

## Chapter 19.10. Single-particle reconstruction with *EMAN*

S. LUDTKE

### 19.10.1. Introduction

Single-particle reconstruction is a technique for determining the three-dimensional structure of identical nanoscale objects without requiring crystallization (Frank, 2006). Typical targets are large proteins or macromolecular assemblies, with  $\sim 200$  kDa regarded as the lower size limit for this technique due to high image noise levels. Typical non-viral targets are in the size range from 500 kDa to 5 MDa, with icosahedral viruses extending to hundreds of megadaltons. Several examples of structures being reconstructed to  $\sim 4$  Å resolution using this technique have been published (Ludtke *et al.*, 2008; Jiang *et al.*, 2008; Zhang *et al.*, 2008; Yu *et al.*, 2008) and subnanometre resolution is frequently achieved, but in negative stain, or with unusually difficult specimens, resolutions may still be limited to 12–30 Å.

### 19.10.2. Overview of *EMAN*

*EMAN* is a suite of scientific image-processing tools with a particular focus on cryogenic electron microscopy (cryo-EM) and single-particle reconstruction. The first stable version was released in 1999 and it has undergone continuous development since that time. Since roughly 2005, development has focused on *EMAN2*, with a completely redesigned modular image-processing library, a workflow system, and a new OpenGL-based graphical user interface (GUI). *EMAN2* supports virtually all image file formats used in the cryo-EM community, as well as various conventions for particle-orientation specification. The image-processing library is completely modular, so new algorithms can be made available throughout the system without any modification of end-user programs. Metadata, which refer to both experimental and derived information describing the image data, are archived using an embedded database system. These metadata can be harvested to produce information for manuscripts or for deposition to centralized databases such as the EMDB (<http://www.emdatabank.org>), operated by the PDBe (Europe) and RCSB PDB (USA).

*EMAN2* has a tiered structure. The GUI interface and workflow represent the top of the hierarchy. Next are individual command-line programs, providing the user with a greater level of control and a wide range of utility functions. Below this is the Python (<http://www.python.org>) layer, for interactive use or high-level programming. Finally, we have the C++ core image-processing library, which handles all computationally intensive image processing. Full documentation is provided through the *EMAN* wiki (<http://blake.bcm.tmc.edu/eman/eman2>).

#### 19.10.2.1. GUI layer and workflow

The workflow interface (Fig. 19.10.2.1) is designed to guide the end user through entire processes, while storing a complete record of all processing completed during any given task. While this interface is gradually being expanded with new tasks, such as tomographic particle averaging, for the purposes of this discussion we will focus on the single-particle reconstruction task, which is broken down into a number of discrete subtasks as discussed in detail in Section 19.10.3. The workflow is designed so

that users wishing to perform only a subset of any task can easily import data into any subtask, proceed through additional subtasks, then export the results at any point. There are future plans to integrate other cryoEM software into the workflow interface to enable users to try alternative algorithms for specific subtasks. The workflow is accessed using either `e2workf1ow.py` or `e2desktop.py`.

The GUI interface (Fig. 19.10.2.2) includes a number of visualization modules for displaying individual two-dimensional images, stacks of two-dimensional images, three-dimensional volumetric models, two-dimensional plots, three-dimensional plots and symmetry information. These modules are used by the interactive file/database browser available in the workflow or *via* the program `e2display.py`, but may also be used directly from the Python interface.

#### 19.10.2.2. Command-line programs

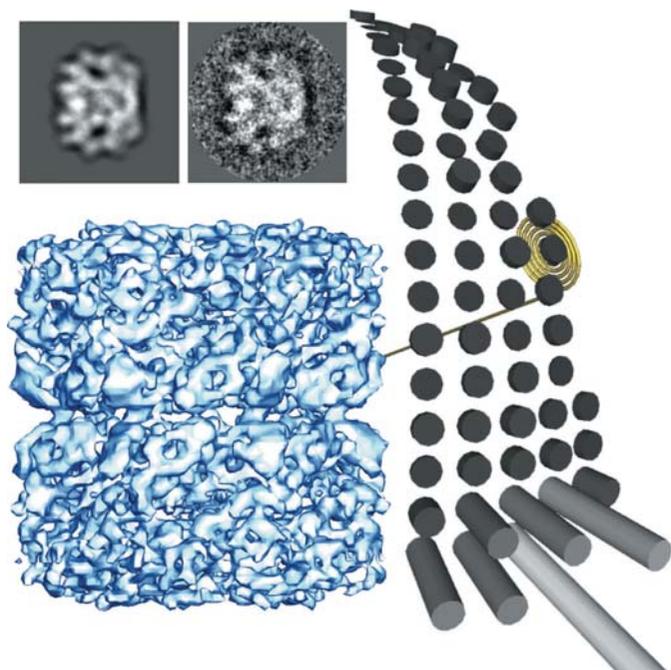
*EMAN2* (Tang *et al.*, 2007) currently contains  $\sim 60$  command-line programs for specific tasks, plus an additional  $\sim 60$  programs as part of the related *SPARX* project (Hohn *et al.*, 2007), included with *EMAN2* releases. High-level programs such as `e2refine2d.py`, `e2initialmodel.py`, `e2eotest.py` and `e2refine.py` are essentially wrapper scripts which call the lower-level Python scripts to complete specific tasks.

