

User Manual for Real-Time Tomography on tomviz

Jonathan Schwartz¹, Chris Harris², Jacob Pietryga¹, Huihuo Zheng³, Prashant Kumar⁴, Anastasiia Visheratina⁴, Nicholas Kotov⁴, Brianna Major², Patrick Avery², Peter Ercius⁵, Utkarsh Ayachit², Berk Geveci², David Muller⁶, Alessandro Genova², Yi Jiang⁷, Marcus Hanwell^{2,8}, and Robert Hovden^{1,9,*}

¹Department of Material Science and Engineering, University of Michigan, Ann Arbor, MI, USA. ²Kitware Inc, Clifton Park, NY, USA. ³Argonne Leadership Computing Facility, Argonne National Laboratory, Lemont, IL, USA. ⁴Department of Chemical Engineering, University of Michigan, Ann Arbor, MI, USA. ⁵The Molecular Foundry, Lawrence Berkeley National Laboratory, Berkeley, CA, USA. ⁶School of Applied and Engineering Physics, Cornell University, Ithaca, NY, USA. ⁷Advanced Photon Source, Argonne National Laboratory, Lemont, IL, USA. ⁸National Synchrotron Light Source II, Brookhaven National Laboratory, Upton, NY, USA. ⁹Applied Physics Program, University of Michigan, Ann Arbor, MI, USA. * e-mail: hovden@umich.edu

Contents

1	Installing tomviz	2
2	Quick Demo: Live 3D Visualization During Reconstruction	2
3	Quick Demo: Live 3D Visualization During Tomographic Experiments (Simulated Demonstration)	6
4	Tutorial: Live 3D Visualization During Tomography Experiments with Electron Microscopes	8
5	Tutorial: Writing Custom Algorithms with Live 3D Visualization	9

Introduction

Electron tomography and cryogenic electron tomography (cryo-ET) generate three-dimensional (3D) reconstructions of native biological and material specimens. However computational bottlenecks in tomographic reconstruction stymie the 3D investigation of specimens. This set of tutorials provides scientists with expedited analysis by enabling real-time tomography with the ability to visualize intermediate volumetric results while reconstruction algorithms or data collection is ongoing. The first two tutorials are quick demonstrations that illustrate real-time 3D visualization capabilities without the need for an electron microscope.

The real-time electron tomography toolset is built using the publicly available tomviz platform. Tomviz is a full featured tomography toolset with real-time analysis and reconstruction capabilities. The software contains a multithreaded pipeline that enables interactive 3D visualization of the current reconstruction state with minimal impact on performance. Thus, scientists can go beyond superficial inspection to quantify specimen features or internal structure while simultaneously operating the microscope. This immediate feedback can save researchers days of effort as reconstructions are no longer processed offline. Real-time tomography also improves offline analysis by dynamically visualizing iterative tomographic reconstructions as they progress. Whether the computation runs online or offline, tomviz users can evaluate 3D specimen structure and optimize the reconstruction accuracy in real-time. These features are highlighted herein as four tutorials: the first two are quick demonstrations of real-time tomography using data pre-bundled with tomviz, two more that demonstrate real-time tomography on an electron microscope, and the last which shows how custom scripts with real-time 3D visualization can be written and used within tomviz.

Tomviz is open-source, freely available to all institutions, and licensed under the permissive 3-clause BSD license.

Schwartz, J. et. al., Real-time 3D analysis during electron tomography using tomviz. *Nature Communications* **13**, 4458 (2022). DOI: 10.1038/s41467-022-32046-0

1 Installing tomviz

To begin, download and install Tomviz from compiled binaries for Mac OS, Linux, or Windows:

tomviz.org/downloads/

The tomviz source-code is available for development or to compile and use:

github.com/OpenChemistry/tomviz

2 Quick Demo: Live 3D Visualization During Reconstruction

In this demonstration, we will visualize a live tomographic reconstruction after all the data has been collected (i.e. offline analysis). Live visualization provides insight into the specimen structure as the computation evolves. This tutorial demonstrates live tomography using projection images collected on a Co₂P hyper-branched star nanoparticle—the dataset is prepackaged with tomviz and does not require additional downloads.

(1) **Load the tilt series data from the drop-down menu.** In this section we're going to start by loading the projection images (i.e. a 'tilt series') from the Sample Data drop-down menu (Fig. 1). The dataset is a collection of projection images acquired across $\pm 75^\circ$ at a $+2^\circ$ increment of a complex nanoparticle. When the data is loaded, it will appear in the 'Pipeline' panel along with visualization modules automatically displayed on the 'RenderView' panel (Fig. 2). To view the projection images at each tilt, select the 'Slice' module () from the 'Pipelines' column, use the 'Slice' slider located below in the 'Properties' panel (highlighted in red) and select the XY plane. These projection images are intentionally slightly misaligned, fortunately our automatic algorithms will implement the necessary corrections.

