

# AreTomo3 User Manual

Shawn Zheng

Chan Zuckerberg Imaging Institute

[shawn.zheng@czii.org](mailto:shawn.zheng@czii.org)

**AreTomo3** is a multi-GPU accelerated software package that enables real-time fully automated reconstruction of cryoET tomograms in parallel with cryoET data collection. Integrating MotionCor3, AreTomo2, and GCTfFind in a single application, AreTomo3 has established an autonomous preprocessing pipeline that, whenever a new tilt series is collected, can activate and repeat itself from correction of beam induced motion and assembling tilt series to CTF estimation and correction, tomographic alignment, and 3D reconstruction throughout entire session of data collection without human intervention. The end results include not only tomograms but also a rich set of alignment parameters to bootstrap subtomogram averaging. Our test shows that AreTomo3 can catch up to 9-target PACE Tomo data collection with 4 NVidia RTX A6000 GPUs when it was configured to perform 2D local motion correction on movies and 3D local motion correction on tilt series. The offline testing shows that AreTomo3 runs faster than the data collection. As a result, GPU resources can be shared with other tasks to expand the preprocessing capacity. Tomogram denoising and particle picking now can run concurrently with AreTomo3 to maximize the preprocessing workflow.

Version 1.0.15

June 11, 2024

# Contents

|  |    |
|--|----|
| AreTomo3 User Manual                           | 1  |
| 1. System Requirement and Installation         | 3  |
| 1.1 Install from source                        | 3  |
| 1.2 Compile the library files                  | 3  |
| 1.3 AreTomo3 Configuration                     | 3  |
| 2. mdoc File Requirement                       | 4  |
| 3. Supported Movie Files                       | 4  |
| 4. Use Cases                                   | 5  |
| 4.1 Full processing pipeline                   | 5  |
| 4.2 Interrupting and Resuming                  | 7  |
| 4.3 Skipping correction of beam induced motion | 7  |
| 4.4 Tomographic reconstruction only            | 7  |
| 4.5 CTF estimation only                        | 7  |
| 4.6 CTF estimation and local CTF correction    | 7  |
| 5. Generated Files                             | 8  |
| 5.1 Tilt series                                | 8  |
| 5.2 Tomograms                                  | 8  |
| 5.3 Tilt angles and acquisition sequence       | 8  |
| 5.4 CTF estimation files                       | 9  |
| 5.5 Alignment file                             | 9  |
| 6. Extra Command Line Options                  | 11 |
| 6.1 Motion correction                          | 11 |
| 6.2 Tomographic alignment                      | 12 |
| 6.3 CTF estimation and deconvolution           | 12 |
| Reference                                      | 13 |
| Technical Support                              | 14 |

# 1. System Requirement and Installation

**AreTomo3** is a multi-GPU accelerated Linux application written in C++/CUDA. It is recommended to run AreTomo3 on a Linux platform equipped with four or more state-of-the-art NVidia GPUs. Ideally each GPU has at least 20 GB memory. Systems with GPUs having smaller memory may also work but are not tested. It is recommended to have at least 512 GB CPU memory in total or 100 GB CPU memory per GPU on average. When AreTomo3 is run on fewer GPUs, less CPU memory is needed at the cost of reduced performance.

AreTomo3 is a single-program application. Once unzipped, no further installation is needed. However, AreTomo3 depends on **CUDA driver** and **toolkit** as well as **libtiff** library. Ideally they are installed by system administrators and thus shared across different user accounts. If not, users can install them using anaconda or miniconda in their local accounts. Once installed, their installation paths need to be added to **LD\_LIBRARY\_PATH**, a Linux environment variable in **.bashrc** or **.cshrc** file.

## 1.1 Install from source

When the source code is downloaded, the binary can be generated using two included **makefile** and **makefile11**. Open a Linux terminal and execute at the command prompt **make exe** if CUDA 10.x is installed or **make exe -f makefile11** when CUDA 11.x or 12.x is installed. If you do not know what CUDA version is installed, it can be found by running **nvcc -- version** at the command prompt.

Both makefile and makefile11 define **CUDAHOME**, a variable that holds the full path to CUDA toolkit installation point. Use CUDAHOME in the provided makefile and makefile11 as an example to help locate where CUDA toolkit is installed on your system. Then revise CUDAHOME accordingly.

