

What should be copied                : 2d_images
Exclusive copy operations            : EXTRACT
Input file                           : whgb_c4_part_sort_2
Output file                          : whgb_c4_part_sort_3
Source of image locations            : interactive
Location numbers wanted              : 1-...          your choice
Numbers wanted                       : all
```

7. As already mentioned, further parameters to exclude "bad" images are: AVDENS (average density), MINDENS (minimal density value) and MAXDENS (maximal density value). If wanted sort/extract as shown for parameter SIGMA.

8. Finally call the file containing your best images whgb_c4_part_best.

YOUR NOTES:

NOTE:

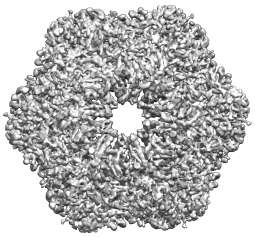
To "synchronise" the data of all Brazil School participants we do not continue with the created best particles file (whgb_c4_part_best).

We are going to copy a file with picked particles from the Brazil School network drive so that all participants will continue with the same images.

Please download the files whgb_c4_part_0 from the data directory whgb_dataset_2016/06_whgb_particle_picking of the Brazil School server.

The copied file (whgb_c4_part_0) contains picked/boxed particles, which we have prepared for you.

YOUR NOTES:



9. Pre-Treatment

Very often at this stage of the analysis you would call the command PREPARE-IMAGES to pre-treat the boxed particles (whgb_c4_part_0). As you already know PREPARE-IMAGES band-pass filters, normalizes and zero-float the data set. And finally, it masks the images.

PREPARE-IMAGES was already applied onto the micrographs. So this pre-treatment is no more needed here.

1. But it usually is a good idea to normalize and mask the particle image (whgb_c4_part_0). Call the command PRETREAT-IMAGES with options NORM_VARIANCE and CIRCLE:

```
IMAGIC-COMMAND: pretreat-im

Mode of operation                    : PRETREAT
Please specify option                : norm_variance
Type of variance mask                : circle
How to use the norm variance mask    : always
Input file, loc#s                    : whgb_c4_part_0
Output file, loc#s                   : whgb_c4_part_filt
Mask radius,drop-off                 : 0.75,0.05
Desired new sigma                    : 10
```

YOUR NOTES:

10. Alignment-by-Classification

Alignment by classification is a method by which you can obtain class averages by aligning the particles to references generated from the data set itself. Neither external references nor references generated from 3-D volumes are used at this stage. It consists of the following steps:

- Centring (chapter 11)
- MSA classification (chapter 12)
- Sometimes followed by a Multi-reference alignment (MRA) of the particles against the (selected) class average. We do not use this step here.

11. Centring

The first step in alignment-by-classification is to do centring. This means all the images will be aligned translationally (but not rotationally) to the total sum of the data set. As you already know, the command to do this is CENTER-IMAGE:

```
IMAGIC-COMMAND: center-image

Input file, image loc#s              : whgb_c4_part_filt
Output file, image loc#s             : whgb_c4_part_cent
Options for centering                : totsum
Correlation functions available      : ccf
Max shift (pixels or as fraction)    : 3
Number of centering iterations       : 3
Options to filter the total sum      : low
Halfwidth value for low-pass filter  : 0.1
Mask radius, drop-off (0: no mask)   : 0.8,0.05
Reduce box size (central cut)        : no
```

Check the results with DISPLAY. If the particles are not well centred re-do command CENTER-IMAGES and try out other options (TOTSUM, SELF_ROTATE…)

12. Multivariate Statistical Analysis (MSA) Classification

The aim of MSA classification is to find similar images (views of the particle) so that we can average them to reduce the noise level (improve the signal-to-noise ratio "SNR") and to find the "typical" views, which we would like to use to calculate a 3-D reconstruction.

This step is performed using three different commands: MSA-RUN, which performs an eigenvector data compression ("hyper space"), MSA-CLASSIFY which classifies similar images into groups of similar images ("classes") and MSA-SUM which performs the averaging into class averages (class-sums).

1. In order to run MSA classification we must create a mask indicating, which parts of images are to be analysed ("area of interest"). Only pixels falling within this mask are used for the classification analysis, so that we can make it focus on the actual particles and not on the surrounding noise. Call command MSA-MASK:

```
IMAGIC-COMMAND: msa-mask

Mode of operation                 : total_sum
Input header file (no loc#s)      : whgb_c4_part_cent
Output filename, image loc#s      : whgb_c4_msamask
Additional mask                   : circular
Circular mask radius              : 0.6
```

2. DISPLAY the MSA mask (whgb_c4_msamask) and check if the circular mask fits the shape of the total sum of all particles. If not re-do MSA-MASK and use another radius parameter.

3. Next call MSA-RUN. Bear in mind that this will take some time to run:

```
IMAGIC-COMMAND: msa-run

Choose mode of operation          : fresh
MSA distance                      : modulation
Input (= output) "images"         : whgb_c4_part_cent
Input MSA mask file               : whgb_c4_msamask
Eigenimages output file           : whgb_c4_eigen
```

```
Use default MSA options          : yes
Number of iterations             : 100
Number of eigenimages            : 60          "real science": more
Rootname for results files       : whgb_c4_msa
```

In another terminal/command window you can use option WATCHDOG of the command DISPLAY to continuously display the eigenimages (whgb_c4_eigen) as they are being updated throughout the MSA-RUN iterations.

4. When MSA-RUN had finished look at the eigenimages (whgb_c4_eigen) in DISPLAY. The eigenimages of a centred dataset are a good way of examining the information content of a dataset.

   The worm-hemoglobin has D6/622 symmetry and the images were not yet rotationally aligned, so you should find eigenimages that are rotated to each other (like a sine and cosine wave, as the 2nd and 3rd in the example below) showing this 6-fold cyclical symmetry.

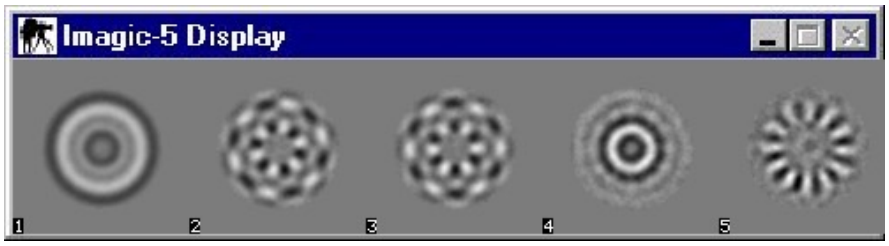


Fig. 11: First eigen-images of the (rotational) unaligned worm hemoglobin data set. (Remember that worm hemoglobin has D6/622 symmetry)

Note that the first eigenimage always shows (a sort of) average of all images.

5. The classification is performed with the command MSA-CLASSIFY. The number of classes you choose is related to the average number of images per class you would like. You can play with this value to see how the quality of the classes is affected. Ideally, you would have as few members per class as possible whilst still obtaining high contrast class averages.

```
IMAGIC-COMMAND: msa-classify

Input to be classified          : images
Classification option:          : hac
Input (= output) file           : whgb_c4_cent
Percentage to be ignored        : 0
Active eigenimages for classify : 60
Use default classify options    : yes
Number of classes wanted        : 1000                  your choice
Name of output results files    : whgb_c4_classify
```

The images should now have been grouped into the wanted classes each of which should contain ~20 similar images.

6. After MSA-CLASSIFY average all the particles that belong to the same class. Call the command MSA-SUM:

```
IMAGIC-COMMAND: msa-sum

Input images to be summed       : whgb_c4_cent
Rootname of input classify files: whgb_c4_classify
Output class averages           : whgb_c4_classums
Downweight small classes        : yes
Fraction of members to ignore   : 0.1
Mode of summing statistics      : none
```

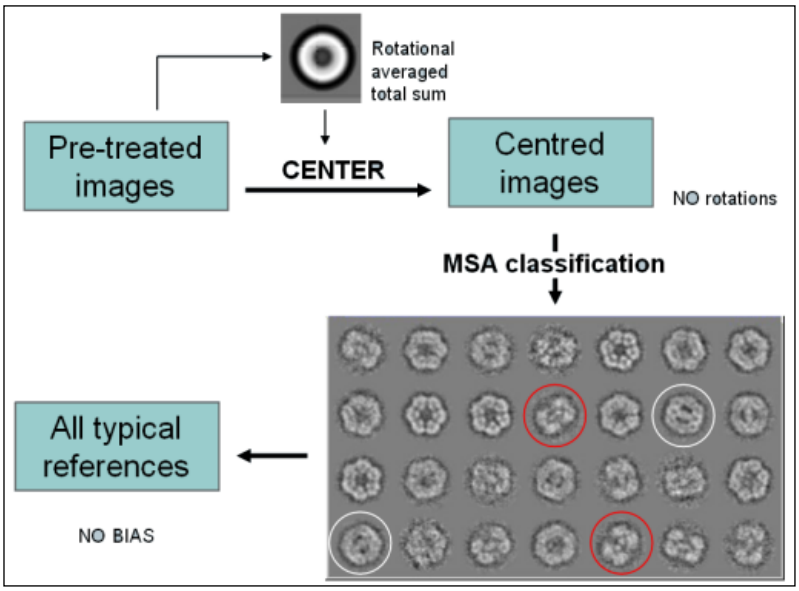


Fig. 12: Alignment by Classification

7. DISPLAY the class averages (whgb_c4_classums) created in MSA-SUM.

> REMEMBER:
>
> You should DISPLAY the class averages using the same grey value for all images. Either use option GREYVALUE with SURVEY and GLOBAL or GREYVALUE with option INTERACTIVE.

You will see that there are a lot of "good" showing particles views with high resolution but also a number of "bad" class averages showing classes with low resolution.

8. Good criteria to exclude "bad" class averages are the number of members per class (NUMCLS) and the overall class quality (OVQUAL) (refer to chapter (8.3)).

First sort by number of class members (NUMBER_OF_CLASS_MEMBERS) with command SORT-IMAGE:

