TheNogalesLab
a Cryo Electron Microscopy Research Lab
HHMI Berkeley UNIVERSITY OF CALIFORNIA

Orthogonal tilt reconstruction (OTR)

Original reference: Leschziner & Nogales, 2006 [http://www.ncbi.nlm.nih.gov/pubmed/16431136]

Methods: Leschziner, 2010 [http://www.ncbi.nlm.nih.gov/pubmed/20888964]

Comparison with RCT: Chandramouli et al. 2011 [http://www.ncbi.nlm.nih.gov/pubmed/21536134]

Picking particles

There are two choices: APPION-based picking [http://www.ncbi.nlm.nih.gov/pubmed?term=tiltpicker] and manual picking. Since Web (SPIDER) does not work on our workstations we have been using XMIPP:

(1.) Place micrographs into working directories

Symbolically link 'untilted' micrographs to folder '00' and 'tilted' micrographs to '01:'

```
ln -s /leginon/michael/10sep12a/rawdata/*en_00.mrc 00/
ln -s /leginon/michael/10sep12a/rawdata/*en_01.mrc 01/
```

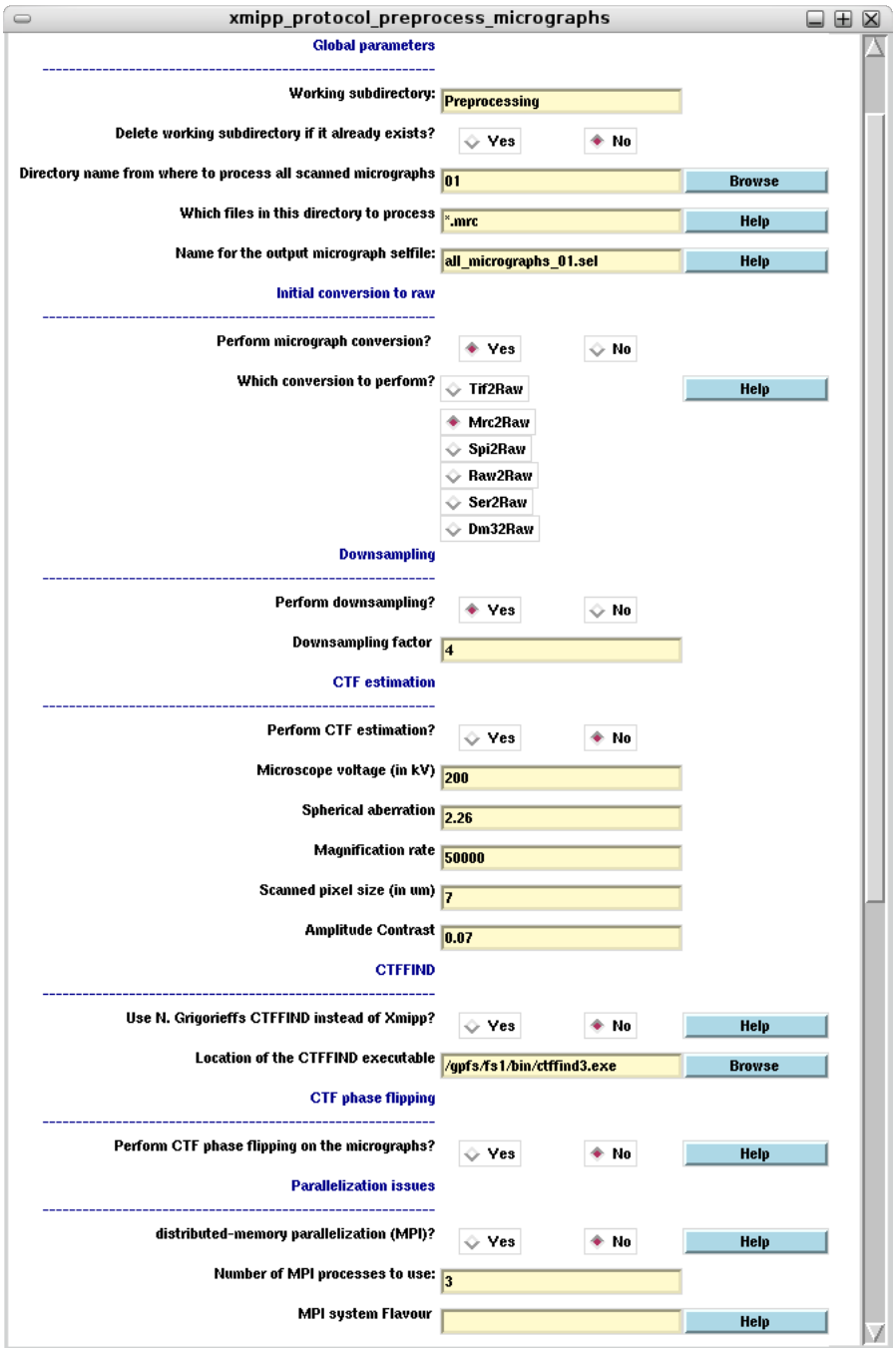
(2.) Import micrographs into XMIPP format