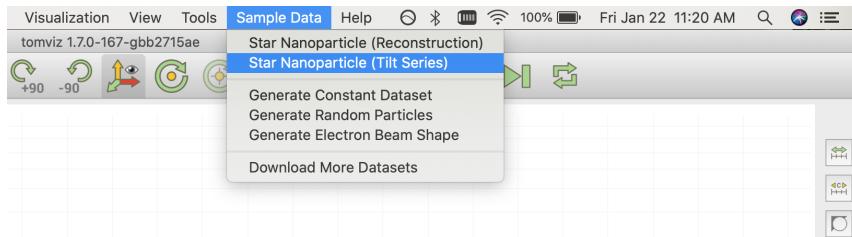


Fig.1 | Load the sample dataset Tomviz comes packaged with sample datasets that include projection images taken across many specimen tilts (i.e. a tilt series).

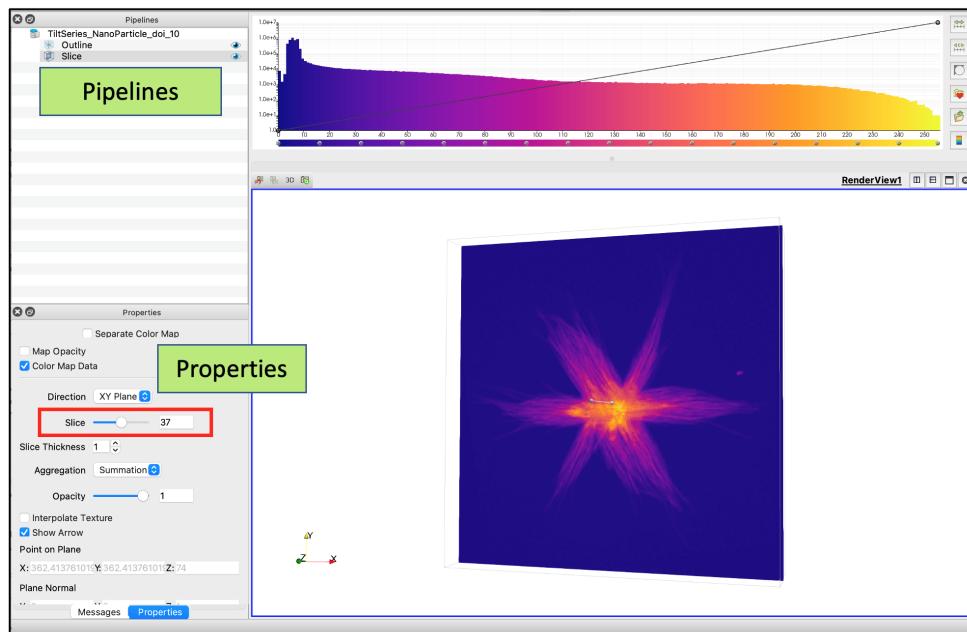


Fig.2 | View the tilt images. When projections are loaded into tomviz it is automatically displayed as orthogonal slices. The parameters of each visualization mode can be set in the 'Properties' panel in the lower left. To view images along different tilts, use the 'Slice' slider (red box) or manually enter the image you want to view.

(2) Preprocess and Align the Tilt Series. For BF-TEM, contrast inversion and CTF correction is often applied. The data can be inverted by ‘Invert Data’ in Data Transforms and CTF correction can be accessed in the Tomography dropdown menu. After the CTF of the instrument is specified [1–3] the image data will be reweighted in Fourier space [4, 5].

The performance of tomography reconstruction algorithms depends on alignment quality. We will begin the alignment process by shifting the projection images so the specimen’s center of mass is located at the origin. Select ‘Image Alignment (Auto: Center of Mass)’ from the Tomography drop-down menu (#1 in Fig. 3b). In addition to aligning all the tilt images, tomviz needs the tilt axis to be centered and parallel to the x-axis. Users can automatically rotate and translate the tilt axis with algorithms available in the tomography drop-down menu. First, apply the auto rotation alignment (#2 in Fig. 3b) and then shift alignment (#3 in Fig. 3b). The data is ready for tomographic reconstruction after all alignments are applied.

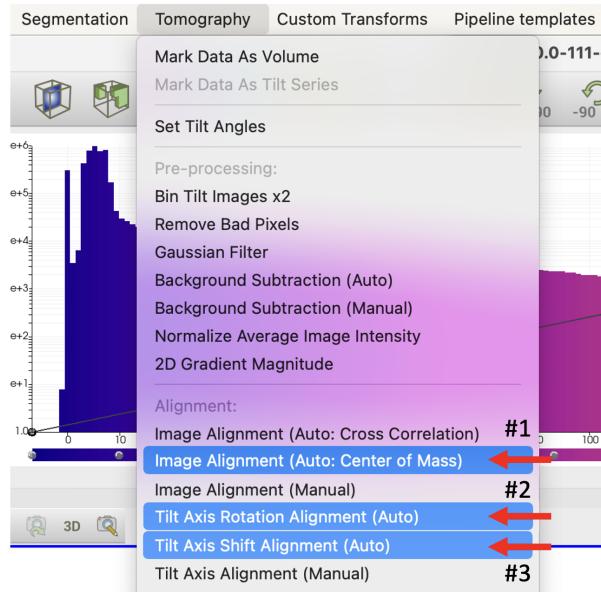


Fig.3 | Align the Tilt Images. The sample tilt series is slightly misaligned. We can automatically correct misalignments by centering the images with the Center of Mass method (#1), rotating the tilt axis (#2), and shifting its location (#3) so its center and parallel to x-axis.