```
IMAGIC-COMMAND: sort-image

What should copied                : 2d
Exclusive copy operation          : SORT
Input file                        : whgb_c4_classums
Output file                       : whgb_c4_classums_sort_num
Source of SORT values             : header
Criteria for SORT                 : number
Sort UP or DOWN                   : down
How many of sorted images         : 0                    0: all
```

9. Again look at the histogram of the number of class members in HEADERS option HISTOGRAM now using ALL_IMAGES, INDEX, LABEL, NUMCLS and check if there are bad classes (refer to chapter (8.3)). DISPLAY the sorted class averages (whgb_c4_classums_sort_num) and find out the "bad" classes

YOUR NOTES:

Last class location with enough members:

10. As usual, extract the good class averages with the command EXTRACT-IMAGE. Check the extracted images in DISPLAY.

11. Next use the command SORT-IMAGE to once more sort the good averages, now by classification over all quality (OVQUAL) (refer to 8.). As before (9.) create a histogram (HEADERS with options HISTOGRAM and ALL_IMAGES, INDEX, LABEL, OVQUAL) and check if there are bad classes.

YOUR NOTES:

Last class location with good quality:

12. As before (10). extract the good class averages with the command EXTRACT-IMAGE. Check the extracted images in DISPLAY.

13. Name your final "best" class averages whgb_c4_classums_best.

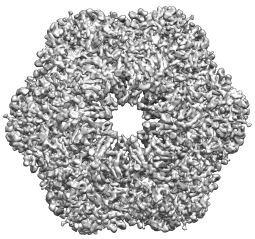
NOTE:

Depending on you data set you will continue with multi-reference alignment (chapter 13) or immediately continue with angular reconstitution and 3-D reconstruction (chapter 16).

In a "real science" analysis of worm hemoglobin you would usually skip the multi-reference part here, because the particle is well centred and highly symmetric.

Also here in the course you will continue with angular reconstitution and 3-D reconstruction.

YOUR NOTES:



13. Angular Reconstitution - Initial Angular Assignment

Once you have selected the best class averages you need to find their relative orientation (Euler angles).

Worm hemoglobin is a molecule with D6/622 point-group symmetry. This high degree of symmetry makes the initial angular assignment much easier than with lower degrees of symmetry.

NOTE:

Usually you would create the first initial 3-D volume with an automatic random start-up command.

To better understand what such "automatic" programs are doing, we are first performing all steps one after the other before using the automatic RANDOM-STARTUP option.

13.1. Prepare Class Averages

1. Centre the class averages (whgb_c4_classums_best). This can help with the accuracy of the angular assignment. Call CENTER-IMAGE:

```
IMAGIC-COMMAND: center-image

Input file, image loc#s          : whgb_c4_classums_best
Output file, image loc#s         : whgb_c4_classums_cent
Options for centering            :                        your choice
...
```

DISPLAY the centred images (whgb_c4_classums_cent) to check if the chosen centring option worked correctly (give a **?** when asked for the option and/or refer to chapter 11)).

Play around with the various centring options Finally you should use the option SELF or TOTSUM.

2. Mask the centred images:

```
IMAGIC-COMMAND: mask-image

Mode of operation:            : MASK_IMAGE
Mode of mask                  : soft
Input file, image loc#s       : whgb_c4_classums_cent
Output file, image loc#s      : whgb_c4_classums_masked
Mask radius, drop-off         : 0.6,0.05              a tight mask
```

Check the mask with DISPLAY.

13.2. Angular Reconstitution - Self Search

The next step is to check how well each single class average conforms to the given point-group symmetry of the particles.

The idea of option SELF_SEARCH in command ANGULAR-RECONSTITUTION is to sort the class-averages with the smallest residual for the given point-group symmetry to start up a 3-D reconstruction. Each class-average image is examined exclusively with respect to itself. This option only works for highly symmetric point-groups like the D6/622 symmetry of worm hemoglobin.

Remember that the class average images (whgb_c4_classums_masked) should contain good classes of all typical views.

1. Call ANGULAR-RECONSTITUTION, option SELF_SEARCH:

```
IMAGIC-COMMAND: ang-rec

Point-group symmetry           : d6
Minimal stay-away from equator : 10
Option for angular reconst.    : self
```

```
Mode of output                    : update_header
Input (classum) images            : whgb_c4_classums_masked
Sinogram file, image loc#s        : none
Output sinecorr file              : none
ASQ filter the sinogram lines     : yes
Linear mask radius of sinograms : 0.6
Wanted angular increment          : 2
Full output of the results        : no
```

At the end of the SELF_SEARCH calculations you will find a list like this:

```
  ======================================
  SELF_SEARCH Euler angles (sorted list)
  ======================================

          ERROR     LOC
  #         (%)       #      ALPHA       BETA        GAMMA

  1 :>    e.eeee      nn     aaa.aaa     bbb.bbb     ggg.ggg
  2 :>    e.eeee      nn    -aaa.aaa     bbb.bbb     ggg.ggg
  ...

         Average error of the set = e.eeeee %
```

2. Now DISPLAY the class averages (whgb_c4_classums_masked) with options NAME and EULER so that the images are displayed with location number and Euler angles as just estimated (do not forget to later set this option back to LOCATION!).

3. For the subsequent 3-D reconstruction you should select a few good class-averages with very different Euler angles Beta:

   a) First select two intermediate view. Both views should look like intermediate views and should have "intermediate" Euler angles Beta (around 40°-60°)

   b) Next select a (close-to-) side view, which are the ones that have rectangular like shape. Try to find such a view, whose Euler angle Beta is around 70-80°.

   c) Finally select a (close-to-) top view (round shape) with a Beta angle around 10°.

   d) Select 2-3 other intermediate or side views.

LOCATION NUMBERS:

Intermediate:

Close to side:

Close to top:

IMPORTANT NOTE:

SELF-SEARCH is NOT an Euler angles determination but rather a consistency check to see how well each single image conforms to the given point-group symmetry.

Here SELF-SEARCH is only used to find intermediate views with different Euler angles (especially intermediate views), which you can use as input to ANGULAR-RECONSTITION option NEW_IMAGE.

13.3. Angular Reconstitution - New Projections

To assign Euler angles to the selected class averages use option NEW of command ANGULAR-RECONSTITUTION. In contrast to SELF_SEARCH, the Euler angles of each new image are now calculated in relation to the Euler angles of all images, which already have assigned Euler angles:

1. Call ANGULAR-RECONSTITUTION, option NEW:

```
IMAGIC-COMMAND: ang-rec

Point-group symmetry                : d6
Minimal stay-away from equator      : 10
Option for angular reconstitution: new
Option of NEW                       : fresh
```

```
Input (classum) images, NO loc#s : whgb_c4_classums_masked
Location numbers wanted          :                 you choose:
                                                   3 of the selected
                                                   class loc#s
                                                   seperated by ";"s
Output (selected) image file     : whgb_c4_select_1
Output sinogram file, NO loc#s   : whgb_c4_sino
ASQ filter the sinogram lines    : yes
Linear mask radius for sinograms : 0.6
Output sinecorr file, NO loc#s   : whgb_c4_sinecorr
Wanted angular increment         : 2
Full output of the results       : no
```

2. The ouput file (whgb_c4_select_1) contains all selected images with Euler angles stored in their headers. This file is created because Euler angles were not assigned one-by-one to each image in the input file.

3. Remember that in option SELF_SEARCH each image was assigned an Euler angle exclusively with respect to itself. In contrast, option NEW Euler angles are assigned to each image with respect to itself and to all other previous images with assigned Euler angles.

4. After ANGULAR-RECONSTITUTION has finished DISPLAY the selected classums (whgb_c4_select_1) with option NAME and EULER, in which case loc#s and the Euler angles will be printed. Check that the angles obtained make sense. The beta and gamma angles should be different and apart from each other for at least 50 degrees. If all Beta angles are the same (usually close to 90° - ("Minimal stay-away from equator") your first three class averages are too similar and you should try another combination of first three images, always starting with an intermediate view.

5. Assign Euler angles to the next (2 or 3) selected good class averages: Call ANGULAR-RECONSTITUTION, options NEW and ADD:

```
IMAGIC-COMMAND: ang-rec

Point-group symmetry             : d6
Minimal stay-away from equator   : 5
Option for angular reconstitution: new
Option of NEW                    : add
Input (classum) images, NO loc#s : whgb_c4_classums_masked
```

