TEM alignment in TrakEM2

by Albert Cardona (2016)

NOTE: This is an outdated protocol for TEM that is being kept for archival purposes. Most of the relevant information for the Harris Lab has been transferred to the new TrakEM2 Alignment Protocol. - Patrick Parker (May 2018)

Image settings: The images to which I applied the following settings are: 4096x4096 px, 16-bit, with a resolution of approx. 4x4x40 nm/px.

Backup XML files: At every step, save to XML and make a copy of the XML file. Will help you go back and restart at every point, saving time.

Step 1: disable mipmaps in TrakEM2

1. Right-click, select "Display - Properties" and uncheck the "use mipmaps" checkbox.
2. Also set the "snapshots mode" to "outlines" for best performance.

Step 2: import all image tiles from a text file that contains these columns:

1. file path, x, y, section index, width, height, min intensity, max intensity, type
2. Above, min and max can be e.g. 0 and 65535 for a 16-bit.
3. The type is the ImageJ's ImagePlus type: 0 for 8-bit, 1 for 16-bit, 2 for 32-bit, 3 for 8-bit color, 4 for RGB.
4. Having all settings in place will speed up the importing considerably.

Step 3: Adjust image filters

1. Select an image, right-click, and choose "Adjust image filters". Add these filters:
    - DefaultMinAndMax (no parameters)
    - NormalizeLocalContrast, with 400, 400, 3 std, center true, stretch true.
2. To try out which filters work best, first open a throw away TrakEM2 project, drag and drop a single image, and explore the parameters for best contrast. Particularly look for homogeneous contrast that removes the brightfield.
3. If your images contain banding artifacts, the rolling ball filter can remove them at the expense of some degradation of the high frequencies, which works wonders for registration.
4. Apply to a series of layers, to include all images.

Step 4: set the image mesh resolution for all images.

1. Right-click and choose "Transform - Adjust mesh resolution layer-wise".
2. For images of 4096x4096, a mesh resolution of 64 will suffice. This is the number of triangles (64x64) into which subdivide the image for rendering a non-linear transform as a bunch of local affine transforms that approximate it.

Step 5: compute the lens correction.

1. First, put a few, heavily overlapping images in a layer, and montage them using linear alignment by right-click and "Align - Montage all images in this layer". The [OK] means we change dialog to the next:

- initial gaussian blur: 1.6 px
- steps per scale octave: 4 (from 64 to 512 px)
- mininum image size: 64
- maximum image size: 512
- feature descriptor size: 8
- feature descriptor orientation bins: 8
- closest/next closest ratio: 0.92
- [OK]

- maximal alignment error: 20 px (5 times lower than the default)
- minimal inlier ratio: 0.1 (use tile-to-tile correspondences if they amount to 10% or more of the total amount of extracted features in those tiles)
- minimal number of inliers: 7 (at least 7 feature correspondences or discard that tile-to-tile relation)
- expected transformation: rigid
- ignore constant background: false
- [OK]

- desired transformation: rigid
- correspondence weight: 1.0
- regularize: false
- maximal iterations: 2000
- maximal plateau width: 200
- filter outliers: false
- mean factor: 3.0
- [OK]

- tiles are roughly in place: true (if you had reasonably accurate stage coordinates in the import file)
- consider largest graph only: true
- hide tiles from non-largest graph: true (if you only expect one single graph)
- delete tiles from non-largest graph: false (can be deleted later)
- [OK]

2. This should run pretty fast. You want to see that between 200 and 1000 features are extracted per tile, in the log window.

3. When done, select all images, right-click and choose "Patch - Lens correction" from the menu. In the dialogs, choose the exact same values for each parameter as in the successful montaging above. For the lens model part, I use the default values:

- power of polynomial kernel: 5
- lambda: 0.01
- tiles are roughly in place: true (we just montaged them, above)
- apply distortion correction: start: 1, end: 1. (can be applied to the other sections later)
- clear present transforms: true (there are any yet, but you might redo it)
- visualize distortion model: true (will open the quiver plot image)

4. When done, run the linear montaging again for this section. Choose one single tile, and push space bar to make it 50% transparent. Check that, particularly around the corners, images look good. Bear with TrakEM2 which might be a tiny bit slow at rendering images given that we aren't using mipmaps yet.

5. If the results are not good enough (e.g. larger than 1 or 2 pixels), increase the maximum image size to 1024 and try again. The high resolution will provide more of the high-frequency features, which can improve the results.

Step 6: apply the lens correction to all images.

1. Select one image that has the lens correction, right-click, and choose "Set coordinate transform of selected image layer-wise". Choose these parameters:
- existing coordinate transform: replace
- only the lens distortion correction: true (but there isn't any other)
- ... for the whole range of layers.

Step 7: montage all layers

1. Right-click and choose "Align - Montage multiple layers".
2. Choose the exact same parameters as above, but for the whole range of layers.