In the folder *above* 00/ and 01/, start xmipp_protocols:

```
[michael@angula manualOTR ]$ xmipp_protocols
 + Xmipp_protocols was never executed before in this directory.
 + Make sure you run xmipp_protocols only in your project directory.
 + You are in directory: /local/michael/TFIID_IIA_SCP1/Negative_stain/OTR/Preprocessing
 + Do you want to setup Xmipp protocols here? [y/n]:y
```

Click on **Preprocess Micrographs**

After inputing & selecting the options shown below, click **Save & Execute**:



Perform the same command for '00' with the **Preprocess Micrographs** GUI.

(3.) Create select file for tilt pairs

Copy create_tiltSelFiles.py from:

```
~michael/BATCHLIB/OTR_scripts/TiltPickingXMIPP/
```

Within the *PreProcessing* directory created by xmipp_protocols, run:

```
./create_tiltSelFiles.py > all_tilts.sel
```

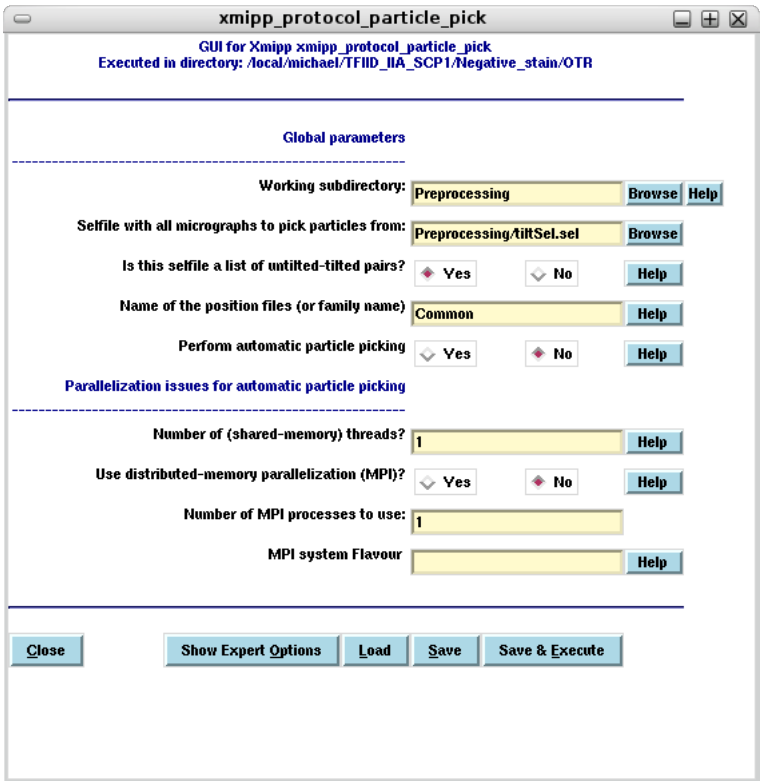
This outputs the names of tilt mates in the proper XMIPP format:

```
<tilt_00.raw>   <tilt_01.raw>   1
```