## 1.2 Compile the library files

There are two precompiled library files, **libutil.a** and **libmrcfile.a** in **LibSrc/Lib**. If you run into any problems in linking these two files, try recompiling them using the provided makefiles in their respective source directories. **Compile libutil.a first** followed by **libmrcfile.a**. Run **make clean** and then **make all** in **LibSrc/Util** directory and this will generate **libutil.a**, which is automatically moved into **/LibSrc/Lib** at the end of compiling. Repeat the same steps in **LibSrc/Mrcfile**.

## 1.3 AreTomo3 Configuration

Users need to set up environment variable **LD\_LIBRARY\_PATH** to include the full path of CUDA library. You use check the value of it by running “**echo \$LD\_LIBRARY\_PATH**”. In the following screen shot, the path of CUDA library is underscored with the yellow line.

```
(base) [shawn.zheng@czii-gpu-a-1 ~]$ echo $LD_LIBRARY_PATH
/hpc/apps/x86_64/cuda/12.0.0_525.60.13/lib64:/hpc/apps/x86_64/cuda/12.0.0_525.60.13/lib:/home/shawn.zheng/miniconda3/lib:/hpc/slurm/installed/current/lib
```

When you see an error message like the below, it is likely that **LD\_LIBRARY\_PATH** is not set up properly.

```
/home/shawn.zheng/szheng/CuProjs/AreTomo2/AreTomo2: error while loading shared libraries:
libcufft.so.11: cannot open shared object file: No such file or directory
```

## 2. mdoc File Requirement

Tilt series **mdoc** files are needed for running AreTomo3. A tilt series mdoc file contains multiple **ZValue** sections, one for each tilt angle. The following is an example of a ZValue section.

```
[ZValue = 0]
TiltAngle = -0.03
StagePosition = 54.95 56.35
StageZ = 0.82
Magnification = 81000
Intensity = 0.00
ExposureDose = 3.76
PixelSpacing = 1.54
SpotSize = 6
Defocus = -3.97
ImageShift = 3.24 0.33
RotationAngle = 83.94
ExposureTime = 1.00
Binning = 1
MagIndex = 29
CountsPerElectron = 2121.78
MinMaxMean = 502.00 4407.00 2154.42
TargetDefocus = -2.00
PriorRecordDose = 0.00
SubFramePath = 20240216_002_Krios1_RP_5proteinsMixB1s_grid1\Position_5_10_001_-0.01_20240216_111455_EER.eer
NumSubFrames = 1
FrameDosesAndNumber = 3.76 1
DateTime = 16-Feb-2024 11:14:58
FilterSlitAndLoss = 10.00 0.00
ChannelName =
CameraLength = NaN
```

**Fig. 1** Excerpt of tilt series mdoc file showing the first ZValue section. Each tilt image has a ZValue section. AreTomo3 parses each ZValue section to extract the tilt angle and the file name of movie. In some cases, data collection software may also produce per-tilt mdoc file. **Make sure to use tilt series mdoc file.**

AreTomo3 uses **ZValue**, **TiltAngle**, and **SubFramePath** to assemble motion-corrected tilt images into tilt series and produce the ordered list file needed by Relion subtomogram averaging. Since movie files can be transferred for one location to another, the directory path provided by SubFramePath is ignored in AreTomo3. Instead, **AreTomo3 requires mdoc files to be with their movie files in the same and flat directory.**

Note that some data collection software may also generate per-tilt mdoc files. **AreTomo3** needs **tilt-series mdoc** files that contain multiple ZValue sections.

## 3. Supported Movie Files

AreTomo3 supports both eer and tiff movies. For eer movies, it is strongly recommended to use **frame integration file** to generate rendered movies for measuring and correcting beam induced motion since processing raw movies is extremely time consuming and demanding on computing resources. Frame integration is a software implementation variable frame rates in exposure and considers the fast early motion. An example is given below.

```

50 10 0.006
100 20 0.006
300 30 0.006

```

**Fig. 2** An example of frame integration file, a 3-column text file that specifies how raw frames are summed to generate rendered frames. The measurement and correction of beam induced motion are performed on rendered frames, not raw frames. In this example, the first column shows that the raw frames are divided into 3 non-overlapping groups of sizes 50, 100, and 300, respectively. The second column decides how many raw frames are summed up to form a rendered frame in each group. Column 3 shows the raw frame dose that is used to calculate the rendered frame dose. If the sum of the first column exceeds the total eer frames, AreTomo3 will make automatically adjustment.