```
Location numbers wanted                 :                    you choose:
                                                             next selected
                                                             class loc#s
Output (selected) image file            : whgb_c4_select_1
Output sinogram file, NO loc#s          : whgb_c4_sino
ASQ filter the sinogram lines           : yes
Linear mask radius for sinograms        : 0.55
Output sinecorr file, NO loc#s          : whgb_c4_sinecorr
Wanted angular increment                : 2
Full output of the results              : no
```

6. If the Euler angles look okay, assign Euler angles to the all remaining selected good class averages:

```
...
Option for angular reconst.             : new
Option of NEW                           : add
Input (classum) images, NO loc#s        : whgb_c4_classums_masked
Location number(s) wanted               :
                                                    remaining selected
                                                    class loc#s
Output (selected) image file            : whgb_c4_select_1
...
```

NOTE:

There is another option, which should be mentioned here, although you will normally not use it during this practical.

You can remove bad images (with a too high ERROR) by using ANGULAR-RECONSTITUTION with options NEW and REMOVE:

```
Option of NEW                           : remove
Re-calculate Euler angles               : yes
Location number(s) wanted               :                  loc#s in the
                                                           selected file,
                                                           you choose
```

14. Initial 3-D Reconstruction

1. Once you have assigned angles to your (selected) class averages you are ready to build your first 3-D using the command THREED-RECONSTRUCTION:

```
IMAGIC-COMMAND: th-reconst

Mode of 4D operation                    : all_in_one
Point-group symmetry                    : d6
Use default 3D reconstruction options   : yes
Input 2D (classum) images               : whgb_c4_select_1
Source of Euler angles                  : angrec_header
Update output header                    : no
Output file for 3D reconstruction       : whgb_c4_3d_1
Output file for re-projections          : whgb_c4_repro_1
Output file for error projections       : whgb_c4_err_1
Spherically mask the reconstruction     : yes
Radius of the mask                      : 0.6
Hamming window factor                   : 0.5
Object size as fraction of image size   : 0.8
Also create a normalized 3D volume      : yes
Give new sigma                          : 1        helpful in Chimera
```

2. It is important to check "by eye" how well the re-projections match the class averages.

   To do this, DISPLAY the class averages (whgb_c4_select_1) which you used to create the 3-D in one window and the re-projections (whgb_c4_repro_1) in another.

   The DISPLAY settings such as SCALE should be the same in both windows.

   By flicking back and forth between the two overlaid windows compare how well these two match. If they do not match, the angular assignment was

not correct and you should re-run ANGULAR-RECONSTITUTION and THREED-RECONSTRUCTION with other class averages.

3. DISPLAY and check the sections of the 3-D volume. Do not forget to use the GREYVALUE options SURVEY and 3D_LOCAL. May be, you also want to have a look at a surface representation in **Chimera** (refer to chaper 16.3).

NOTE:

You can visualize a 3-D volume with the command DISPLAY. The display will show slices through the 3-D from bottom to top.

To look at surface views of the 3-D volume use the commands THREED-SURFACE, MOVIE or the program **CHIMERA**.

Please refer to chapter 17.

4. Use the error listing at the end of THREED-RECONSTRUCTION to remove bad" class averages from your selected images (whgb_c4_select_1).

BAD CLASS AVERAGES:

As before, exclude the "bad" class averages with the command EXCLUDE-IMAGE. Input files are the last class averages (whgb_c4_select_1). Output will be whgb_c4_select_1_best. Redo THREED-RECONSTRUCTION with the new class averages whgb_c4_select_1_best.

5. Check the updated 3-D volume with DISPLAY or in **Chimera** (chaper 16.3).

6. The resulting 3-D volume may still look very "artificial", which usually is due to the small number of class averages which were used. To get rid of those artifacts it can be helpful to low-pass filter the 3-D volume:

7.

```
IMAGIC-COMMAND: th-filter

3D filter option                          : lowpass
Input 3D volume file                      : whgb_c4_3d_1
```

```
Output file containing masked input 3D : whgb_c4_3d_1_lp
High frequency cut-off                 : 0.2
```

8. Check the filtered 3-D volume with DISPLAY or in **Chimera** (chaper 16.3).

9. If you need a normalised 3-D volume and did not create it within THREED-RECONSTRUCTION you can still normalise it with the command THREED-NORM-VARIANCE:

```
IMAGIC-COMMAND: th-norm

Mode of operation                  : NORMVAR
Input file, 3D loc#s               : whgb_c4_3d_1_lp
Output file, 3D loc#s              : whgb_c4_3d_1_lp_norm
Desired new sigma                  : 1          helpful in Chimera
```

YOUR NOTES:

15. Angular Reconstitution - Random Start-Up

In chapter 13.3 you had choosen a number "good" class averages, which were used to assign Euler angles, using three class averages as a starting point and adding new images.

This procedure is automated in the option RANDOM_STARTUP of the command ANGULAR-RECONSTITUTION. Each image is assgined Euler angles by using the other images as a reference ("anchor-set"). The starting Euler angles are assigned randomly to all images. Please refer to the lectures.

Remember that the input images (whgb_c4_select_1) are the "best" and most typical images, which also have to be well centred.

1. Call the command ANGULAR-RECONSTITUTION, option RANDOM_STARTUP:

```
IMAGIC-COMMAND: ang-rec

Point-group symmetry                : d6
Minimal stay-away from equator      : 10
Option for angular reconst.         : random_startup
How are the images available        : images
Input (=output) image file          : whgb_c4_select_1
Sinogram file                       : whgb_c4_sino
Apply 1D filter to sinogram lines   : no
ASQ filter the sinogram lines       : no
Linear mask radius for sinograms    : 0.55                  tight
Delete output sinograms             : yes
Wanted angular increment            : 2
Random number generator seed        : 1          (later: old value + 1)
Number of iteration steps           : 25
Full output of the results          : no
```

2. Reconstruct the 3-D volume with the command THREED-RECONSTRUCTION (refer to chapter 14). Output file will be whgb_c4_3d_startup_1, for example.

3. As before check in DISPLAY if the input images and the re-projections are the same.

4. Also visualize the 3-D results with DISPLAY or in **Chimera** (chaper 16.3).

5. Compare the new reconstruction (whgb_c4_3d_startup_1) with the previous results (whgb_c4_3d_1).

6. If necessary, redo angular-reconstitution / random-start-up with another combination of input images and/or another seed for the random number generator.

7. As before you can filter the 3-D volume with THREED-FILTER.

YOUR NOTES:

16. 3-D Visualization

Before you continue, a few explications on how to visualize a 3-D volume.

16.1. 2-D Sections of the 3-D volume

The 3-D volume is stored as a stack of 2-D sections stored in one **IMAGIC** image file.

Usually the command DISPLAY is used to visualize these 2-D sections. Note that the sections are displayed from bottom to top.

16.2. 3-D Surface Views

Instead of looking at the sections of the 3-D volume you can also create surface representations of the 3-D volume.

1. Use the command THREED-SURFACE:

```
IMAGIC-COMMAND: th-surf

Input 3D file, ONE 3D loc#s              : whgb_c4_3d_1_lp
Output file for 2D surface view(s)       : whgb_c4_3d_1_surf
Threshold 3D density value               : 0.05
                                                 important!! see below
                                                 you select
Choose projection option                 :
                                                 you select
                                                 SPIRAL for example
...
Blow-up 3D volume before 3D rendering : yes
3D blow-up dimension                  : 256
...
Default rendering parameters             : yes
...
```

It is very difficult to define the correct surface of an object. Playing around with the threshold (surface rendering) value will result in different looking surface views. A very high threshold value would wipe out the particle; a very low one would keep all the sensible and non-sensible parts.

THREED-SURFACE prints a protein mass value (in kDalton), which was calculated according to the specified threshold value:

```
Threshold value used for depicting 3D volume      :       0.250
Number of voxels with density > threshold         :        7962

With a scale (Angstrom per pixel) of              :      10.500
  this corresponds to cubic Angstrom  > threshold : 9217010.000
Assuming a protein density (Dalton/cub.Angstrom  ):       0.844
  this corresponds to a protein mass (kDalton) of :    7779.157
Scale (Angstrom per voxel) was specified by       :        user
```

You can play around with the threshold value until the kDalton value for your particle is correct. Please don't take this value too seriously; it is only a helping hint!

2. Use DISPLAY to look at the surface views.

3. If you have created a sequence of surface view images (with option SPIRAL, TOMOGRAPHY etc.) you can use the command MOVIE to display the surface representation images in an endless loop.

   Call MOVIE and answer the related questions. When the movie is displayed move the cursor into the image and click with the right mouse button to get the control panel. If you have created STEREO IMAGES click the "**oo**" switch in the control window to get moving stereo images. Now roll your eyes and try to see neighboured images in stereo (3-D). Use the "●" switch to leave MOVIE.

YOUR NOTES:

16.3. Use Chimera

Chimera is a nice non-IMAGIC program to visualise 3-D volumes.

If you did not create a normalised 3-D volume in THREED-RECONSTRUCTION it might be helpful to normalize the 3-D volume to a sigma of 1 using the command THREED-NORM (chapter 14).

Chimera has an **IMAGIC** plugin and can read **IMAGIC** maps (use the **IMAGIC** file with the extension .hed. The extension .img will also work).

If you need to convert the **IMAGIC** map to MRC format use the IMPORT-EXPORT command (same as EM2EM):