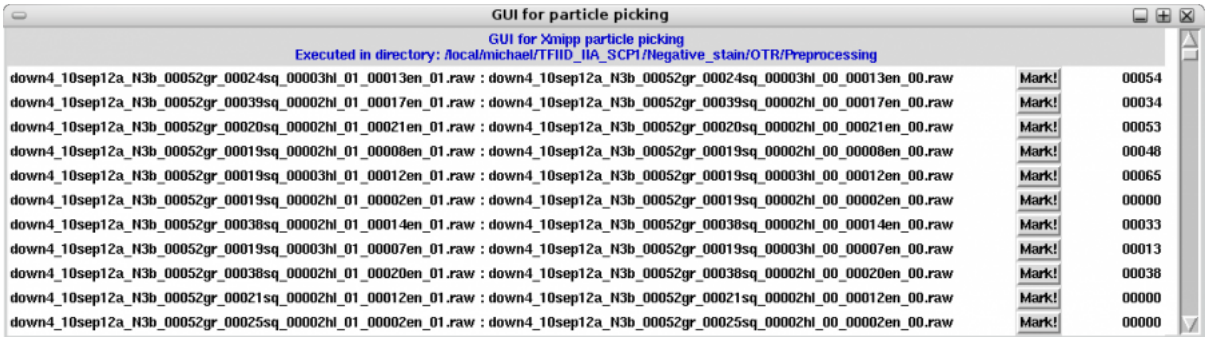
- 1 - denotes that the tilt mates are 'active' for picking

(4.) Picking tilt pairs

Click on **Particle Selection** from the main xmipp_protocols GUI and the following window opens:

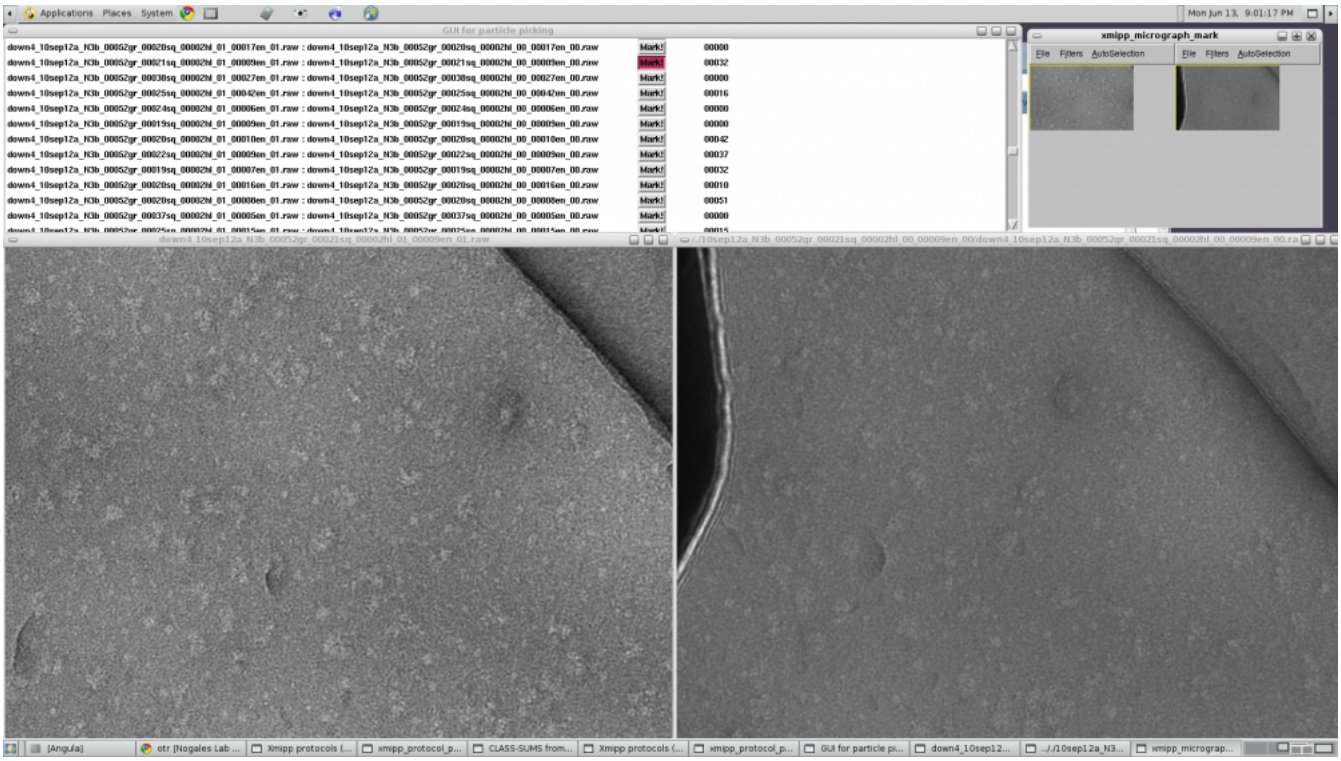


Click **Save & Execute** to get to xmipp_mark [http://xmipp.cnb.csic.es/twiki/bin/view/Xmipp/Mark].



This first GUI organizes all of the tilt pairs for picking, allowing you save your picking progress and come back anytime to finish picking.

Clicking on **Mark!** will open tilted micrographs for picking:



xmipp_mark will slowly learn the angle between the micrographs, getting better at picking tilt mates as you pick more particles.

(5.) Saving particle coordinates

When you are finished picking particles, be sure to save the coordinates & angles for **each micrograph** by clicking **File** -> **Save Coords** and **File** -> **Save angles**.

Extracting particles

All scripts in this section can be found:

```
~michael/BATCHLIB/OTR_scripts/extract/
```

(1.) Convert XMIPP coordinates (.pos) to BOXER coordinates (.box)

Copy all files (.Common.pos) within the Preprocessing folder to a new folder, along with *raw2boxdb2.pl* and *raw2boxdb_all.csh*. After editing the filenames within *raw2boxdb_all.csh*, run:

```
./raw2boxdb_all.csh
```

This will output .box filenames that will correspond directly to micrograph names within leginon/, allowing you to import the box files into APPION or use batcboxer for making the stack.

Extract particles and calculate tilt angles

Symbolically link all of your micrographs to your working directory:

```
ln -s /leginon3/michael/12feb01b/rawdata/*en_??.mrc .
```

To simultaneously extract particles while also generating a per-particle CTF & tilt angle file (from CTFTILT [http://www.ncbi.nlm.nih.gov/pubmed/12781660]), copy *ctftilt.py* to your working directory and edit the beginning of the file.

If you'd like to perform **Ace2 Wiener Filtering** a la APPION, run the script *ctftilt_ace2_weiner.py* instead of *ctftilt.py*.

```
#Output filenames
paramOUT1 = 'parameter_00.par'  #param filename for untilted particles
paramOUT2 = 'parameter_01.par'  #param filename for tilted particles
stack1 = 'stack00'              #stack name for untilted particles
stack2 = 'stack01'              #stack name for tilted particles

#Inputs
shrink = 4                      #Binning factor for final particle stack
new = 64                        #Box size for binned particles
scale = 4                       #Binning factor used for micrograph from which the particles were picked

#CTFTILT inputs
parm3 = "2.2,120.0,0.2,99993,15,2\n" # !CS[mm],HT[kV],AmpCnst,XMAG,DStep[um]
parm4 = "128,400.0,8.0,2000.0,30000.0,500.0,30,5\n" #!Box,ResMin[A],ResMax[A],dFMin[A],dFMax[A],FStep
```

Then run:

```
./ctftilt.py &
```

Be patient, this will need ~2 minutes per micrograph (for CTFTILT) as it simultaneously determines the CTF of your images & makes your particle stacks. This will make the particle stacks and parameter files as specified above. The format of the parameter file is:

```
 DEF1            DEF2           ASTIG         TILT ANGLE

5879.46         6073.91         34.13           -3.07
5879.46         6073.91         34.13           -3.07
5879.46         6073.91         34.13           -3.07
5879.46         6073.91         34.13           -3.07
5879.46         6073.91         34.13           -3.07
```

Inspecting each particle stack, check to see if there are any edges that have been boxed out. If there are, write down the number of the particle in a text file using the EMAN numbering scheme (i = i - 1). Chances are that you'll be inspecting the particles using EMAN.