Use **-FmIntFile** followed by the path of the frame integration file to enable this function.

## 4. Use Cases

**AreTomo3** can be invoked from Linux command line. The version number and built date can be found by running **AreTomo3 --version**. If this command fails, this usually means that there are some issues with CUDA toolkit installation. One problem could be the LD\_LIBRARY\_PATH is not configured to include the path to the proper CUDA library. Users can get quick help information about the command line parameters by running command **AreTomo3 --help**.

The major operations in **AreTomo3** include correction of beam induced motion, assembling full, odd, and even tomographic tilt series, CTF estimation, tomographic alignment, local CTF correction of tilt series, and tomographic reconstruction of local-CTF corrected full, odd, and even tomograms.

**AreTomo3** has multiple entry points that allow users to repeat a subset of operations. These entry points are described below. The different entry points are controlled by **-Cmd**.

| Entry Point   | Operations                             |
|---------------|--|
| <b>-Cmd 0</b> | Full processing, default               |
| <b>-Cmd 1</b> | Skip only motion correction            |
| <b>-Cmd 2</b> | Repeat only tomographic reconstruction |
| <b>-Cmd 3</b> | Repeat only CTF estimation             |

### 4.1 Full processing pipeline

Full processing pipeline starts with the correction beam induced motion and ends with the generation of 3D tomograms. Here is the command line that we used to run AreTomo3.

```

[shawn.zheng@czii-gpu-a-1 ~]$ ~/szheng/CuProjs/AreTomo3_1.0.7/AreTomo3 -InMdoc Position_
-InSuffix .mdoc -Gain /hpc/instruments/czii.krios1/OffloadData/ImagesForProcessing/EF-Fal
con/300kV/20240308_100035_EER_GainReference.gain -FmIntFile ~/FmIntFile.txt -OutDir ~/szh
eng/Temp -McPatch 5 5 -McBin 1 -Group 2 4 -PixSize 1.54 -AtBin 6.49 -AtPatch 4 4 -FlipVol
1 -OutImod 1 -Wbp 1 -Gpu 0 1 2 3 4 -Serial 90000 2>/dev/null

```

**Fig. 3** A command line example for running live or offline full processing. Since **-Cmd** is missing in this command line, **-Cmd 0** is used as default that starts full processing from correction of beam induced motion on movies to reconstruction of tomograms.

**-InMdoc** is followed by the prefix of mdoc files. In this example, **AreTomo3** processes all mdoc files in the current directory and whose names start with “Position\_”. If mdoc files are saved in a different directory, then the path should be included.

**-InSuffix** is used together with **-InMdoc**. In this example, **AreTomo3** processes all files whose names end with “.mdoc”. Users can include more ending characters to make it more restrictive.

**-Gain** specifies where to load the gain reference file. **AreTomo3** can load gain reference saved in .gain or .mrc files.

**-OutDir** specifies where the output directory is. In this example, the generated tilt series, tomograms, CTF estimation results, and alignment parameters are saved in **~/szheng/Temp** directory.

**-InFmIntFile** specifies where to load the frame integration file that converts raw frames into rendered frames, non-overlapping sums of raw frames. This is a software implementation of variable frame rates in exposure in collecting movies to consider fast early beam induced sample motion. As a result, early rendered frames usually have less raw frames.

**-McPatch 5 5** informs **AreTomo3** to run 5x5 patch-based motion correction. If not specified, only global motion is corrected.

**-McBin 1** informs **AreTomo3** not to perform Fourier cropping on the motion corrected images.

**-Group 2 4** specifies how to group rendered frames for measuring global and local beam induced motion. Beam induced motion is measured on the group sums and then interpolated/extrapolated to individual rendered frames. In this example, every 2 rendered frames are summed for measuring global motion and every 4 rendered frames for measuring local motion.

**-PixSize** specifies the pixel size of the input movies. The pixel size of motion corrected images is the multiplication of this pixel size and the **-McBin** value.

**-AtBin 4** informs **AreTomo3** to perform 4x binning by Fourier cropping the aligned tilt series before tomographic reconstruction. The pixel size of the generated tomogram is the multiplication of the **-PixSize**, **-McBin**, and **-AtBin** values.