```
IMAGIC-COMMAND: em2em

Convert 2D images or 3D volumes          : 3d
Data format of the input to be converted: imagic
Export to which data format              : mrc
How to store output 3D volume            : 3d_volume
Input 3D image file                      : whgb_c4_3d_1_norm
Output 3D image file                     : whgb_c4_3d_1.mrc
...
Use standard em2em coordinate conversion: yes
In case of conflicts, which preference   : change_format
How to get the image names/titles        : name_of_import
```

Run **Chimera** with whgb_c4_3d_1_norm.hed or whgb_c4_3d_1.mrc as input file.

17. Angular Reconstitution - Anchor-Set

17.1. Align Input Images to their Re-Projections

To improve the 3-D reconstruction you can apply the following refinement step:

The input class averages used for 3-D reconstruction can be aligned to the related re-projections created in THREED-RECONSTRUCTION. The aligned input class averages can be used to calculate a refined 3-D reconstruction.

> **NOTE:**
>
> The re-projections were created from the same 3-D volume which means that they are perfectly "3-D aligned".

Call ALIGN-PARALLEL:

```
IMAGIC-COMMAND : ali-para

Alignment modes available       : both
Start option                    : translation_first
Correlation functions available : ccf
Input file, image loc#s         : whgb_c4_select_1
Output file, image loc#s        : whgb_c4_select_1_alipara
Reference file, image loc#s     : whgb_c4_repro_1
Max shift                       : 0.1
Min, max rotation angle         : -180,180
Precision for rot. alignment    : medium
Min,max radius for rot. align   : 0,0.6
Maximum number of iterations    : 3
```

Re-do the THREED-RECONSTRUCTION now using the aligned selected class averages (whgb_c4_select_1_alipara) as input file. Name the output files whgb_c4_3d_1_alipara, whgb_c4_repro_1_alipara…).

17.2. 3-D (Automatic) Masking

Use the command THREED-AUTO-MASK to automatically generate a mask for the 3-D volume.

THREED-AUTO-MASK calculates a modulation volume, which will be binarised. Use MODULATION as opposed to VARIANCE (check the help for this question if you are curious why to do this).

1. Call THREED-AUTO-MASK.

   The filter parameters for the modulation calculation need to be specified manually. Note that the low-pass filter parameter needs to be greater-equal the low bound of the band-pass. The higher this filter is set, the "finer/sharper" the mask will be. Use the ? before answering any question.

```
IMAGIC-COMMAND: th-auto-mask

Automasking options                    : do_it_all
Input 3D volume file                   : whgb_c4_3d_1_alipara
Output file with masked input 3D       : whgb_c4_3d_1_masked
Output modulation/variance volume      : whgb_c4_3d_1_modvar
Output file containing 3D mask         : whgb_c4_3d_1_mask
Masking based on local modulation      : yes
Band-pass parameters                   : 0.05,0.25
Low-pass filter parameter              : 0.04
Threshold options                      : automatic
Auto-threshold percentage              : 16
```

2. DISPLAY the output modulation volume (whgb_c4_3d_1_modvar) and the masked 3-D volume (whgb_c4_3d_1_masked). Also use THREED-SURFACE and MOVIE or **Chimera** to visualize the results.

3. Then, start THREED-AUTO-MASK again, change the filter parameters (for example, change the low-pass filter parameter to 0.1), and observe how the modulation and the masked 3-D volume is affected.

4. An ideal mask removes noise outside of the object, but leaves your object completely intact.

18. Angular Reconstitution - Anchor-Set

You have created a 3-D volume, which can be used to refine all previous image-processing steps.

First use the 3-D volume to get all "typical" views with command THREED-FORWARD-PROJECTION. The resulting "2-D forward projections" have well-defined Euler angles and can serve as references (a so called "anchor-set") to refine the Euler angles of the class averages.

1. You will create the forward projections with command THREED-FORWARD-PROJECTION. Remember to forward project in the asymmetric triangle of D6:

```
IMAGIC-COMMAND: thr-forw

Option used                              : FORWARD
Input 3D image file                      : whgb_c4_3d_1_masked
Output file for forward projections      : whgb_c4_arset_1
Threshold 3D density value               : -9999
Use default interpolation mode           : yes
Choose Euler angles option               : asym_triangle
Point-group symmetry to be used          : d6
Option to chose Euler angles             : random
Number of projections wanted             : 10
Minimum angular distance                 : 3.0
Also generate mirror projections         : no
Option for Euler angle alpha             : zero
Random number generator seed             : 0
Full output of all parameters            : no
```

2. Assign Euler angles to ALL class averages (whgb_c4_classums_masked) using the anchor set (whgb_c4_arset_1):

```
IMAGIC-COMMAND: ang-rec

Point-group symmetry                : d6
Minimal stay-away from equator      : 0
Option for angular reconstitution : anchor_set
Option of ANCHOR_SET                : fresh
Anchor set options                  : single_anchor
How are the input images available: images
Input(=output) (classum) images     : whgb_c4_classums_masked
Sinogram file, image loc#s          : sino
ASQ filter the sinogram lines       : yes
Linear mask radius for sinograms    : 0.55
How is the anchor set available     : images
Input anchor set IMAGES             : whgb_c4_arset_1
Output anchor set sinograms         : whgb_c4_arsino_1
Output sinecorr file, NO loc#s      : whgb_c4_sinecorr
Delete output sinecorr file(s)      : yes
Wanted angular increment in search: 4
Criterion for peak search           : fisher_transform
...e
Output of results                   : final_output
Print histograms                    : no
```

You will find error values (printed on the screen) to give you an idea about the quality of the Euler angle assignment of each class average:

```
          ====================================
              EULER results (sorted list)
          ====================================

          ERROR      LOC
  #         (%)       #      ALPHA       BETA        GAMMA

  1 :>   e.eeee      nn     aaa.aaa     bbb.bbb     ggg.ggg
  2 :>   e.eeee      nn    -aaa.aaa     bbb.bbb     ggg.ggg
         ...
         Average error of the set = e.eeeee %
```

3. You will now look for the class averages, which Euler assignments are best. In general, class averages with a low angular error tend to be better.

   Use the angular reconstitution error to select some 50-100 of the "best" class averages using the command SORT-IMAGE with criterion ANGREC_ERROR and option UP, so that the class averages with the lowest angular reconstitution error will be at the beginning of the output file.

```
IMAGIC-COMMAND: sort-image

What should be copied          : 2d_images
Exclusive copy operation       : SORT
Input file, NO loc#s           : whgb_c4_classums_masked
Output file, image loc#s       : whgb_c4_classums_sort_errar
Source of sort values          : header
Criteria for sort              : angrec_error
Sort up or down                : up
How many of sorted images      :                    you select
```

4. You should always check the selected/sorted images "by eye" with the command DISPLAY as those "best" class averages are not necessarily the best ones. Also make sure that you did not miss a typical view.

19. Refined 3-D Reconstruction

1. Use these sorted/selected class averages (whgb_c4_classums_sort_errar) to calculate a refined 3-D reconstruction:

```
IMAGIC-COMMAND: th-reconst

Mode of 4D operation           : all_in_one
Point-group symmetry           : d6
Use default 3D rec. options    : yes
Input 2D (classum) image       : whgb_c4_classums_sort_errar
Source of Euler angles         : angrec_header
Update output header           : yes
Output file for 3D volume      : whgb_c4_3d_2
Output file re-projections     : whgb_c4_rep_2
Output file error projections  : whgb_c4_err_2
Spherically mask the reconst.  : yes
Radius of the mask             : 0.55
Hamming window factor          : 0.9            larger value
Object size                    : 0.75
Create a normalized 3D volume  : yes
Give new sigma                 : 1              helpful in Chimera
```

The command re-projects the 3-D volume in the same direction as the input class averages. It then calculates an error based on the correlation between the re-projections and the input class averages. After 3-D volume is calculated THREED-RECONSTRUCTION displays an error list:

```
          Error in input images (sorted list)
         ====================================

         Loc     class    alpha    beta     gamma     error
          #        #                                  in 3-D

 1:>     nn       cc      aa.aa    bb.bb    gg.gg    x.xx %
 2:>     nn       cc      aa.aa    bb.bb    gg.gg    x.xx %

                         ...