(3) Launch a live-tomographic reconstruction. Several reconstruction algorithms are available with tomviz in the ‘Tomography’ drop-down menu. Run the SIRT algorithm for this live-reconstruction visualization demonstration (Fig. 4).

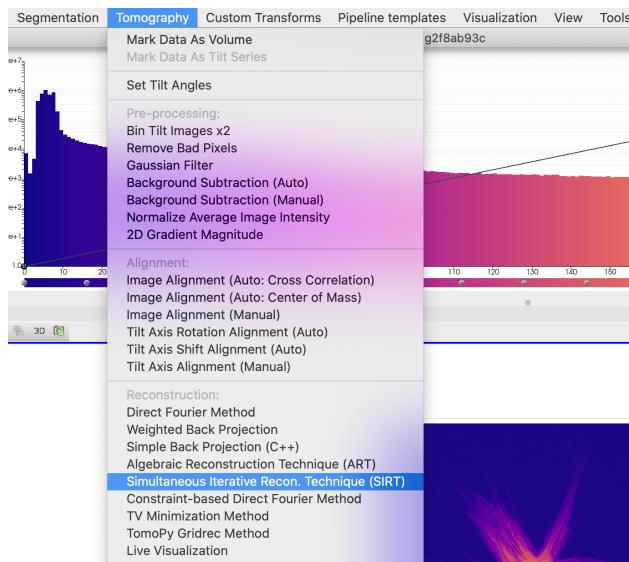


Fig.4 | Select the reconstruction algorithm for live visualization. The tomography menu contains many algorithms for electron tomography. The reconstruction sub-section contains all the iterative and direct algorithms for tomography. Here we will be selecting the Simultaneous Iterative Reconstruction Technique (SIRT).

SIRT is a fast and efficient iterative algorithm. The frequency of visual updates can be specified in the final input box. For this example, we recommend setting the percentage to 100% (Fig. 5). Reducing the frequency of visual updates to 25-50% is useful when datasets are large or computational hardware is limited (e.g. anytime a reconstruction proceeds slower than 1 iteration per second). Tomviz also provides live-visualization capabilities for ART, WBP, or TVmin from the Tomography drop-down menu. Once SIRT parameters have been specified, press the blue ‘Ok’ button (Fig. 5, bottom right hand corner) to begin.

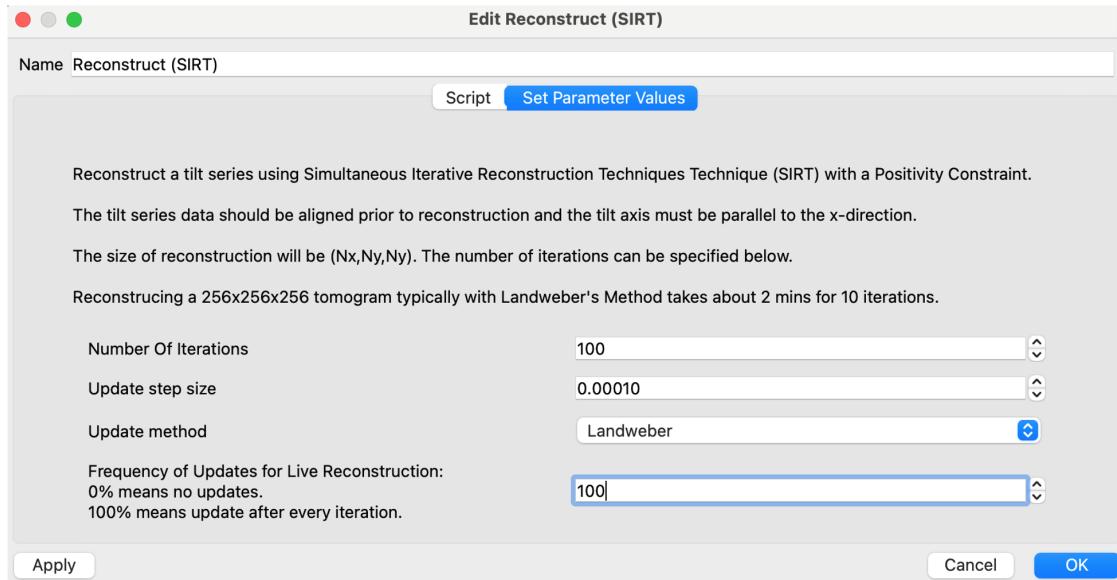


Fig.5 | Enter parameters for the SIRT algorithm

(4) **Visualize the live volumetric process.** Once the reconstruction begins, two additional elements in the ‘Pipelines’ labeled ‘Reconstruct (SIRT)’ and ‘Reconstruction’ will appear. Tomviz by default will continue visualizing the projection tilt images which can be identified from the lack of modules below the ‘Reconstruction’ () dataset icon (Fig. 6a). Delete the previous ‘Outline’ () and ‘Slice’ () modules, or click the eye () to make the previous displays invisible (Fig. 6c), and reassign the modules to the reconstruction. You can visualize the 3D reconstruction by selecting the ‘Volume’ () or ‘Slice’ () from the visualization modules toolbar (Fig. 7) after selecting the ‘Reconstruction’ dataset (). These modules need to appear below the dataset icon (6b,c).