**-AtPatch 4 4** invokes 4x4 patch-based local alignment after global tilt series alignment.

**-FlipVol 1** flips the generated tomogram by rotating it around x axis. When the tomogram is saved into a MRC file, each section is a x-y slice of the tomogram. By default, tomograms are not flipped. In this case, a MRC section is a x-z slice of tomograms.

**-OutImod 1** generates needed Imod files to bootstrapping Relion subtomogram averaging. These Imod files are saved in a created sub-folder in the output directory.

**-Wbp 1** specifies weighted back projection for tomogram reconstruction. Another option is SART reconstruction.

**-Gpu** is followed by GPU IDs that are involved in the computation.

**-Serial 90000** means that AreTomo3 will wait maximum 90000 seconds for the next tilt series to be collected. If still no available, AreTomo3 will quit after finishing all the ongoing processing. This is specific for the live processing. When a new tilt series becomes available, AreTomo3 will reset the timer and wait another 90000 seconds before quitting.

## 4.2 Interrupting and Resuming

**AreTomo3** can be interrupted by **ctrl c** command by pressing **ctrl** and **c** keys at the same time. This mechanism can terminate running AreTomo3 either at the end of data collection or when users need to adjust AreTomo3 command line parameters.

**AreTomo3** can also be resumed after it is interrupted without reprocessing tilt series that have been reconstructed. This is implemented by tracking finished tilt series in a text file (**MdocDone.txt**) saved in the output directory. **-Resume 1** works with **-Cmd 0** to resume the processing when AreTomo3 is terminated prematurely. This function instructs **AreTomo3** to skip tilt series that are already reconstructed and proceeds with newly corrected tilt series.

## 4.3 Skipping correction of beam induced motion

This is a reprocessing without starting all over. As a result, full processing (**-Cmd 0**) needs to be finished first. When later one users just want to re-align the tilt series, **-Cmd 1** lets users do so. **AreTomo3** needs information from both input and output directories to determine what tilt series to be aligned. For this reason, the **-InMdoc** and **-OutDir** setup in the command line must be the same as that in the previous full processing. Users can change settings for tomographic alignments (**-AtBin**, **-AtPatch**, **-VolZ**, **-AlignZ**, **-OutImod**, **-FlipVol**, **-Wbp**, and **-Sart** etc.) Motion correction related parameters are ignored even if they are present in the command line.

## 4.4 Tomographic reconstruction only

**-Cmd 2** goes straight to reconstruction of tomograms. The same as section 4.3, full processing needs to be done first. The **-InMdoc** and **-OutDir** setup in the command line must be the same as that in the previous full processing. Users can change how volume is reconstructed by changing the settings of **-VolZ**, **-FlipVol**, **-Wbp**, and **-Sart**.

## 4.5 CTF estimation only

**-Cmd 3** lets users repeat only CTF estimation. The same as section 3.2, full processing needs to be done first. The **-InMdoc** and **-OutDir** setup in the command line must be the same as that in the previous full processing.

## 4.6 CTF estimation and local CTF correction

CTF estimation is enabled when the pixel size is passed into AreTomo3 using **-PixSize**. Otherwise, no CTF estimation will be performed, nor will be the CTF correction. The default values for high tension and spherical aberration are 300 and 2.7, respectively. Users can use **-kV** and **-Cs** to pass different values.

**-CorrCTF 1** enables local CTF correction of original tilt series when CTF estimation is performed. The CTF correction is tile based with each tile having its own CTF calculated based on its distance from the tilt axis. CTF correction is done on odd, even, and full tilt series before they are reconstructed into tomograms.

## 5. Generated Files

### 5.1 Tilt series

After each movie is motion corrected, sums of odd frames, even frames, and all frames are generated. After all tilt movies are corrected, the odd, even, and full sums are assembled into odd, even, and full tilt series in the ascending order of tilt angles. A tilt series is named after the corresponding mdoc file. The odd and even tilt series are appended with **\_ODD** and **\_EVN** in their names, respectively. For example, **Position\_5\_10.mrc**, **Position\_5\_10\_ODD.mrc**, and **Position\_5\_10\_EVN.mrc** are generated and associated with **Position\_5\_10.mdoc**.

