

EMAN2.31 Particle Flexibility Tutorial

This tutorial was updated in December, 2019. It should not be used with versions of EMAN2 older than EMAN2.31.

- ➔ Boxes like this one will contain additional information and tips. You can complete the tutorial without reading any of these, but they may help you better understand what you are doing.
- ➔ The main source for EMAN2 documentation is the Wiki at: <http://eman2.org> . There is also a Google Group for support and discussions: <http://groups.google.com/group/eman2> . Anyone can read/search these sites, you only need to join if you want to post.
- ➔ **GUI Tips:** EMAN2 will work best with a 3-button scroll mouse, though there are alternatives using keyboard modifiers for one button mice or trackpads on Macs.
 - In most EMAN2 windows (2-D images, 3-D volumes, plots, etc.), the middle mouse button will open a control panel for the widget, which is different for each widget type
 - The right mouse button is used for panning in 2-D or 3-D image windows, and can be used to zoom (by shift+dragging), and to reset the zoom (clicking) in plot windows.
 - The scroll-wheel will generally act as a zoom. control-panel for more precise control
 - If you have a one button mouse or trackpad, holding down the alt/option key combined with a mouse click will serve the same role as a middle-click.
 - In the control panels, and other places in the EMAN2 interface you may encounter 'Value Sliders'. A slider is attached to a text-box with a number in it. Dragging the slider controls the number, and entering a number will change the slider. In addition, the text-box can be used to control the range of the slider and get more precise control. By typing **<value** or **>value** in the text box you can change the limits of the slider.
- ➔ Text you see in *italics* will generally refer to labels in the GUI, such as buttons to press. Text you see in **bold**, are commands to be typed in. Items like: <param> are parameters you should fill in (without the < >). Items like: [param] are optional parameters (again, fill without the []).
- ➔ Check your version: The command e2version.py will tell you exactly what version of EMAN you are using. When reporting bugs or asking questions on the mailing list it is critical to include all of the lines of the output of this program with your question.
- ➔ **Mac Laptops** - If you have a Retina display (most will nowadays), then you have a lot of control over the resolution of your display. By default, your display will give you an effective resolution of 1440x900 pixels. This isn't enough space to do image processing well. For the workshop, we strongly suggest opening System Preferences-> Display and adjusting it to "More Space" which will be 1920x1200 effective resolution.
- ➔ **Windows users** - Windows **will** make your life more difficult with EMAN2. If you have no choice but Windows, please remember that programs must be run from the windows command-line, not by clicking on icons. Installation of an 'enhanced command-line' tool for windows, such as *Console 2.x*, will make your life somewhat easier.
- ➔ **Windows** likes to aggressively kill "unresponsive" programs. This means if you have asked EMAN2 to do something from the GUI, and it hasn't finished yet, simply clicking on an open window while you wait may cause the whole program to be killed. We have no solution for this at present other than "don't do that".

Introduction

Many particles in solution have flexible domains, and in some cases the entire structure may be flexible in solution. Even when it is possible to solve a 3-D structure in such cases, very often large domains are missing or blurred out. There are techniques for dealing with this as part of 3-D refinement, but they cannot be used in every situation. Classification methods often rely on having very little preferred orientation, and having extremely large (>1 million) particle sets.

The method described in this tutorial is not a substitute for 3-D classification or other methods for fully addressing this problem. This tutorial will teach you to use an interactive tool for qualitatively exploring the flexibility and dynamics present in your data. It can help you understand the dynamics before taking the computationally much more expensive steps of solving many 3-D structures. While it can certainly be used on very large data sets, it is also especially well suited to cases where the particle populations are somewhat smaller and cases where there are one or more fairly strongly preferred orientations.

1) Extract the demo data from the .zip file, 'cd' into the resulting folder, and make sure eman2 is active (conda activate).

2) Align all of the particles to a single 2-D reference

You will need two things for this analysis, a 2-D reference image, and a stack of particles. If you like, you can be selective and make a set containing only the particles associated with a single class-average, or you can simply align all of the particles in your set to a single reference, then use the N particles that are the most similar to the reference. We will use the second approach in this tutorial. A 2-D reference is also provided for this tutorial.

- ➔ If you prefer to make your own reference, you can simply follow the first few steps of the single particle analysis tutorial, which will provide you with a set of reference-free class averages. You can then use any one of those class averages as an alignment reference. You could also use a projection of the 3-D map resulting from a 3-D refinement.
- ➔ If you wish to extract only particles already associated with one or more class-averages, you can use **e2classextract.py** from the command line or use the **e2evalparticles.py** GUI.