 Average error in the set of input images:  e.ee %
```

2. If one re-projection has a much higher error than the rest, i.e. if the error suddenly in the sorted list jumps then you could exclude all the following "bad" class average using EXCLUDE-IMAGE.

   But, as before, DISPLAY and check "by eye" and make sure not to remove needed "typical" views.

3. Calculate a new THREED-RECONSTRUCTION without using the excluded bad class-averages.

4. If necessary, use THREED-FILTER (chapter 14).

5. Use THREED-AUTO-MASK to automatic mask the 3-D volume (chapter 17).

6. As before visualize the 3-D volume with commands DISPLAY, MOVIE, THREED-SURFACE or the program **Chimera** (chapter 16).

YOUR NOTES:

20. Iterate Angular Reconstitution and 3-D Reconstruction

1. Call the command THREED-FORWARD to forward project the last (filtered and masked) 3-D volume to create a new anchor set.

2. Use this anchor set to run ANGULAR-RECONSTITUTION with option ANCHORSET to refine the Euler angles assignment and to calculate a new 3-D reconstruction with the command THREED-RECONSTRUCTION. See Figure 14.

3. Repeat these steps until the Euler angles are stable.

YOUR FILE NAMES:

YOUR NOTES:

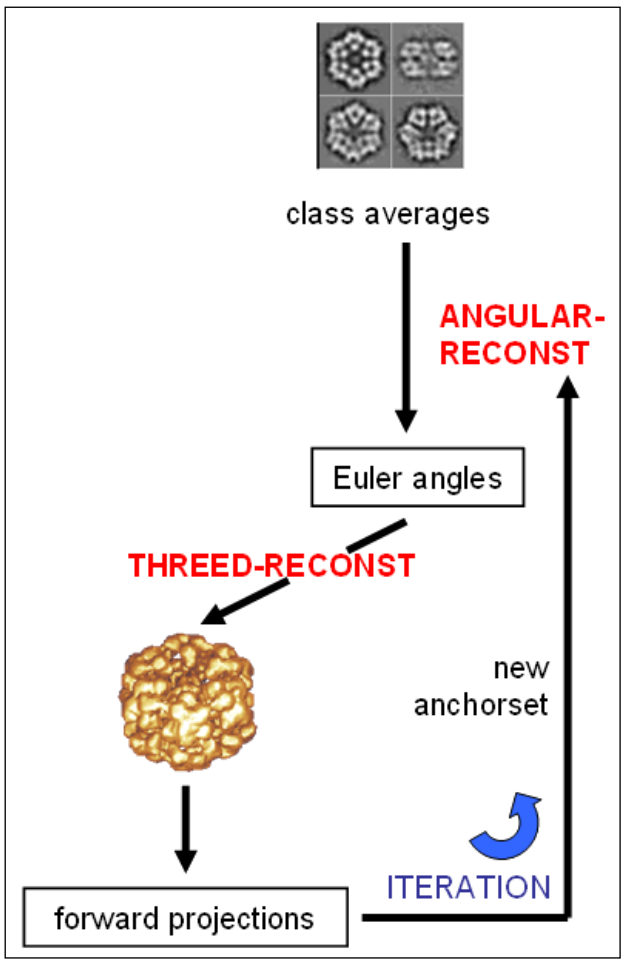


Fig. 14: Iteration of Angular Reconstitution and 3-D Reconstruction

NOTE:

If there is time you can continue with a multi-reference alignment (next chapter).

But probably there is no time any more during the practical. So enjoy your current 3-D volume ☺.

Nevertheless you will find a number of refinement steps and iterations in the following chapters.

21. Multi-Reference-Alignment (MRA)

The input file for the multi-reference alignment is your particle stack (the first time: whgb_c4_part_filt, later the aligned file whgb_c4_part_ali_X). The original (pre-treated) file does not change (whgb_c4_part_filt). However, you need to make sure that the alignment header values XSHIFT, YSHIFT, EXSHIFT, EYSHIFT in the picked particle stack are equal to 0. They might be different from 0, if, for example, a movie alignment had been performed.

IMPORTANT NOTE:

For zeroing the header values use the command HEADER with options WRITE, WIPE and ALIGN.

Do NOT use the commands PREPARE-IMAGES, NORM-VARIANCE and PREPARE-MRA any more because the references come from one single 3-D and are already centred, i.e. perfectly "3-D aligned".

1. Create references for a subsequent MULTI-RERERENCE-ALIGNMENT:

```
IMAGIC-COMMAND: threed-forward

Option used                              : FORWARD
Input 3D image file                      : whgb_c4_3d_X_masked
Output file for forward projections      : whgb_c4_mraref_X
Threshold 3D density value               : -9999
Use default interpolation mode           : yes
Choose Euler angles option               : asym_triangle
Point-group symmetry to be used          : d6
Option to chose Euler angles             : equidist
Minimum angular distance                 : 3.0
Also generate mirror projections         : yes              YES !!!
Option for Euler angle alpha             : zero             ZERO !!!
Wanted angular increment                 : 7.5
Full output of all parameters            : no
```

NOTE:

In contrast to the creation of anchor sets we here create a higher number of references and also use the mirror versions. Make sure the Euler angles Alpha are zeroed.

2. Run the MULTI-RERERENCE-ALIGNMENT. As was explained in the lecture, in order to avoid interpolation artefacts, **IMAGIC** keeps the original filtered data and continually calculates the equivalent rotation necessary to reach the latest alignment. Use NO location numbers.