### 5.2 Tomograms

Three tomograms are reconstructed for each mdoc file. They are odd, even, and full tomograms reconstructed from odd, even, and full tilt series. They are named after the corresponding tilt series with **\_Vol** appended in the file names such as **Position\_5\_10\_Vol.mrc**, **Position\_5\_10\_ODD\_Vol.mrc**, and **Position\_5\_10\_EVN\_Vol.mrc**. The odd and even tomograms are used to train in real time the model of CZII denoiser (**DenoisET**). Upon completion of model training, DenoisET then starts denoise full tomogram in real time.

### 5.3 Tilt angles and acquisition sequence

A tilt angle file, **Position\_5\_10\_TLT.txt** for example, also named after the corresponding mdoc file, contains two columns. The first column is tilt angles of the corresponding odd, even, and full tilt series. The second column shows the order of tilt images acquired during the data collection and is 1-based indices.

|        |    |
|--------|----|
| -21.01 | 15 |
| -18.01 | 12 |
| -15.01 | 11 |
| -12.02 | 8  |
| -9.02  | 7  |
| -6.02  | 4  |
| -3.02  | 3  |
| -0.03  | 1  |
| 2.98   | 2  |
| 5.98   | 5  |
| 8.98   | 6  |
| 11.98  | 9  |
| 14.98  | 10 |
| 17.98  | 13 |
| 20.98  | 14 |

**Fig. 4** An excerpt of the content in a TLT.txt file. The first column lists the tilt angles of images in a tilt series. The second column lists the indices that show the order of images acquired during data collection.



## 5.4 CTF estimation files

Two text files are generated per tilt series, one compatible with CTFFind4 format and one compatible with Imod. Again these two files are named after the corresponding mdoc file. The CTFFind4 compatible one is appended with \_CTF in its name. The Imod compatible one is further appended with \_CTF\_Imod in its file name.

## 5.5 Alignment file

Tomographic alignment parameters are saved in the **aln** file that saves both global and local alignment information. When the local alignment is enabled with -AtPatch, the aln file has two sections, one for global and one for local alignment. Otherwise, only global section is present. An aln file starts with the header section where each line starts with the number sign # as shown in the following example.

```
# AreTomo Alignment / Priims bprmMn
# RawSize = 4096 4096 31
# NumPatches = 16
# DarkFrame = 28 0 29.98
# DarkFrame = 29 0 32.98
# DarkFrame = 30 0 35.98
# AlphaOffset = -9.00
# BetaOffset = 0.00
```

**Fig.5** Header section of the aln file. DarkFrame entries exist only when dark images are detected.

**# RawSize** shows the image width and height in pixels and number of images in the tilt series.

**# NumPatch** indicates number of patches used in local alignment. **# DarkFrame** lists the dark tilt image that is excluded in the alignment. There are three entries of **# DarkFrame** rejected in this example. The first value is the zero-based image index in the raw tilt series, i.e. the one saved onto disk. The second value is reserved for future use. The third one is its tilt angle.

**# AlphaOffset** gives the detected tilt angle offset that is the actual tilt angle of sample at nominal zero degree. By default (**-TiltCor 0**) this angular offset is automatically measured unless users specify **-TiltCor -1**. The measured tilt angle offset is added to the nominal tilt angles only when **-TiltCor 1** is used. Please be **cautious** with this option when subtomogram average is the next processing. Adding the tilt angle offset computationally rotates the sample in the tomograms and thus changes the z coordinates of particles. In this case, the corrected tilt angles should be used for subtomogram averaging, not the nominal tilt angles.

**# BetaOffset** is zero and reserved for future implementation.

The next section is the **global alignment parameters** organized in 5 columns in a 10-column table. **SEC** column shows the zero-based indices of tilt images in the corresponding mrc file. **ROT** column lists the angle of tilt axis in degree relative to the y (vertical) axis of the unaligned tilt images. **TX** and **TY** are the translational shifts in pixel in unaligned image plane. **TILT** lists the tilt angles of images. The values of tilt angle are the sums of nominal tilt angles (microscope readings) and **# AlphaOffset**. Users can retrieve the nominal values by subtracting **# AlphaOffset** from the **TILT** column. Note that there are no entries for rejected dark images.