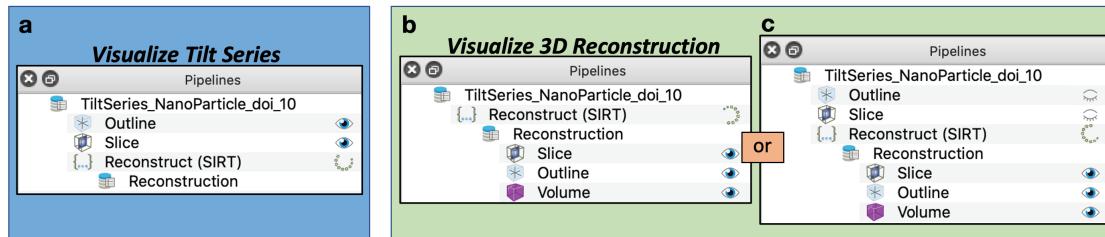


Fig.6 | Visualizing Elements in the Data ‘Pipelines’. Once a dataset has been loaded and live reconstruction begins, the pipeline can be populated with visualization modules of choice. a, By default, tomviz will continue displaying the input dataset after the reconstruction is initialized. b-c, To visualize updates for the live volumetric process, visualization modules should be present below the ‘Reconstruction’ dataset icon.

Volume rendering () is an exceptional method of displaying volumetric objects. Each voxel in the volume is assigned both a color and an opacity based on its intensity. In tomviz, the color-opacity map can be adjusted interactively by dragging points on the line overlaying the histogram. The simplest method for examination of internal specimen structure is with individual 2D slices through the tomogram. Orthogonal slicing () allows users to view slices through the data perpendicular to principal axes (x, y, z). Additional visualization modules such as a constant intensity contours () can be selected for surface rendering.

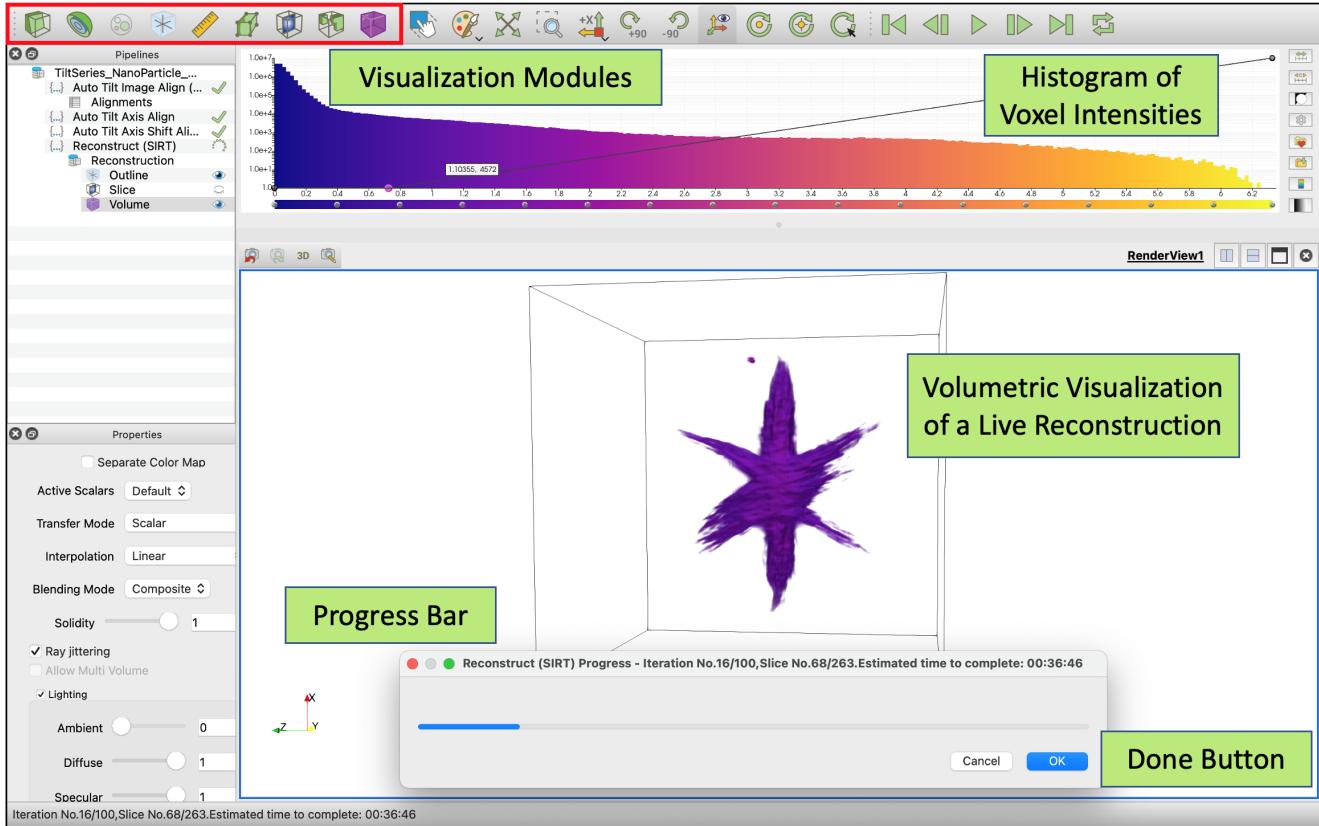


Fig.7 | Visualizing a Live-SIRT reconstruction. The tomviz graphical user interface for 3D visualization contains a variety of visualization tools for analyzing the 3D structure of specimens. 2D () and 3D (or) visualization modules can be selected from the top left menubar. Data transformations and visualizations are recorded in the Data Pipelines column on the left side for reproducible workflows. A histogram of voxel intensities is displayed on the top center where the black line represents the opacity map. Users can exit the reconstruction early by pressing the 'Cancel' or 'Ok' (Done) button.