In addition, there are a number of general-purpose utility programs, such as `e2proc2d.py` for two-dimensional image processing, `e2proc3d.py` for three-dimensional image processing and `e2bdb.py` for access to the embedded database. Both `e2proc2d.py` and `e2proc3d.py` take similar arguments and



**Figure 19.10.2.1** The overall workflow interface, and the dialogue for the three-dimensional refinement subtask.

## 19. OTHER EXPERIMENTAL TECHNIQUES



**Figure 19.10.2.2**

A representation of the Euler GUI tool. The three-dimensional reconstruction is shown (lower left), along with a projection and class average for a particular orientation identified in the asymmetric triangle on the right. Each cylinder in the asymmetric triangle represents a projection direction, with its height representing the number of particles found in that orientation.

permit application of any of the ~170 modular image-processing algorithms in the core library *via* the `--process` option. These include filters, masks, mathematical operations and a range of other processing algorithms. The `e2help.py` program provides detailed documentation for all of the modular algorithms. All *EMAN2* programs are able to read automatically any of the supported file formats, and will write to any supported file format based on file extension or explicit specification of output format. For example, the command to convert a three-dimensional volume from HDF format to MRC format is simply `e2proc3d.py input.hdf output.mrc`.

### 19.10.2.3. Python wrapper

Python is an easy-to-learn scripting language in wide use in both scientific and non-scientific disciplines. It provides both an interactive prompt as well as a programming interface, and is easily extensible using higher-performance languages such as C++. *EMAN2* was designed such that all command-line programs and the entire GUI interface are written in Python. Thus, any of the distributed programs in *EMAN2* can be modified by experienced end users without a full *EMAN2* compilation environment. In addition, the program `e2.py` offers an *iPython*-based (<http://ipython.scipy.org>) interactive prompt which gives full access to all of the library functions and GUI tools of *EMAN2*. An introduction to this interface can be found at <http://blake.bcm.edu/emanwiki/Eman2ProgQuickstart>.

### 19.10.2.4. C++

The C++ interface contains a modular system, so algorithms of various types can be trivially made available to the entire workflow system. The C++ and Python interfaces were intentionally designed to be as similar to each other as possible, so prototypes of new algorithms can be developed in Python and

**Table 19.10.2.1**

Listing of *EMAN2* supported file formats and whether each has read (R) and/or write (W) support

BDB refers to the *EMAN2* embedded database. DM2/3 are Gatan Digital Micrograph formats. LST files are text files from *EMAN1*.

Format	R or W support	Format	R or W support
BDB	R/W	TIFF	R/W
HDF5	R/W	PGM	R/W
MRC	R/W	PNG	R/W
<i>IMAGIC</i>	R/W	JPEG	W
<i>SPIDER</i>	R/W	LST	R/W
PIF	R/W	<i>AMIRA</i>	R/W
DM3	R	<i>XPLOR</i>	W
DM2	R	VTK	W
EM	R/W	<i>FITS</i>	R/W
ICOS	R/W	SAL	R

then later ported to C++. See <http://blake.bcm.edu/emanwiki/Eman2CProgQuickstart> for an introduction.

### 19.10.2.5. Cross-platform support

All of the major computing platforms are supported, including Linux workstations and clusters, Mac OSX and Windows. Our primary development platforms are Linux and OSX. While we are committed to full support of *EMAN2* on Windows, we have encountered idiosyncratic behaviour on some specific machines running Vista which have defied explanation, but we continue to strive for full support.

### 19.10.2.6. Parallel processing

The three-dimensional refinement process can be extremely computationally intensive. While a refinement of a small ~1 MDa particle with some symmetry to 15 Å resolution may be accomplished on a single workstation in a few hours, a project like an asymmetric reconstruction to 4 Å resolution could easily require hundreds of thousands of CPU hours. Updated details of the modular parallelism strategy of *EMAN* and its GPGPU (general purpose graphics processing unit) methodology can be found in the wiki at <http://blake.bcm.edu/emanwiki/EMAN2/Parallel>.

### 19.10.2.7. File formats and other conventions

*EMAN2* supports all cryo-EM file formats for which specifications were available, in addition to its embedded database storage mechanism. While any *EMAN2* program can read and write images in any supported format, we have adopted HDF5 as an interchange format and the internal database BDB for use during workflow operations. These two permit arbitrary metadata to be associated with each image, unlike the standard cryo-EM formats. Table 19.10.2.1 contains a list of the currently supported formats.

The other primary convention of concern to single-particle reconstruction is three-dimensional orientation specification. While *EMAN2* has its own convention, it can convert to and from the most common conventions in use in the cryo-EM community, including *MRC*, *SPIDER*, *IMAGIC*, quaternions and spin axis. *EMAN*'s own convention uses *Z-X-Z'* Euler angles named *az*, *alt* and *phi*, respectively.