Local alignment data is presented right below **# Local Alignment**. This section contains a 7-column table as shown in the following example. Column 1 shows zero-based indices of image indices in the corresponding mrc file. Column 2 lists indices of patches in each tilt image. Columns 3 and 4 present the x and y coordinates of patch centers, respectively. Columns 5 and 6 show the measured translations in pixel in x and y directions, respectively. Note that the x and y coordinates are relative the image center. Column 7 indicates whether the measurement is reliable or not with 1 denoting reliable. The unreliable measurements are excluded in the local correction of sample motions. Importantly, local alignment data presented here represents the residual alignment errors after the global alignment. Therefore, the coordinates and translations are relative to the global translation aligned tilt images without rotation.

| #  | SEC | ROT     | GMAG    | TX      | TY       | SMEAN | SFIT | SCALE | BASE | TILT   |
|----|-----|---------|---------|---------|----------|-------|------|-------|------|--------|
| 0  |     | 84.0033 | 1.00000 | -26.113 | -137.927 | 1.00  | 1.00 | 1.00  | 0.00 | -45.02 |
| 1  |     | 84.0033 | 1.00000 | -20.276 | -79.669  | 1.00  | 1.00 | 1.00  | 0.00 | -42.01 |
| 2  |     | 84.0033 | 1.00000 | -11.406 | -112.696 | 1.00  | 1.00 | 1.00  | 0.00 | -39.02 |
| 3  |     | 84.0033 | 1.00000 | -20.730 | -79.387  | 1.00  | 1.00 | 1.00  | 0.00 | -36.01 |
| 4  |     | 84.0033 | 1.00000 | -16.668 | -114.443 | 1.00  | 1.00 | 1.00  | 0.00 | -33.01 |
| 5  |     | 84.0033 | 1.00000 | -14.818 | -68.274  | 1.00  | 1.00 | 1.00  | 0.00 | -30.02 |
| 6  |     | 84.0033 | 1.00000 | -22.440 | -92.762  | 1.00  | 1.00 | 1.00  | 0.00 | -27.01 |
| 7  |     | 84.0033 | 1.00000 | -8.281  | -52.114  | 1.00  | 1.00 | 1.00  | 0.00 | -24.02 |
| 8  |     | 84.0033 | 1.00000 | -23.853 | -74.965  | 1.00  | 1.00 | 1.00  | 0.00 | -21.01 |
| 9  |     | 84.0033 | 1.00000 | 9.043   | -36.378  | 1.00  | 1.00 | 1.00  | 0.00 | -18.01 |
| 10 |     | 84.0033 | 1.00000 | -31.795 | -65.251  | 1.00  | 1.00 | 1.00  | 0.00 | -15.01 |
| 11 |     | 84.0033 | 1.00000 | 2.055   | -22.090  | 1.00  | 1.00 | 1.00  | 0.00 | -12.02 |
| 12 |     | 84.0033 | 1.00000 | -16.796 | -49.033  | 1.00  | 1.00 | 1.00  | 0.00 | -9.02  |
| 13 |     | 84.0033 | 1.00000 | -3.999  | -21.075  | 1.00  | 1.00 | 1.00  | 0.00 | -6.02  |
| 14 |     | 84.0033 | 1.00000 | -18.390 | -93.674  | 1.00  | 1.00 | 1.00  | 0.00 | -3.02  |
| 15 |     | 84.0033 | 1.00000 | -5.265  | -36.435  | 1.00  | 1.00 | 1.00  | 0.00 | -0.03  |
| 16 |     | 84.0033 | 1.00000 | -0.940  | 46.147   | 1.00  | 1.00 | 1.00  | 0.00 | 2.98   |
| 17 |     | 84.0033 | 1.00000 | 5.306   | -17.315  | 1.00  | 1.00 | 1.00  | 0.00 | 5.98   |
| 18 |     | 84.0033 | 1.00000 | 0.000   | 0.000    | 1.00  | 1.00 | 1.00  | 0.00 | 8.98   |
| 19 |     | 84.0033 | 1.00000 | 3.946   | -11.816  | 1.00  | 1.00 | 1.00  | 0.00 | 11.98  |
| 20 |     | 84.0033 | 1.00000 | -6.927  | 11.334   | 1.00  | 1.00 | 1.00  | 0.00 | 14.98  |
| 21 |     | 84.0033 | 1.00000 | -0.986  | -6.024   | 1.00  | 1.00 | 1.00  | 0.00 | 17.98  |
| 22 |     | 84.0033 | 1.00000 | -12.766 | 7.682    | 1.00  | 1.00 | 1.00  | 0.00 | 20.98  |
| 23 |     | 84.0033 | 1.00000 | -1.761  | 4.175    | 1.00  | 1.00 | 1.00  | 0.00 | 23.98  |
| 24 |     | 84.0033 | 1.00000 | -9.999  | 19.771   | 1.00  | 1.00 | 1.00  | 0.00 | 26.98  |
| 25 |     | 84.0033 | 1.00000 | -6.673  | 7.913    | 1.00  | 1.00 | 1.00  | 0.00 | 29.98  |
| 26 |     | 84.0033 | 1.00000 | -10.817 | 10.732   | 1.00  | 1.00 | 1.00  | 0.00 | 32.98  |
| 27 |     | 84.0033 | 1.00000 | -6.994  | 41.365   | 1.00  | 1.00 | 1.00  | 0.00 | 35.98  |