To exclude the bad particles:

```
proc2d stack00.hed stack00_sel.hed exclude=bad.txt
```

To exclude corresponding lines from parameter files:

```
./select_params.py
Usage: select_params.py -p <parameter> -b <exclude>

Options:
  -h, --help  show this help message and exit
  -p FILE     Parameter file with per-particle CTF information from ctftilt.py
  -l FILE     List of particles (numbered in EMAN format)
  -o FILE     Output filename
  -b          Flag if particle list is for particles to EXCLUDE
  -g          Flag if particle list is for particles to INCLUDE
  -d          debug
```

2D alignment of particles

Using the 'good' tilted particles or merged stack of both tilts, perform 2D reference-free alignment with autoAlign or MSA-MRA.

Calculate class volumes

(1.) Select iteration of 2D reference-free alignment from autoAlign or MSA-MRA that looks good

Output particle membership info and rotation/shifts. Rotation & shifts should come from **previous** iteration:

```
IMAGIC-COMMAND : header

  ** HEADERS (vs. 8-Oct-2010) welcomes you **


Options available:
  READ/LOOK  WRITE/SET  HOW_MANY   PLT_OUT    TAKE_OVER
  SORT       COMPARE    HISTOGRAM  MEANINGS
Please specify option [PLT]                         :

Read options available:
  ALIGNMENT               MOVE(ROTATION&SHIFT)
  MSA/CLASSIFICATION      THREED_RECONSTRUCTION
  ANGULAR_RECONSTITUTION  PROJECTION_MATCHING
  EULER_ANGLES            INDEX/LABEL
Please specify option [ALIGN]                       : MOVE
Input (header) file, loc#s [current_mra]            :
PLT output file to store values [rotAngles.plt] :

IMAGIC-COMMAND : header

  ** HEADERS (vs. 8-Oct-2010) welcomes you **


Options available:
  READ/LOOK  WRITE/SET  HOW_MANY   PLT_OUT    TAKE_OVER
  SORT       COMPARE    HISTOGRAM  MEANINGS
Please specify option [PLT]                         : plt

Read options available:
  ALIGNMENT               MOVE(ROTATION&SHIFT)
  MSA/CLASSIFICATION      THREED_RECONSTRUCTION
  ANGULAR_RECONSTITUTION  PROJECTION_MATCHING
  EULER_ANGLES            INDEX/LABEL
Please specify option [ALIGN]                       : MSA
Input (header) file, loc#s [current_mra]            :
PLT output file to store values [membership.plt] :
```

(2.) Convert each IMAGIC output into SPIDER readable format

```
cat membership.plt | gawk '{printf ("%5d %1d  %-6.2f \n",NR,1,$0)}' >> membership.spi

cat rotAngles.plt | gawk '{printf ("%5d   %1d   %-6.6f     %-6.6f     %-6.6f \n",NR,3,$2,$3,$4)}' >> rotAngles.spi
```

(3.) Generate angular file for each particle

Run the following for MSA-MRA: (or run shiftRotImagicParams_autoAlign.spi if you used autoAlign)

```
[michael@angula reconstruction]$ spider spi @shiftRotImagicParams

 \__`O O'__/        SPIDER  --  COPYRIGHT
 ,__xXXXx___        HEALTH RESEARCH INC., ALBANY, NY.
   __xXXXx__
 /  /xxx\  \        VERSION:  UNIX  18.10 ISSUED: 03/23/2010
   /     \          DATE:     14-JUN-2011    AT  16:58:34



 Results file: results.spi.0
 Running: spider
```

```
 .?Input the rotation & shift file from IMAGIC?: rotAngles
 .?Input the class membership file from IMAGIC?: membership
 .?Input particle-micrograph list?: micrographs_list
 .?Input basename from ctfTilt_Parser.py?: ctfTiltparsed
```

