

Xmipp 3.0: Advances in image processing for 3D Electron Microscopy of Single Particles.

J.M. de la Rosa-Trevín, R. Marabini, J. Otón, A. Zaldívar-Peraza, J. Vargas, J.M.

Carazo, C.O.S. Sorzano.

Single particle analysis requires intensive computational analysis of the electron micrographs, employing sophisticated image processing algorithms capable of extracting the weak macromolecular signal from a strong background noise. The use of comprehensive software packages covering all image processing steps from the micrographs to the three-dimensional structure is currently one of the trends in the field.

From a computational point of view, single particle analysis conditions have dramatically changed in the last years: the number of images has been increased by orders of magnitude thanks to microscope automation, and the same time that high-performance computers have evolved into complex architectures with multiple nodes and multiple cores per node. Additionally, users now demand user-friendly, fully traceable, and fast analysis.

With these constraints, we here present a major revision of Xmipp (<http://xmipp.cnb.csic.es>), Xmipp 3.0, including the following main features:

- New image processing algorithms have been incorporated since its latest version (2.4) introduced more than 3 years ago.
- The software has now the concept of project and better integration among the different processing steps has been achieved.
- Typical processing sequences have been reorganized into new image processing workflows.
- All steps are recorded in a database, so that full traceability is guaranteed.
- The User interface has been completely redesigned in order to be more user-friendly.
- Input/output has been made independent from the file format, making Xmipp compatible with the major software packages in the field.
- Careful internal design has been carried out in order to address computational challenges in the future, like the processing of millions of projections and the reconstruction of very large volumes.
- The source code has been completely reengineered, making it shorter, 60% faster, and more efficient in memory use.
- Special attention has been paid to the parallelization of algorithms in supercomputer environments, while assuring its efficient running in small clusters.
- Bindings to the core C++ functionality have been granted from Java and Python.
- Automatic testing procedures have been adopted, improving the quality of the software.

As a result of all these improvements, we have recently presented an example of a 3D reconstruction of the Bovine Papilloma Virus at a resolution about 4Å in less than 24h, using a cluster with 32 processors.