To align the particles to the reference:

```
e2a2d_align.py sets/highc__ctf_flip_lp7.lst aliref.hdf --align rotate_translate_tree:flip=1 --ralign refine --cmp ccc --threads 5 -v 1
```

This should take 2-5 minutes to run and assumes you have a 4 core laptop. It will result in a new folder called m2d_01. This doesn't actually save an aligned copy of the particles, but just a file containing information about the alignment parameters: *m2d_01/particle_parms_01.json*

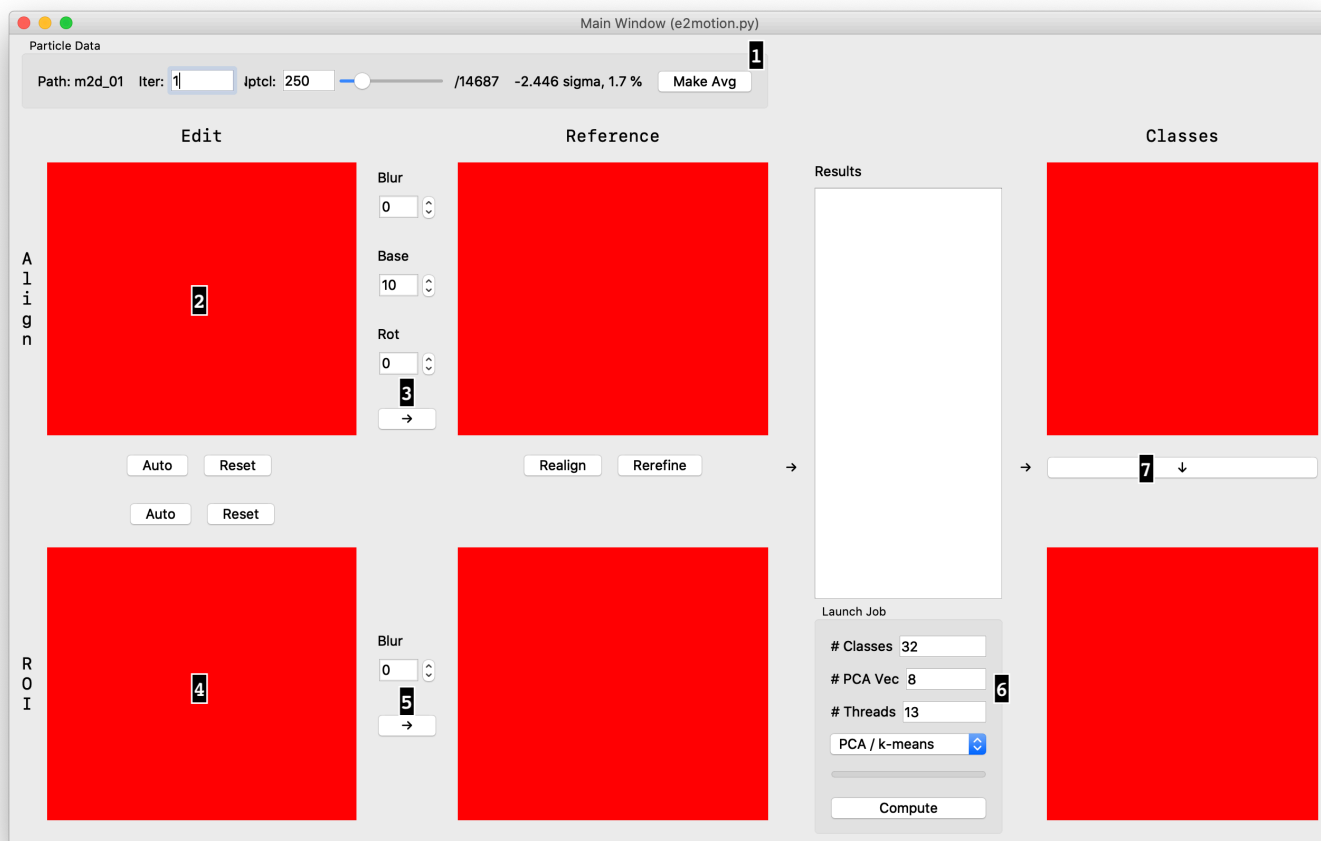
3) Launch e2motion.py

Next we will launch the main GUI for the tutorial. If you run e2a2d_align more than once, the new runs will produce new iterations within the m2d folder. You could also trigger the generation of new m2d_ folders if you like. Assuming you ran the program only once:

```
e2motion.py --path m2d_01 --iter 1
```

When you first launch the program, you should see something like this:

The big red squares may be some other random color. As soon as we have images, these will be replaced. This program remains a usable work in progress, so many aspects of the GUI haven't been fine-tuned yet.