```
IMAGIC-COMMAND: m-r-a

MRA options:                          : fresh
4D options:                           : all_references
Methods available                     : align
Alignment modes available             : both
Start options available               : rotation_first
Correlation functions available       : ccf
Input file, loc#s                     : whgb_c4_part_filt
                                                    or later:
                                        whgb_c4_part_mra_(X-1)
Output file, loc#s                    : whgb_c4_part_mra_X
Original (pretreated) file, loc#s     : whgb_c4_part_filt
Reference file, loc#s                 : whgb_c4_mraref_X
Option to filter the reference(s)     :                your choice
                                                       give ?  and
                                                       and read help
Max shift (compared to originals)     : 0.1
Max shift (during this alignment)     : 0.1
Min,max rot. angle (originals)        : -180,180
Min,max rot. angle (alignment)        : -180,180
Precision for rotational alignment    : high
Min,max radius for rot alignment      : 0,0.7
Number of alignment iterations        : 3
Full output of all parameters         : yes
```

3. After the alignment has completed, you should run a new round of MSA classification (MSA-RUN / MSA-CLASSIFY / MSA-SUM) on the aligned images (whgb_c4_part_mra_X) to get new class averages (whgb_c4_classums_X). The input images are aligned, so you can choose a smaller number of classes.

4. DISPLAY the new class averages and SELECT the good class averages (creating the output PLT file whgb_c4_classums_X_best.plt).

5. Extract these new "best" class averages using command EXTRACT-IMAGES with option PLT_FILE. Input files are the new class averages (whgb_c4_classums_X) and the PLT file whgb_c4_classums_X_best.plt. Output will be whgb_c4_classums_X_best.

BEST CLASS AVERAGES:

Of course, these iterations are usually no more part of the practical.

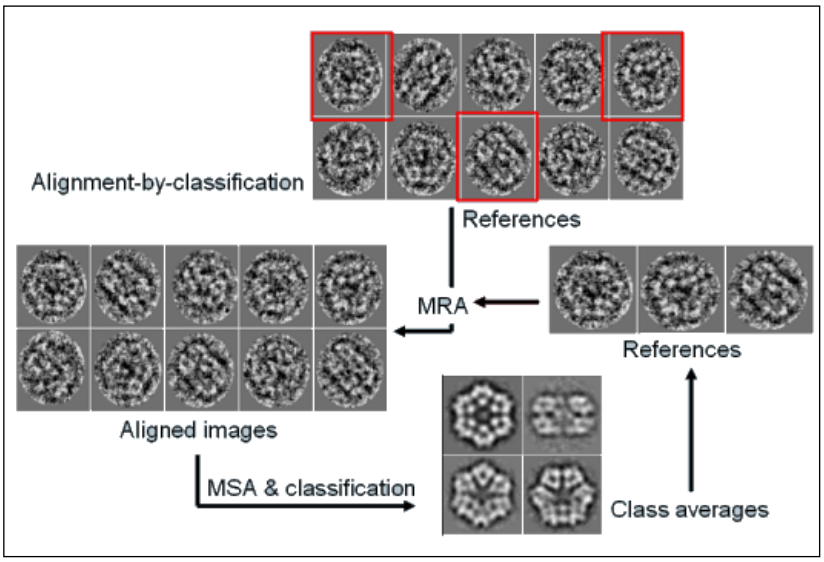


Fig. 15: Alignment by Classification and Multi-Reference Alignment (2D)

22. Iteration Cycle(s) of Multi-Reference Alignment and MSA Classification

As mentioned in the lectures you can iterate this MRA / MSA classification cycle until you feel your class averages are of sufficient quality.

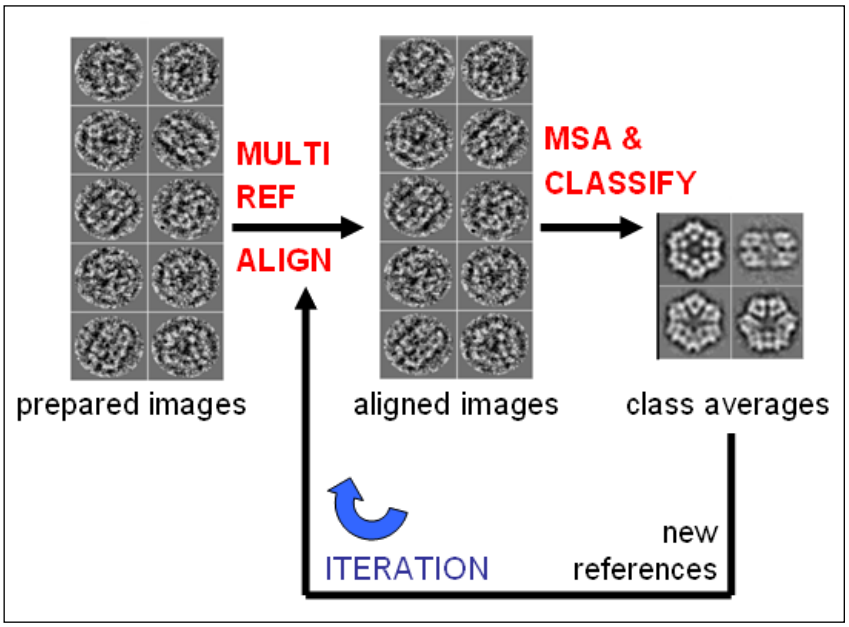


Fig. 16: Iteration of Multi-Reference Alignment and MSA Classification

Again: These iterations are usually no more part of the practical.

YOUR FILE NAMES:

YOUR NOTES:

23. Iterate MRA/MSA Classification and Angular Reconstitution/3-D Reconstruction

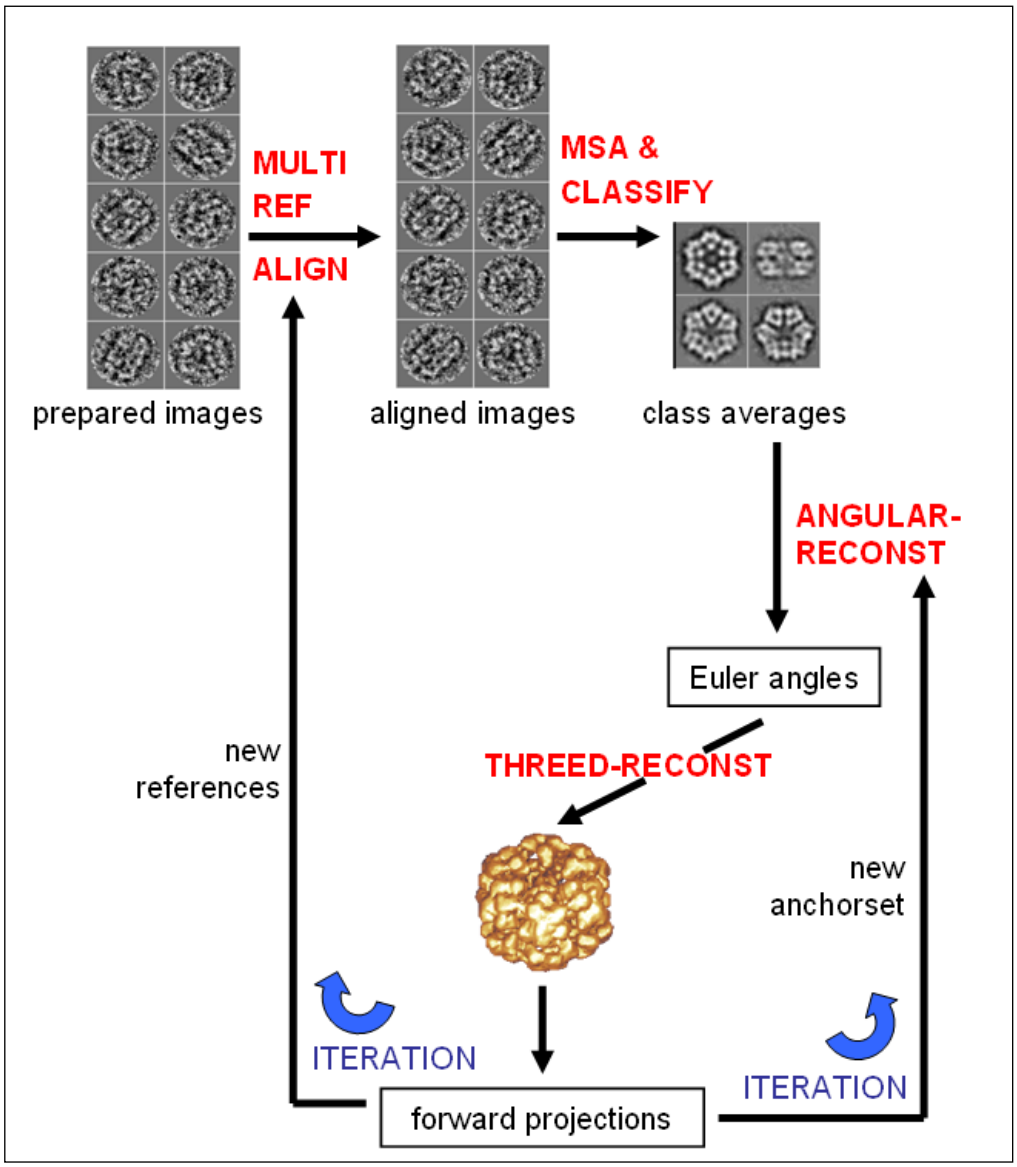


Fig. 17: Iterate MRA / MSA Classification and Angular Reconstitution. / 3D Reconstruction

This refinement loop is repeated many times until you reach your desired resolution or convergence. This iterative refinement is the most time consuming process. As the quality of your reconstruction increases you can use a finer angular increment for forward projecting your M-R-A references.

Also one can go back to particle picking now using the THREED-FORWARD images as (better) references for a CORRELATION picking of particles (PICK-IMAGES) and repeat the MRA / MSA classification and angular reconstitution (anchor set) / 3-D reconstruction iterations.

But make sure that you (strongly) low-pass filter your references before picking to avoid reference bias/over-fitting.

Of course, this iteration is not part of this practical.

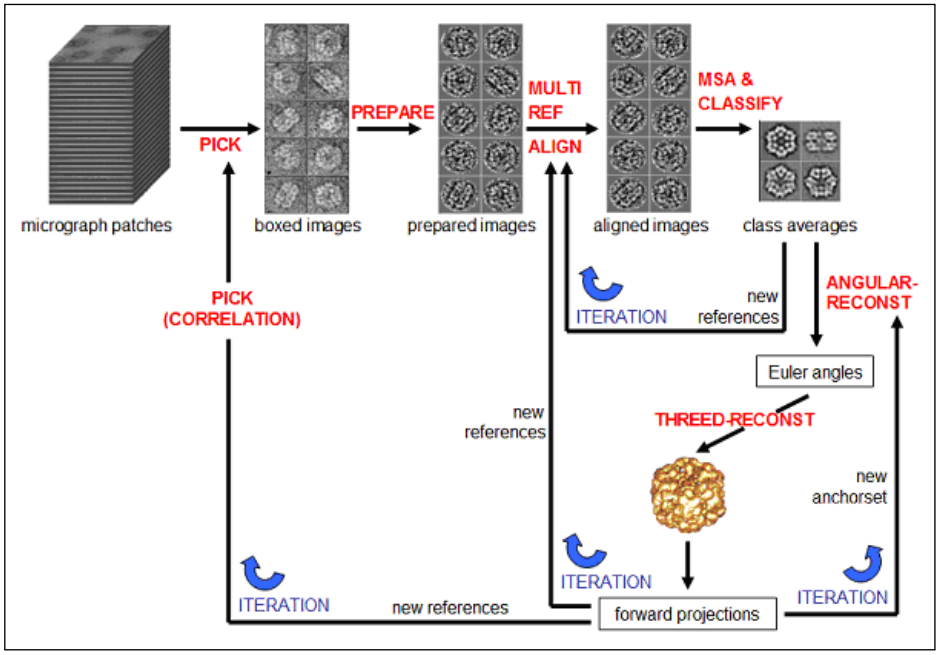


Fig. 18: Re-do Particle Picking / Iterations

You can also pick from the un-coarsened data set if the limit of the current sampling is reached.

Again, this is not part of this practical.

24. Fourier Shell Correlation (Estimate the Resolution)

In nearly all publications the Fourier shell correlation (FSC) is used to estimate the resolution of a 3-D reconstruction.

Remember, that the FSC is not really a resolution measure but a criterion to compare the similarity of two 3-D reconstructions. If it is used to estimate the resolution of a 3D reconstruction one has to make sure that the two 3D subsets do not contain artificial similarities or the same systematic errors. The best approach would be to calculate two 3-D volumes completely independently.

You cannot do this in this practical. To get an idea how the FSC can be calculated and interpreted you will calculate 3-D volumes from two-subsets, which we assume to be the "two independent data sets".

1. The command THREED-RECONSTRUCTION (option FOURIER_SHELL_CORRELATION) can create the needed two 3-D reconstructions needed to calculate Fourier Shell Correlation.

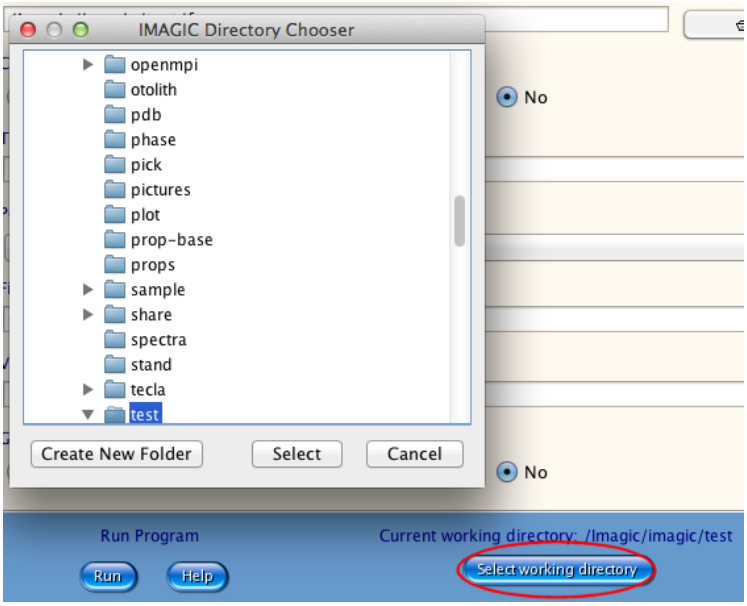
IMPORTANT NOTE:

Do NOT use any mask!!

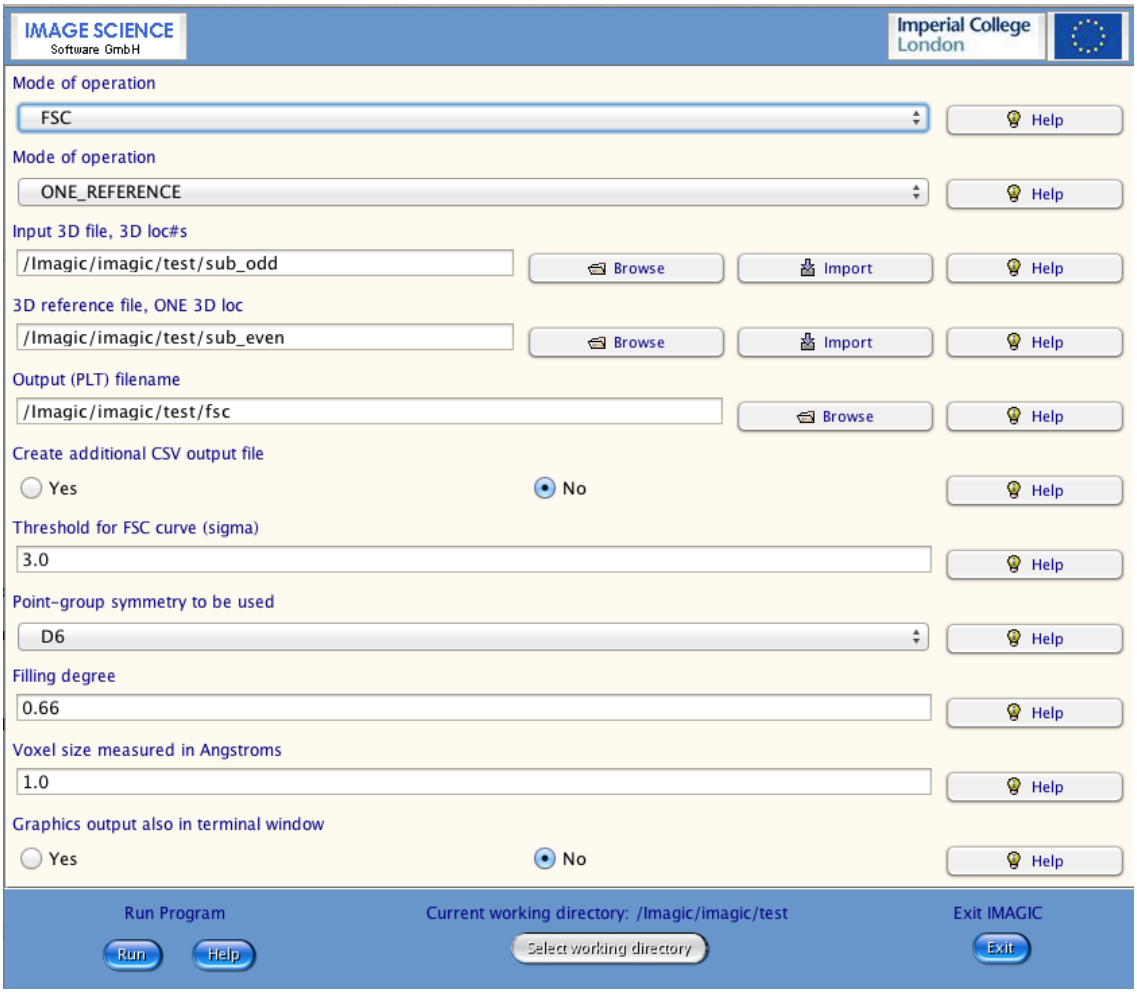
```
IMAGIC-COMMAND: th-reconst