3 Quick Demo: Live 3D Visualization During Tomographic Experiments (Simulated Demonstration)

In this section we will simulate real-time tomography during tomographic experiments without the need for an electron microscope. Tomviz will monitor a local directory (folder) and automatically append new projections into the reconstruction process. To simulate an experiment, we will sequentially add pre-acquired images into the folder as though the data was being acquired. The steps in this tutorial closely correspond to an experimental scenario where tomviz has been installed on an electron microscope computer except the data has already been acquired.

(1) Initialize real-time tomography. To start a real-time reconstruction, load an empty dataset from the Sample Data drop down menu (Fig. 8a) and press the blue ‘OK’ button (Fig. 8b). A volume comprised of elements with the value zero should appear. To start a real-time reconstruction, users can load any other dataset for this initial step.

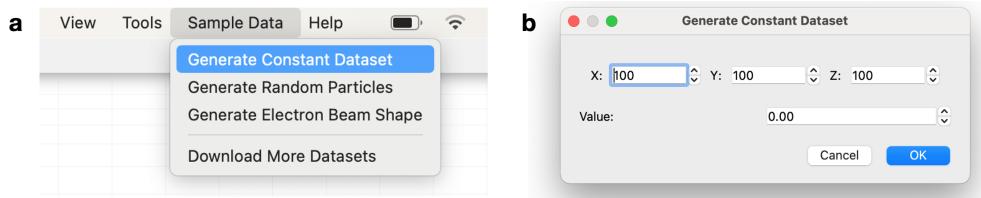


Fig.8 | Load an empty dataset to begin real-time tomography. Generate a constant dataset of all zeros from the Sample Data drop down-menu.

After a dataset is selected, select ‘Mark Data As Tilt Series’ from the Tomography drop down menu (Fig. 9a). Here the tilt angles are irrelevant, continue by pressing the blue ‘Ok’ button and then ‘Initialize Real-Time Tomography’ (Fig. 9b).

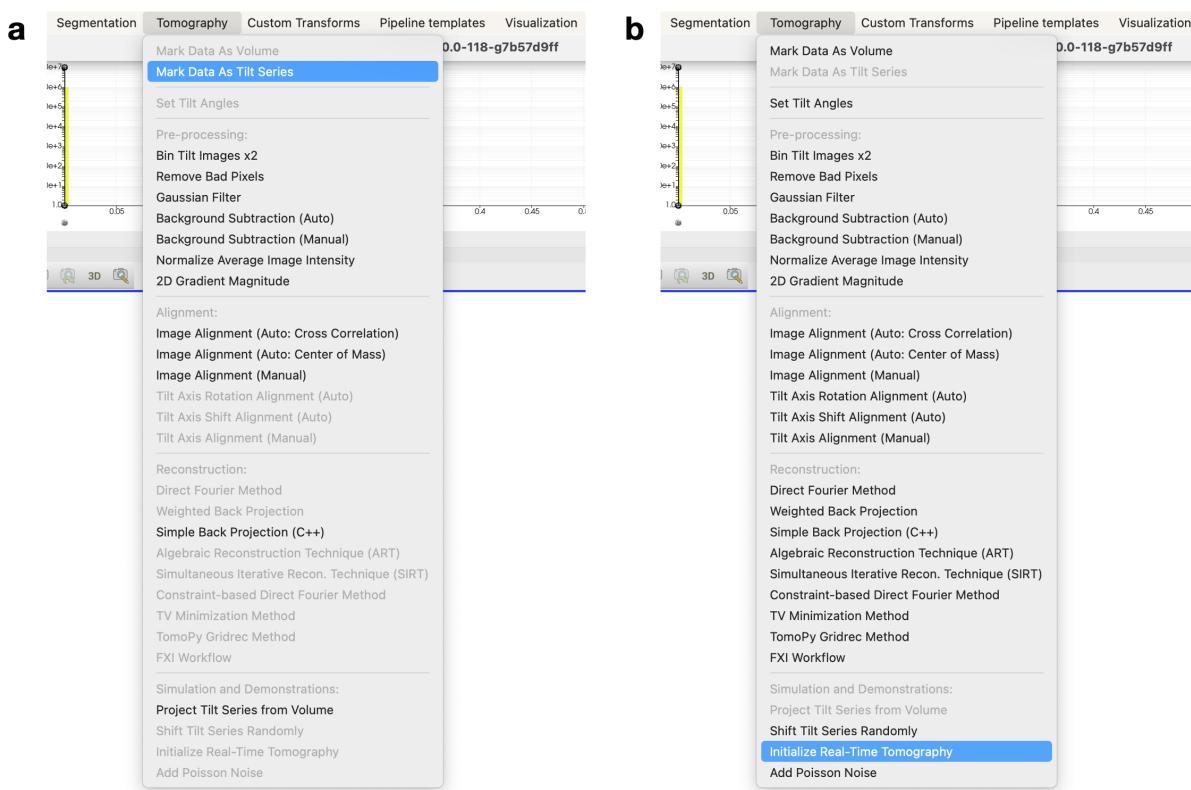


Fig.9 | Initialize a Real-Time Tomography Reconstruction.