Output

The output ctfTiltparsed_angular.spi will have the euler angles calculated by taking into account the in-plane rotations from MRA with the tilt angles between micrographs (see Leschziner, 2010 [http://www.ncbi.nlm.nih.gov/pubmed/20888964] for equations to calculate euler angles):

```
    1 3   3.7200      -94.700       165.19
    2 3   3.7200      -94.700       96.541
    3 3   3.7200      -94.700       8.4850
    4 3   3.7200      -94.700       157.41
    5 3   3.7200      -94.700       223.97
    6 3   3.7200      -94.700       153.73
    7 3   3.7200      -94.700       87.393
    8 3   3.7200      -94.700       204.88
    9 3   3.7200      -94.700       291.60
```

(5.) Calculate class volumes

By combining the per-particle euler angles with select files of particles in each classum, you can calculate the 3D volumes of each class:

```
[michael@angula reconstruction2]$ spider spi @bp3f_filter_OTR_volumes

 \__`O O'__/        SPIDER  --  COPYRIGHT
 ,__xXXXx___        HEALTH RESEARCH INC., ALBANY, NY.
   __xXXXx__
 /   /xxx\   \      VERSION:  UNIX  18.10 ISSUED: 03/23/2010
   /     \          DATE:     14-JUN-2011    AT  17:09:18



 Results file: results.spi.3
 Running: spider
 .?Input basename of select files?: sel_class
 .?Input tilted stack?: OTR_norm_dc4_sel3_01
 .?Number of classums?: 23
 .?Angular file?: ctfTiltparsed_merge_angular
 .?Pixel size?: 6
```

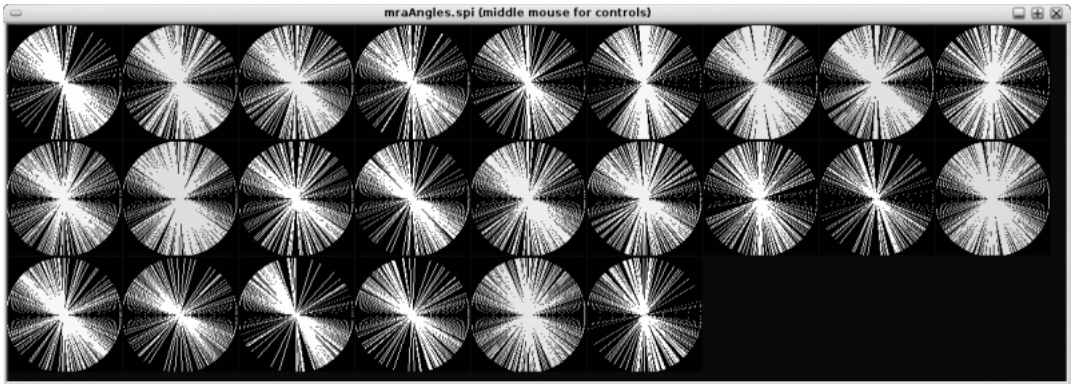
And if you would like to center the particles against the initially calculated volume, run *bp3f_filter_refine_OTRvols_FSC.spi*.

Evaluating model quality

Please see Chandramouli et al. 2011 [http://www.ncbi.nlm.nih.gov/pubmed/21536134] for a more detailed discussion in evaluating OTR model quality.

(1.) Angular bias

This script will plot the distribution of in-plane rotations within classums:



To run:

```
[michael@angula reconstruction_classsums9]$ spider spi @createMRAplot

 \__`O O'__/        SPIDER  --  COPYRIGHT
 ,__xXXXx___        HEALTH RESEARCH INC., ALBANY, NY.
   __xXXXx__
 /   /xxx\   \      VERSION:  UNIX  18.10 ISSUED: 03/23/2010
   /     \          DATE:     14-JUN-2011    AT  18:56:47



 Results file: results.spi.5
 Running: spider
 .?Input MRA angle output from IMAGIC?: rotAngles
 .?Input select file basename?: classsums9_class_
 .?Number of averages?: 23
 .?Name of output stack?: MRAplot
```

And can be found:

```
~michael/BATCHLIB/OTR_scripts/evaluate/
```

(2.) Fourier Ring Correlation between models & reference-free class averages

See Model Validation by 2D reference-free averages.

otr.txt · Last modified: 2012/06/07 16:18 by michael

E-mail to Webmaster