**Fig. 6** An example of the global alignment parameters in an aln file. AreTomo3 uses only column SEC, ROT, TX, TY, and TILT. There are no entries for dark images since they are excluded from alignment process. TX and TY are the translational shift in pixel with respect to tilt images, not movies since tilt images are likely binned after motion correction.

| # | Local | Alignment |          |         |          |     |
|---|-------|-----------|----------|---------|----------|-----|
| 0 | 0     | -1272.49  | -1050.00 | -3.27   | -21.57   | 1.0 |
| 0 | 1     | -455.00   | -1020.88 | -4.86   | -20.73   | 1.0 |
| 0 | 2     | 570.77    | -1005.25 | -2.59   | -2.58    | 1.0 |
| 0 | 3     | 1598.27   | -974.22  | 682.84  | -2083.76 | 0.0 |
| 0 | 4     | -1555.11  | -490.34  | 7.90    | 28.16    | 1.0 |
| 0 | 5     | -503.87   | -437.69  | -3.67   | -9.53    | 1.0 |
| 0 | 6     | 519.96    | -311.99  | 1790.80 | 2056.45  | 0.0 |
| 0 | 7     | 1545.34   | -331.74  | 8.26    | 3.15     | 1.0 |
| 0 | 8     | -1597.80  | 52.57    | 0.58    | 1218.57  | 0.0 |
| 0 | 9     | -541.09   | 377.55   | -5.85   | -40.03   | 1.0 |
| 0 | 10    | 498.38    | 579.78   | 2.25    | 20.35    | 1.0 |
| 0 | 11    | 1529.58   | 516.46   | 7.23    | 41.23    | 1.0 |
| 0 | 12    | -1510.65  | 1345.49  | 3.71    | 15.67    | 1.0 |
| 0 | 13    | -584.90   | 1325.02  | 2.51    | 14.29    | 1.0 |
| 0 | 14    | 471.12    | 1292.97  | 1.65    | 20.05    | 1.0 |
| 0 | 15    | 1524.38   | 1268.09  | -0.90   | 16.34    | 1.0 |
| 1 | 0     | -1279.45  | -1112.42 | -3.43   | -26.24   | 1.0 |
| 1 | 1     | -461.72   | -1081.16 | -3.97   | -23.64   | 1.0 |
| 1 | 2     | 564.30    | -1063.30 | -2.09   | -9.49    | 1.0 |
| 1 | 3     | 1591.97   | -1030.66 | 586.73  | -2132.66 | 0.0 |
| 1 | 4     | -1556.77  | -505.24  | 4.27    | 14.12    | 1.0 |

**Fig. 7** An excerpt of the local alignment parameters in an aln file. This section appears only when users enable local alignment using -AtPatch. Columns 1 and 2 are the image and patch indices, respectively. Columns 3 and 4 are x and y coordinates of patch centers, respectively. Columns 5 and 6 are the translational shifts in x and y directions, respectively. Column 7 shows whether the measured shifts are reliable or not. The reliable measurements are labeled with 1.

## 6. Extra Command Line Options

### 6.1 Motion correction

**-Group** is a very useful option for measuring beam induced motion on challenging movies of low SNRs. This option decides how to divide movie frames or the rendered frames into non-overlapping groups. Beam induced motion is measured on the group sums and then interpolated/extrapolated to individual frames. This option takes two integers that are the group sizes of which the first one is for global measurement and the second for local measurement. The default values are 1 and 4. Users can selected different numbers but make sure the number of frames divided by the group size be larger than or equal to 3. The second number should be larger since individual patches have less SNRs than the full image and thus need more frames to increase SNRs.