Mode of 4D operation                    : fourier_shell
Minimal number of images per 3D         : 0.7
Point-group symmetry                    : d6
Use default 3D reconstruction options   : yes
Input 2D (classum) images               :          your last aligned
                                                   class averages
Source of Euler angles                  : angrec_header
Update output header                    : no
Output file for 3D reconstruction       : whgb_c4_3d_fsc
Output file for re-projections          : none
...
Also create a normalized 3D volume      : no
```

2. Use the FSC program (www.ImageScience.de/fsc), which is the GUI version of the command FOURIER-SHELL-CORRELATION.

   Change to your working directiory (button "Change working directory")



   and answer all questions:



Press the "Run" button to calculate the FSC. In the results window use the buttons "Next" and "Previous" to get the FSC/3-sigma and the FSC/½-bit curve.

The crossing of the Fourier Shell Correlation and the ½-bit information threshold curve (in output plot #2) expresses where you have already collected a sufficient amount of data in the final 3-D reconstruction to allow a direct structural interpretation at that resolution level. The ½-bit curve is calibrated to approximately yield resolution values comparable to resolution values in use in X-ray crystallography (FOM).

The crossing of the Fourier Shell Correlation and the (modified) 3-sigma curve (in output plot #1) indicates where the FSC systematically emerges above the expected random correlations of the background noise. This criterion indicates at which spatial frequency you are systematically gaining information significantly above the random noise level. When you continue collecting information by adding more data of the same quality to the data set you would certainly improve the data set up to - and maybe even somewhat beyond - this point.

3. If you do not want (or cannot use) the FSC standalone program use FOURIER-SHELL-CORRELATION and PLOT to visualise the FSC curves:

```
IMAGIC-COMMAND: f-s-c

Mode of operation                          : sequential
Input file, 3D loc#s                       : whgb_c4_3d_fsc
Output FSC (PLT) filename                  : whgb_c4_fsc
Create additional CSV output file          : no
Threshold for FSC curve (sigma)            : 3
Pointgroup symmetry to be used             : d6
Filling degree                             : 0.66
Voxel size measured in Angstroms           : 4.44
Graphics output also in terminal window: no


IMAGIC-COMMAND: plot


Input file (image or plot)                 : whgb_c4_fsc
...
```

Note that when using PLOT the horizontal axis shows 1/Resolution. Also do not miss to consider the exponent! Estimate your resolution based on where the ½-bit curve crosses the FSC curve.

NOTE:

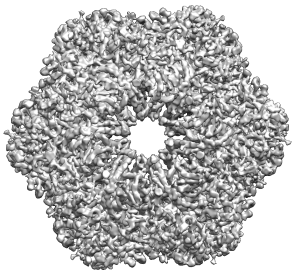
The FSC is a measure to compare the similarity of two 3-D data sets. If it is used to estimate the resolution of a 3-D reconstruction you have to make sure that the two 3-D subsets do not contain artificial similarities (like masks, for example).

It is good practise not to interpret resolution curves, which are too close to the high end of the resolution curve (the right hand side of the FSC curve). In other words: you should never claim any resolution level beyond 2/3rd of the Nyquist frequency.

If the sampling size is 5.2 Angstrom per pixel/voxel then the attainable resolution is about 15.6 Angstrom rather then the theoretical Nyquist frequency of 10.4 Angstrom. If the resolution is better then 3x the sampling size your data set is under-sampled and you should re-scan your micrographs with a higher resolution and re-do the image analysis.

Whilst the ½-bit curve provides a single figure for your resolution it is important to always take into account the curve as a whole when judging the quality of the reconstruction.

The best resolution measure is still the resolution of the biological details, which you can see in your 3-D reconstruction.



25. Advanced Topics

Copy the files whgb_frames_70 that you can find in the data directory whgb_dataset_2016/02_whgb micrographs_imagic of the Brazil School server to your working directory. The file contains 10 raw micrograph movies (70 movie frames).

25.1. *A posteriori* Camera Correction

The *a posteriori* camera correction procedure allows improving the quality of the micrographs images based on the statistics of the full dataset. In many cases, it also helps to improve the movie alignments. We perform the a posteriori camera correction on each frame image using the total average image and the corresponding sigma image of all the frames in the dataset.

NOTE:

For the practical done until now the camera correction and the movie alignment was already performed for you.

Here you can try to run the commands using the smaller dataset, which you just copied from the Brazil School server.

1. First run command SURVEY:

```
IMAGIC-COMMAND: survey

Mode of survey                          : 2d_local
Mode of output                          : update_header
Input file                              : whgb_frames_70
```

2. Then calculate the statitics histogram of the average densities