Step 8: inspection of montaging results.

1. A quick way to do it: zoom in a lot, and at the navigator in the lower left, move the now tiny red rectangle to one corner. Then browse through sections and see that tiles are arranged in a way that makes sense.
2. A more expensive way: enable mipmaps (right-click, "Display - properties", use mipmaps, then right-click, "Project - Regenerate all mipmaps").
3. Be sure to FIRST set the number of threads for generating mipmaps. If your images are smallish like 4kx4k, then use as many threads as CPUs your machine has, in the right-click "Project - Properties". The default is just 1!

4. Can take hours if there are many images. When done, and even while it is being done, you should be able to swiftly browse through and inspect all montages. This includes selecting some images and making them semi-transparent with the space bar, to check how well they overlap.

Step 9: elastic montaging

1. If montaging is not good enough, perform elastic montaging. If you did the lens correction, it is unlikely that you will ever have to do this.

Step 10: align all layers, regularizing to an affine.

1. Right-click and choose "Align - Align layers", choose least squares as the mode and the whole layer range (minus the layer that has the lens correction, which in any case you could remove anytime), and tell it to use visible images only (it is checked, "true" by default). Push [OK] and then:

Window #2:
- initial gaussian blur: 1.6 px
- steps per scale octave: 3 (e.g. 512, 1024, 2048)
- minimum image size: 512
- maximum image size: 2048
- feature descriptor size: 8
- feature descriptor orientation bins: 8
- closest/next closest ratio: 0.92
- clear cache: true
- feature extraction threads: 32
- [OK]

Window #3:
- maximal alignment error: 100 px (half the default; try 200 too)
- minimal inlier ratio: 0.2 (use tile-to-tile correspondences if they amount to 20% or more of the total amount of extracted features in those tiles)
- minimal number of inliers: 12 (at least 12 feature correspondences or discard that cross-layer relation)
- expected transformation: affine
- test multiple hypotheses: false (try true if you are not getting good results; but that's seldom the case)
- ignore constant background: false
- test maximally: 3 layers (to better constraint the registration by comparing to the adjacent section, the 2nd adjacent and the 3rd adjacent)
- give up after: 3 failures
- [OK]

Window #4:

- desired transformation: affine (default is rigid)
- regularize model: true (default is false)
- maximal iterations: 5000 (default is 1000, too little for large series)
- maximal plateau width: 200
- [OK]

Window #5:

- regularizer: rigid
- lambda: 0.1 (meaning, take 90% of the affine and 10% of the rigid)

2. This will run pretty fast if you had the mipmaps regenerated and available.

3. Write down the results of the alignment, which are the last thing that prints in the log window. The residual maximal error will be useful for estimating the search radius for the cross-correlation aspects of the elastic alignment.

Step 11: inspect the result of the linear section alignment

1. With mipmaps enabled this should be easy and fast. Look for sections that didn't align.
2. Then you can either align them manually, propagating the transform to the rest (see TrakEM2 manual), or align them automatically by increasing the number of features extracted or, most significantly, reducing the percent of correspondences ("minimal inlier ratio"), which can go down a lot when half the section is missing or has an enormous blob of precipitate.
3. Always propagate to other layers to keep the whole series aligned.

Step 12: rotate the sample to the preferred orientation.

1. Go to the first section, select all images, push 't', then rotate, then propagate the transform to all other sections.

Step 13: elastic alignment.

1. When the series of sections are reasonably well montaged, then right-click and select "Align - Align layers", and choose the "elastic" method. Also only for visible images. Use these values:

Elastic window #2:

- layer scale: 0.1 (10% of the actual scale. 0.05 can also work well)
- search radius: 200 (should be about twice as much as your residual maximal alignment error obtained by cross-section affine alignment regularized to rigid)
- block radius: 400
- resolution: 48 (is related to the mesh resolution of 64)
- minimal PMCC r: 0.1 (default is 0.6)
- maximal curvature ratio: 1000 (100x larger than default!)

- max second best r/best r: 0.9
- use local smoothness filter: true
- approximate local transform: affine (default is rigid)
- local region sigma: 200 px
- max local displacement (absolute): 100 px
- max local displacement (relative): 3.00
- layers are prealigned: true
- test maximally: 3 layers (again, compare with next, second-next, and third-next; works wonders for folds and cracks and partially missing data)
- [OK]

Elastic window #3:

- approximate transformation: rigid
- maximal iterations: 1000
- max plateau width: 200
- stiffness: 0.1
- maximal stretch: 2000 px
- maximal iterations: 8000 (8x larger than default)
- maximal plateau width: 200
- use legacy optimizer: true
- [OK]

2. Finally, when done -- the 8000 iterations will take many hours for a large project -- inspect the results and repeat if appropriate.

Image jitter: A good strategy when the images have a lot of jitter: first run the elastic alignment with a larger radius (like 500 px), and then run a second round with e.g. 200 px. This helps a lot.