**-DefectFile** is the same as that in MotionCor2/3. If there are fixed-pattern defects in micrographs, it is a good idea to compile a defect file and pass it to AreTomo3.

**-Gain** canload both mrc and gain files.

**-EerSampling** is the same as that in MotionCor2/3. It might be beneficial to combine -EerSampling 2 and -McBin 2 instead of -EerSampling 1 and -McBin 1 since working on the super-resolution frames may restore more information at lower tilts and in particular Tygress data collection.

**-McPatch** is the same as **-Patch** in MotionCor2/3. It enables local measurement and correction of beam induced motion.

## 6.2 Tomographic alignment

**-AlignZ** is the same as that in AreTomo/AreTomo2. The default value works well on thin and medium thick samples. Its value needs to be increased to be approximately the sample thickness in pixel.

**-VolZ** is the same as that in AreTomo/AreTomo2 and in pixel. It is recommended to reconstruct a slightly larger volume in z to cover the entire sample without clipping. This value is relative to the input tilt series. When **-AtBin** is used, the actual value z dimension is equal to the input value divided by the binning following **-AtBin**.

**-AtPatch** followed by two positive integers enables local tomographic alignment.

**-AtBin** enables floating-number binning by Fourier cropping to generate binned tomograms.

**-FlipVol** is the same as that in AreTomo/AreTomo2. **-FlipVol 1** rotates the volume around its x axis. The reconstructed volume is saved in a mrc file as a stack of xz slices. When this option is enabled, the mrc file contains a stack of xy slices.

**-Outlmod** is the same as that in AreTomo/AreTomo2.

**-DarkTol** is the same as that in AreTomo/AreTomo2. However, the implementation for detecting the dark images is different that considers the Tygress tilt series where the zero tilt image has more dose than the others.

**-TiltCor** is the same as that in AreTomo/AreTomo2. It is not recommended to enable this option (**-TiltCor 1**) when subtomogram average is part of your workflow since this option rotates the volume around its tilt axis. The z coordinates of particles picked from the rotated tomogram would be different from the coordinates in sample.

**-Outlmod** is the same as that in AreTomo/AreTomo2.

**-Wbp** is the same as that in AreTomo/AreTomo2

**-Sart** is the same as that in AreTomo/AreTomo2.

## 6.3 CTF estimation and deconvolution

**-PixSize** is needed to start CTF estimation. It is the pixel size of movies, not tilt series. AreTomo3 calculates the pixel size of tilt series based on this input value and **-McBin** value specified at command line.

**-Kv** is default to 300 kV. Users should change it if the data collection is done at different voltage.

**-Cs** is default to 2.7 mm. Users can choose a different value using this option.

**-AmpContrast** is default to 0.07.

**-ExtPhase** should be used when data collection is done with a phase plate. The option takes two values, one is the estimated phase shift and the second is the search range. They are both in degree.

**-CorrCTF 1** enables local CTF deconvolution of each tilt image. A tilt image is first divided into tiles. Each tile has its own CTF based on its location from the tilt axis. CTF deconvolution is done on each tile. Then CTF deconvolved tiles are put together to form the CTF deconvolved image.

## Reference

Zheng S, Brilot A, Cheng Y, Agard D. [Beam-Induced Motion Mechanism and Correction for Improved Cryo-Electron Microscopy and Cryo-Electron Tomography](#). Chapter 10, Cryo-Electron Tomography Structural Biology In Situ, Springer 2024.

Zheng S, Wolff G, Greenan G, Chen Z, Faas FGA, Bárcena M, Koster AJ, Cheng Y, Agard DA. [AreTomo: An integrated software package for automated marker-free, motion-corrected cryo-electron tomographic alignment and reconstruction](#). J Struct Biol X. 2022 May 10;6:100068. doi: 10.1016/j.yjsbx.2022.100068. PMID: 35601683; PMCID: PMC9117686.

## Technical Support

Shawn Zheng

Chan Zuckerberg Imaging Institute

[shawn.zheng@czii.org](mailto:shawn.zheng@czii.org)