```
IMAGIC-COMMAND: headers

Specify option                          : histogram
Histograms from which images            : all
Histogram option                        : average
Number of bins for histogram            : 56
Width of histogram                      : 79
Input file                              : whgb_frames_70
```

   and the one showing sigma of the densities values:

```
IMAGIC-COMMAND: headers

Specify option                          : histogram
Histograms from which images            : all
Histogram option                        : sigma
Number of bins for histogram            : 56
Width of histogram                      : 79
Input file                              : whgb_frames_70
```

3. If your dataset is big enough, you get two histograms like the follwowing ones, which were calculated from the full dataset of 500 micrographs:

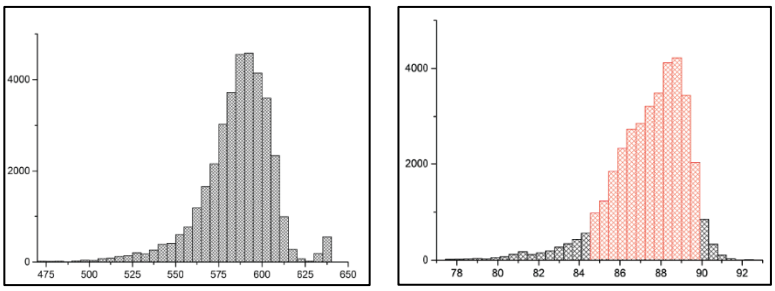


Fig. 19: Histograms of the average values (left) and the sigma values (right)

Unfortunately the small dataset in file whgb_frames_70 does not contain enough images to get such nice distributions.

NOTE:

It is always important to exclude micrograph images containing too high-contrast and features like grid bars, junk, ice crystals, etc.

Those micrograph images often correspond to the extreme values of the histogram of sigma and can be excluded automatically based on the histograms. We selected the "good" micrograph images using the red part of the histogram. Only micrograph images corresponding to this red area will be used for the *a posteriori* camera correction.

4. Write up the GOOD micrograph location number

GOOD MICROGRAPHS:

First,last class location:

5. Set the "good" micrograph images "active" so that only these images are used for the camera correction:

```
IMAGIC-COMMAND: header

Specify option                  : set
Please specify option           : inactive
Input header file               : whgb_frames_70


IMAGIC-COMMAND: header

Specify option                  : set
Please specify option           : active
Input header file               : whgb_frames_70,loc#1,loc#2
                                                  your selection
```

6. To check the performance of each camera pixel calculate the average and sigma image of the "active" micrograph images with command CAMERA-NORM. We have already done the calculation for you so you can ciopy the results from the Brazil School server.

Copy the files whgb_frames_average and whgb_frames_sigma that you can find in the data directory whgb_dataset_2016/02_and micrographs_imagic/camera_norm of the Brazil School server to your working directory.

These files were created with the command CAMERA-NORM:

```
IMAGIC-COMMAND: camera

Mode of operation               : camera_statitics
Use which statistics            : gaussian
Input images, no image loc#s    : whgb_frames_70
Output average file             : whgb_frames_average
Output sigma file               : whgb_frames_sigma
Based on active images          : yes
```

7. Cut the two images into patches (command CUT-IMAGE) to be able to visually inspect the camera performance at full resolution in command DISPLAY.
8.

```
IMAGIC-COMMAND: cut-image

Mode of operation                : checkers
Input images, no image loc#s     : whgb_frames_average
Output file, image loc#s         : whgb_average_512
Output image dimensions          : 512,512
Overlap between fields           : 0

IMAGIC-COMMAND: cut-image

Mode of operation                : checkers
Input images, no image loc#s     : whgb_frames_sigma
Output file, image loc#s         : whgb_sigma_512
Output image dimensions          : 512,512
Overlap between fields           : 0
```

9. DISPLAY the images whgb_average_512 images whgb_sigma_512 and inspect the camera performance.

10. To perform the camera normalisation/correction call the command CAMERA-NORM now using the option CORRECT:

```
IMAGIC-COMMAND: camera

Mode of operation                : correct
Use which statistics             : gaussian
Input images, no image loc#s     : whgb_frames_70
Output file ,image loc#s         : whgb_frames_70_cnorm
Input average file               : whgb_frames_average
Input sigma file                 : whgb_frames_sigma
Based on active images           : yes
```

NOTE:

In "real science" after having set the "good" micrographs "active" you can run the camera normalisation/correction directly using the option MEASURE_AND_CORRECT in command CAMERA-NORM.

25.2. Movie Alignment

1. To speed up the movie alignment, coarsen your micrograph stack by a factor of 4:

```
IMAGIC-COMMAND: coarsen-image

Input file                    : whgb_frames_70_cnorm
Input file                    : whgb_frames_70_cnorm_c4
Summing parameter             : 4
```

2. Run the movie alignment using the command ALIGN-MOVIE:

```
IMAGIC-COMMAND: ali-mov

What is to be aligned         : MOVIE_FRAMES
How many frames per movie     : 7
Overall direct alignment mode : translation
Correlation function          : ccf
Maximal shift                 : 2
Input image (movie) file      : whgb_frames_70_cnorm_c4
Output image file             : whgb_frames_70_cnorm_c4_ali
Using which reference         : create_reference
Summing option for 1st ref.   : total_average
Give this reference a number  : 0
```

```
Mask reference before align    : no_mask
Filter the reference           : bandpass
Low frequeny cut-off           : 0.05
High frequeny cut-off          : 0.2
Refine reference iterativly    : yes
Maximal reference refinements  : 6
Summing option wanted          : total_sum
Over correction factor         : 0.9
Threshold to stop refinement   : 0.02
Store new referenves           : no
Full output                    : yes
```

3. Apply the determined shifts to the uncoarsened micrographs using the command EQUIVALENT-MOVE:

```
IMAGIC-COMMAND: equi-move

Mode of operation              : EQUIVALENT_MOVE
Input image file               : whgb_frames_70_cnorm
Input header file              : whgb_frames_70_cnorm_c4_ali
Output image file              : whgb_frames_70_cnorm_ali
Max. shift (0: no check)       : 0
Correct for diff. image size   : yes
```

4. You can check the quality of the movie alignments by comparing the P-spectra with DISPLAY before and after the alignments. Call command MOVIE-SPECTRA to create the P-spectra:

```
IMAGIC-COMMAND: movie-spec

Use which spectra              : p_spectrum
Input image file               : whgb_frames_70_cnorm
Output image file              : whgb_frames_70_cnorm_ps
Howmany frames per movie       : 7
```

```
Mask input images              : no
Coarsen the spectrum images    : yes
Coarsening parameter           : 8
Band-pass filter the spectra   : yes
Low frequeny cut-off           : 0.02
Low frequeny transmisson       : 0.02
High frequeny cut-off          : 0.7

IMAGIC-COMMAND: movie-spec

Use which spectra              : p_spectrum
Input image file               : whgb_frames_70_cnorm_ali
Output image file              : whgb_frames_70_cnorm_ali_ps
Howmany frames per movie       : 7
Mask input images              : no
Coarsen the spectrum images    : yes
Coarsening parameter           : 8
Band-pass filter the spectra   : yes
Low frequeny cut-off           : 0.02
Low frequeny transmisson       : 0.02
High frequeny cut-off          : 0.7
```

5. Compare the P-spectra in DISPLAY.

6. You can sum the (aligned) frames in each movie with the command SUM-MOVIE-FRAMES:

```
IMAGIC-COMMAND: sum-movie

Mode of summing                : MOVIE_SUM
Input image file               : whgb_frames_70_cnorm_ali
Output image file              : whgb_frames_70_msum
Number of frames per movie     : 7
Movie frames to sum (0:all)    : 2,5                    your choice
```

SOME WEBSITES:

www.single-particles.org/school

www.single-particles.org/school/2016

www.lnnano.cnpem.br/laboratories/lme

grigorieff.org

www.ImageScience.de

www.single-particles.org

3dem.ucsd.edu

www.researchgate.net/profile/Marin_Heel

www.bcm.edu/research/labs/wah-chiu

www.researchgate.net/profile/Rodrigo_Portugal2

www.igbmc.fr/Klaholz

www.icr.ac.uk/our-research/researchers-and-teams/dr-edward-morris

www.biophys.mpg.de/en/moeller.html

grigoriefflab.janelia.org/arohou

www.researchgate.net/profile/Pavel_Afanasyev

www.researchgate.net/profile/Michael_Saur

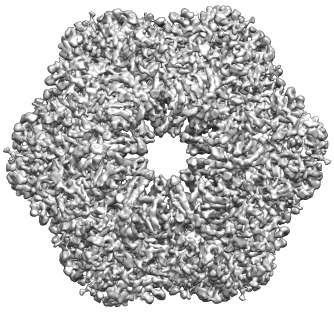
www.psi.ch/lbr/emiliya-poghosyan

www.researchgate.net/profile/Michael_Schatz

ERROR HINTS:

We tried to find and correct all errors and typos before, during and after the Brazil School. If you still find some mistakes please send your error hints to michael@ImageScience.de so that we can improve this tutorial. Thank you very much.

YOUR NOTES:



CONTENT