4) Select the number of particles to use for the analysis

We will be progressing through the GUI in order. Start in section 1 by dragging the slider to pick how many particles to include in the analysis. The more particles you include the more orientation variability you will include, in addition to any conformational variability that might exist. In the tutorial data set, try starting with ~500, then press "Make Avg". You can change this number and regenerate the average as many times as you like to see how it impacts the average.

5) Select the region to use for optional realignment of the particles.

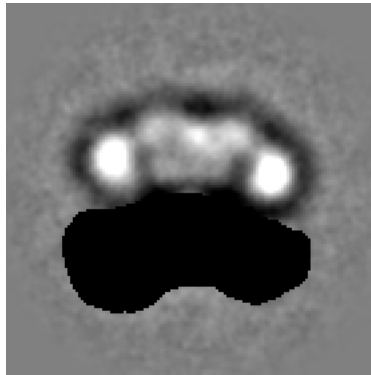
In region 2, you can draw with the mouse to specify a portion of the average to use in realigning the particles. The alignment we start with is an overall alignment of the entire particle to the entire reference. If you wish to hold one portion of the particle fixed, you can focus the alignment on that region.

For the moment, we will skip this part, and use the default alignment. The process for specifying the region to align to is the same as the process we will use next for specifying the region to use for classification. You can return to this step later to play with varying the alignment. Just note that after specifying a region you must press either the **Realign** or the **Rerefine** button to complete the new alignment, and this may take a couple of minutes to complete. If you Realign, note that Iter will be incremented, as this button has a similar effect to running e2a2d_align.py.

6) Select the region to use for classification

In box 4 you should now see a copy of the average of all of the selected aligned particles. We now need to identify one region within this average within which we will search for motion. Use the left mouse button to 'erase' the region you wish to use for classification. You can middle-click to bring up the control panel if you wish to adjust the pen size. If you make a mistake there is unfortunately no undo in this primitive interface, and you will need to press **Reset** and start from scratch.

When you are done, you might have something like:

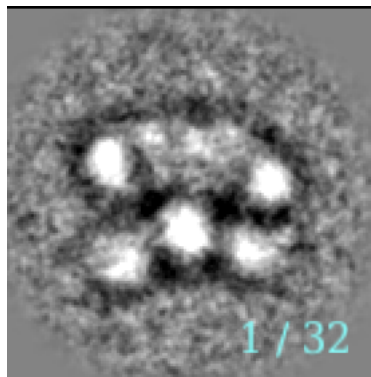


Then press the arrow (marked 5 above). This will show you the region the program will focus the classification on. **Blur** can be used to soften the edge of the region.

7) Perform the classification

In the region marked 5, you can select a number of classes and other parameters. Start with something like 10-20 classes and 8 basis vectors, then press **Compute**.

After a few seconds you should see something like:



in the *Classes* section of the display.

8) Scroll through the class-averages

hold down the up and/or down arrows to scroll rapidly through the class averages. Alternatively, middle-click on the class-average, and you can use the N: slider in the control panel for the same purpose. You should be able to observe variability in the region you selected. You can now repeat the process for different regions and/or with different alignments.

9) Optionally look at the particles in a particular class

If you have a sufficiently current version of EMAN2, you can also press the down-arrow button below the class-average to see the aligned individual particles associated with the currently displayed class-average. This can help you better understand the level of self-consistency among the particles in the class.

10) Optionally return to step 5 for focused alignment

That's the end. Despite its simplicity, this technique can be used to identify some surprisingly complicated conformational changes present in your particle data.

It is, of course, also possible to take this analysis too far, and identify variations in the data which are really just noise. If the changes you observe seem like a rigid body motion of a portion of the particle, it is likely those variations may be real. However, if you observe what seems like random pattern changes and density rearrangements within the selected region, it is likely that those changes are due to noise variability rather than conformational changes in the particles.