(2) Enter parameters to monitor the local directory where data will arrive. Specify the directory that will be monitored for new projections (i.e. the folder where projections are saved). For this simulated experiment we will drag pre-acquired images into this folder. In a real tomographic experiment, the microscope acquisition software would automatically save projections images in this directory. For this demonstration we will define the target local directory as a folder on the Desktop: /Path/To/Desktop/real_time_tomo_demo (Fig. 10).

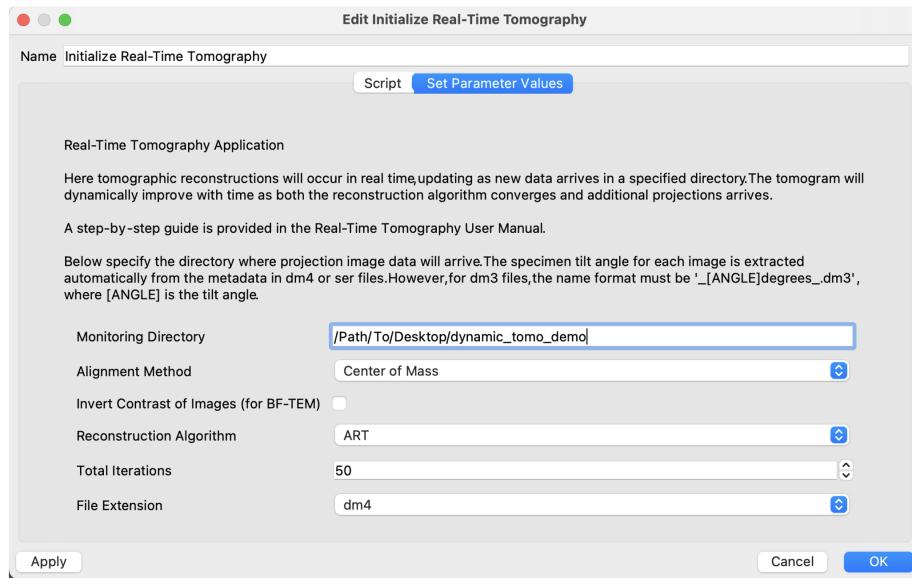


Fig.10 | GUI for real-time tomographic reconstruction.

(3) Enter parameters for tomography reconstruction. Reconstruction parameters such as the desired algorithm, number of iterations, and micrographs' file extension (dm4, dm3, ser) can be specified in this sub-menu. Tomviz can extract the tilt angle from the metadata embedded in dm4 and ser files. Unfortunately the metadata is not available in dm3 files, thus tomviz needs to parse the tilt angle from the file name in names in the following format: ‘*_ANGLEdegrees_*dm3’, where [ANGLE] is the tilt angle. The default parameters provided in the dialog box (Fig. 10) will be sufficient. For this tutorial, use the tilt series provided as Supplementary Data Set 1. Download the dataset from the Supplemental, we will load the data into tomviz in the next step.

(4) Run the real-time tomographic simulation. Once ready, click the blue ‘OK’ button (Fig. 10, bottom right hand corner). You should see a new element in the data pipelines column and a progress bar will appear (shown in Fig. 7). The progress bar will initially display ‘Initialize Real-Time Tomography Progress’, this means the script is currently monitoring the folder dynamic_tomo_demo for new images. Start the simulation by dragging a few tilt-images into the target directory (Fig. 11). Once projections are detected, the progress status message will update to ‘Reconstructing Tilt Angles’ and the reconstruction will begin.

Select the ‘Reconstruction’ dataset from the ‘Pipelines’ panel and either the volume () or orthogonal slices () from the visualization modules toolbar (Fig. 7) to visualize the structure and observe the specimen evolution. New projections will be incorporated into the reconstruction process after the total number of iterations is complete (in this case 50). Copy 1-3 projections into the target folder every minute to simulate usual experimental acquisition speeds. We recommend copying data in chronological order, however, tomviz can still reconstruct the data if projections are passed in a random order. Once satisfied with the reconstruction, press ‘Ok’ (Fig. 7) to exit the reconstruction and continue with any desired post-processing.

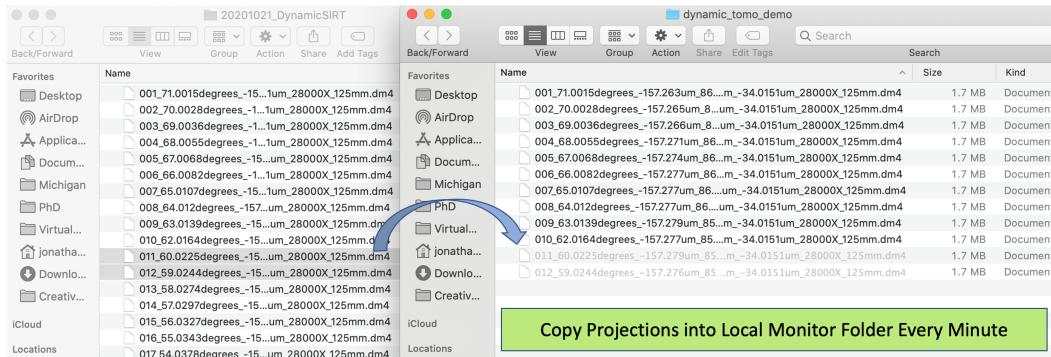


Fig.11 | Copy individual projections into monitor directory. Tomviz responds to changes in monitored folders once files are saved into the directory.

4 Tutorial: Live 3D Visualization During Tomography Experiments with Electron Microscopes

In this tutorial we will explain how to perform real-time tomographic experiments while projection images are collected on a scanning / transmission electron microscope (S/TEM). The process is nearly identical the previous section, however experimental real-time tomography requires that the microscope is aligned for tomography and the newly acquired data can be accessed by tomviz.

(1) Align the microscope and prepare the tomography experiment. We recommend using a high-tilt tomography holder to perform single-axis tomography. The microscope must be properly aligned before the acquisition begins to prevent any correctable distortions. Align the microscope as one normally would for high-resolution imaging to reduce beam aberrations (e.g. stigmation or coma). Next, find the maximal allowable (positive and negative) tilt range such that the region of interest is visible. Achieving a $\pm 70^\circ$ tilt range or larger will provide best results.

Aligning the sample at eucentric height is essential to minimizing specimen drift and the need for any stage refinement during acquisition. Figure 12a shows an object located below the eucentric height (\otimes) with the electron beam traveling from top to bottom. The object shifts and rotates when tilted around eucentric height, while proper alignment allows the specimen to simply rotate without any spatial translation (Fig. 12b).

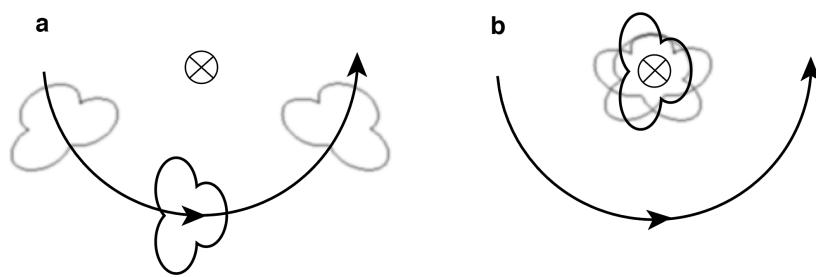


Fig.12 | Specimen drift from eucentric height. **a**, If the eucentric height for the goniometer is incorrect, tilting the specimen holder causes the specimen to sweep along an arc around the tilt axis. **b**, After correcting the stage position, the specimen drift is reduced significantly.

In order to bring the stage to eucentric height, one should measure the sample drift as the stage tilts from -50° to $+50^\circ$. Adjust the stage height by the drift's magnitude and iterate through this process until motion is minimized with minimal movement only due to the goniometer backlash. Choose a field of view which accommodates the specimen or region of interest across all tilts without the need to move the stage. Moreover, check that the detector camera length and gain does not clip image intensities at any one tilt. All calibrations and acquisition parameters for real-time tomography are now set properly and should not be changed during the tomography experiment. Move over to tomviz to set reconstruction parameters prior to starting the experiment.

(2) Initialize real-time tomography. To start a real-time reconstruction, load an empty dataset from the Sample Data drop down menu (Fig. 8), Mark the Data as a Tilt Series (Fig. 9a), and then select ‘Initialize Real-Time Tomography’ from the Tomography drop-down menu (Fig. 9b).

(3) Enter parameters for local directory monitoring. Users can specify the directory that will be monitored for new projections (i.e. the folder where projections are saved). If tomviz is installed on the microscope computer, the acquisition software could automatically save projections images in this directory. However, this is not a necessary requirement as users can manually drag images into the folder or download it onto a personal machine (e.g. laptop). For this demonstration let’s define the target local directory as a folder on the Desktop: /Path/To/Desktop/real_time_tomo_demo (Fig. 10).

(4) Enter parameters for tomography reconstruction. Reconstruction parameters such as the desired algorithm, number of iterations, and micrographs’ file extension can be specified in this sub-menu. Tomviz will read the meta data to extract experimental parameters such as specimen alpha tilt. The default parameters provided in the dialog box (Fig. 10) will be sufficient.

(5) Start the real-time tomography experiment. Once ready, click the blue ‘OK’ button (Fig. 10, bottom right hand corner) and start collecting projection images. After collecting one or two tilt images, transfer the acquired micrographs into the target directory (Fig. 11). The reconstruction will begin immediately after projections are detected inside the folder. Select either the volume () or orthogonal slices () from the visualization modules toolbar (Fig. 7) to observe the specimen evolution. New projections will be incorporated into the reconstruction process after the total number of iterations is complete (in this case 50). Once satisfied with the reconstruction, press Ok (Fig. 7) to exit the reconstruction and continue with any desired post-processing.

5 Tutorial: Writing Custom Algorithms with Live 3D Visualization

Developing custom scripts for live 3D-visualization of any volumetric process simply requires 3 extra steps into any Python script that runs on tomviz. Below we provide a sample script that highlights the necessary steps for live visualization. First, handoff the data from tomviz to Python (line 11). Next, initialize the intermediate container (child) for visualization (line 17). Implement the section for the volumetric computation (line 22) and update the visualization state as the computation progresses (lines 28-29). Users can also initialize a progress bar (line 8) and update its length (line 25) with the current iteration number.

```
1 import tomviz.operators
2 import numpy as np
3
4 class DemoLiveVisualization(tomviz.operators.CancelableOperator):
5     def transform_scalars(self, dataset, Niter=100):
6
7         # Initialize the Progress Bar
8         self.progress.maximum = Niter
9
10        # Get the current volume as a numpy array.
11        vol = dataset.active_scalars
12
13        # Get angles if dataset is tilt series
14        angles = dataset.tilt_angles
15
16        # Create child for recon
17        child = dataset.create_child_dataset()
18
19        # Main Loop
20        for ii in range(Niter):
21
22            # Do Work
23
24            # Update the Progress Bar
25            self.progress.value = ii
26
27            # Visualize Current Iterate
28            child.active_scalars = vol
29            self.progress.data = child
30
31            # One last update of the child data.
32            child.active_scalars = vol
33            self.progress.data = child
34
35            # Return the Child to tomviz for Subsequent Processing
36            returnValues = []
37            returnValues["reconstruction"] = child
38
39        return returnValues
```

To load a custom Python script into tomviz, select Import Custom Transform from the Custom Transforms drop down menu (Fig. 13a). After the Python script is imported, it will be available as an additional data transformation (Fig. 13b).

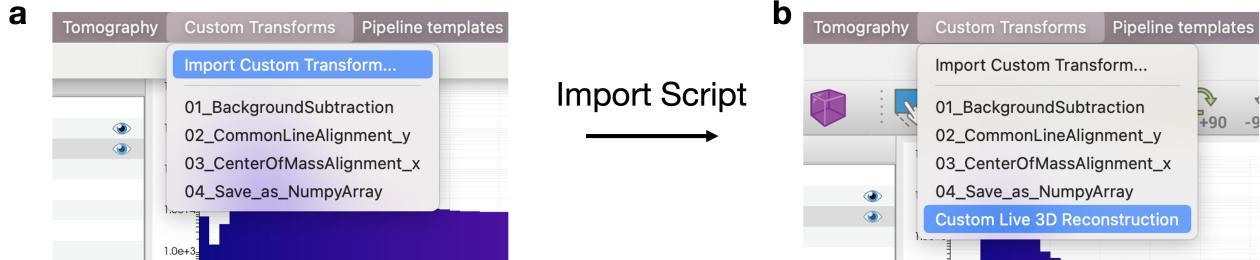


Fig.13 | Load Custom Python Scripts

This will create a copy of the imported Python script in `~/tomviz`. In addition to importing the Python script, we will create an associated JavaScript Object Notation (JSON) file in this location. The JSON file needs to have the same name as the Python script. We can specify its label in the Custom Transforms drop down label in line 3. Users can also specify input parameters (in this case number of iterations - `Niter`) in the JSON file (line 12):

```

1  {
2      "name" : "customReconstruction",
3      "label" : "Custom Live 3D Reconstruction",
4      "description" : "Write a description for the custom Python transformation.",
5      "children": [
6          {
7              "name": "reconstruction",
8              "label": "Reconstruction",
9              "type": "reconstruction"
10         }
11     ],
12     "parameters" : [
13         {
14             "name" : "Niter",
15             "label" : "Number of Iterations",
16             "type" : "int",
17             "default" : 100,
18             "minimum" : 1
19         }
20     ]
21 }
```

It is worth highlighting, tomviz can also monitor local or remote files in directories to dynamically update a 3D volume as the data changes. This means a reconstruction or volumetric process that is running remotely (e.g. on a high-performance cluster) can incrementally write the results to file and these updates will be rendered in tomviz. This approach readily accommodates computing (local or remote) that is ran outside of tomviz while simultaneously providing real-time 3D visualization.

References

1. Rohou, A. & Grigorieff, N. CTFFIND4: Fast and accurate defocus estimation from electron micrographs. *J. Struct. Biol.* **192**, 216–221 (2015).
2. Bell, J., Chen, M., Baldwin, P. & Ludtke, S. High resolution single particle refinement in EMAN2.1. *Methods* **100**, 25–34 (2016).
3. Tegunov, D. & Cramer, P. Real-time cryo electron microscopy data preprocessing with Warp. *Nat. Methods* **16**, 1146–1152 (2019).
4. Mindell, J. & Grigorieff, N. Accurate determination of local defocus and specimen tilt in electron microscopy. *J. Struct. Biol.* **142**, 334–347 (2003).
5. Downing, K. & Glaeser, R. Restoration of weak phase-contrast images recorded with a high degree of defocus: The twin image problem associated with CTF correction. *Ultramicroscopy* **108**, 921–928 (2008).