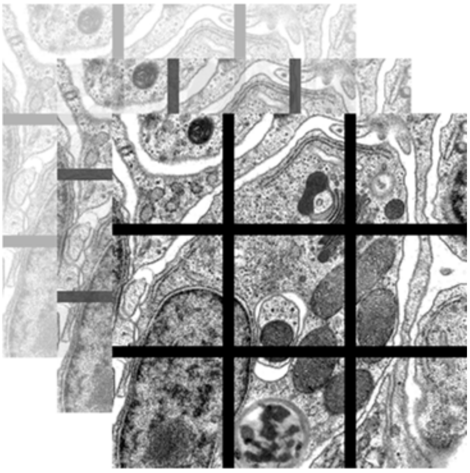


# MAGE3D programs

# User's manual

## Installation:

if you don't have Cygwin nor IMOD installed, use:

***Cygwin_2.897_x86_IMOD_4.10.34_win64_CUDA8.0_ImageMagick-7.0.2-7-Q16-x64-dll_MAGE3D_1.3.93.exe***

If you don't have IMOD installed, use:

***IMOD_4.10.34_win64_CUDA6.5_ImageMagick-7.0.2-7-Q16-x64-dll_MAGE3D_1.3.93.exe***

If you already have Cygwin and IMOD installed, use:

***ImageMagick-7.0.2-7-Q16-x64-dll_MAGE3D_1.3.93.exe***

If you want to upgrade montage to the last version, use:

***MAGE3D_1.3.93.exe***

During installation you will be asked to choose the working directory. It will be the place where you make montage, montage3D and Z alignment.

An example of files are present after the installation in the Montage, AlignZ and Montage3D directory to try the three programs.

## Chapter 1 - Montage:

The ***montage*** command will create automatically or semi-automatically a 2D montage from tile files. The command recognize TIFF, MRC or DM3-DM4 file format. It automatically creates montage in standard TIFF format for files below 4GB or BigTIFF format for files above 4GB. In option it can add a scalebar-infobox on the final montage, crop the montage, and correct its slope. The resulting montage keep the pixel size in its header. The program also creates a .log files.

*Email bug reports to jean dot daraspe at unil dot ch*

*And send the .log file.*

Place your tile files and the ***parameters.sh*** file in a folder. Place this folder in ***Montage*** directory. Ex:

***/Montage/Mouse-Liver/parameters.sh***
***/Montage/Mouse-Liver/tile_001.tif***
***/Montage/Mouse-Liver/tile_002.tif***
***/Montage/Mouse-Liver/tile_003.tif***
***etc…***

You can have many folders in ***/Montage/***, each of them should contain one ***parameters.sh*** file.
***/Montage/Mouse-Liver/***
***/Montage/ Mouse-Brain /***
***/Montage/ Human-Muscle /***
***etc…***

- For Gatan Digital Micrograph .dm3 or .dm4 files or TVIPS EMMenu .tif files, the ***parameters.sh*** files has to be fully informed. Ex:

***dimx=3 dimy=3 overx=25 overy=25 meanerror=10 scalebar=1 crop=1 slopecorrection=1***

- For SerialEM .mrc files (or ending with .st), you can put aside ***dimx***, ***dimy***, ***overx*** and ***overy*** and just inform ***meanerror***, ***scalebar***, ***crop*** and ***slopecorrection***. Ex:

***dimx=0 dimy=0 overx=0 overy=0 meanerror=10 scalebar=1 crop=1 slopecorrection=1***

- For FEI MAPS .tif files you can put aside ***dimx***, ***dimy*** and just inform ***overx***, ***overy***, ***meanerror***, ***scalebar***, ***crop*** and ***slopecorrection***. Ex:

***dimx=0 dimy=0 overx=10 overy=10 meanerror=10 scalebar=1 crop=1 slopecorrection=1***

Finally, open the terminal (Cygwin terminal in Windows) and just enter:

***montage***

The montages will be created automatically or semi-automatically (depending on ***meanerror*** value in ***parameters.sh,*** see Parameters description below) as tiff format in every folders

An example of tiles are installed in ***/Montage/Cells/.*** After installation you can try the montage program by running the ***montage*** command.

## Parameters description:

- The ***parameters.sh*** file description:

***dimx=3 dimy=3 overx=25 overy=25 meanerror=10 scalebar=1 crop=1 slopecorrection=1***

***dimx*** = Number of tiles in X.

***dimy*** = Number of tiles in Y.

***overx*** = Overlap in X (in %).

***overy*** = Overlap in Y (in %).

***meanerror*** = (in pixel) Accepted mean error in the final montage. This value is a quality control value. It's a trigger to use or not Midas to correct misalignments.

Ex: if your montage obtain a mean error of 4 (it means that you just have few errors in the alignment for a big montage) and your ***meanerror*** in ***parameters.sh*** is 10, your montage will be directly created.

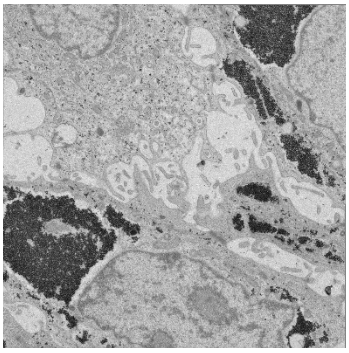
If your montage obtains a mean error above 10, the program ask you if you want to correct the misalignments with Midas. You can answer y or n.

If yes, Midas starts and you are invited to correct the misalignments. After that, the realignment restart automatically and your corrected montage will be created.

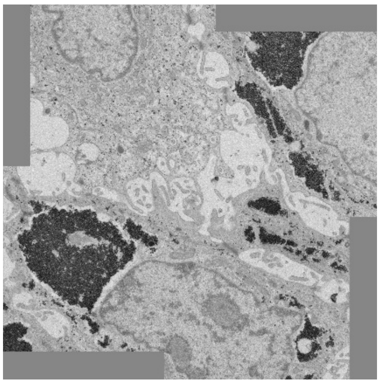
It means that you can force the use of Midas with a value of 0 (in that case you will be always asked to control the quality of your montage) or never use it with a very high meanerror, ex: 9999 (in that case you will be never prompted to correct the misalignment, and the montage will be created has it is, bad or not, automatically).

***scalebar*** = With a value of 1 the program create a scalebar-infobox at the bottom of the montage.
With a value of 0 there is no scalebar-infobox.
There is the possibility to have a signature in the infobox. This signature can be changed by editing the ***scalebar.py*** script (in /usr/local/IMOD/bin directory). At the beginning of the script you can change the ***signature*** = "EMF-UNIL" variable.

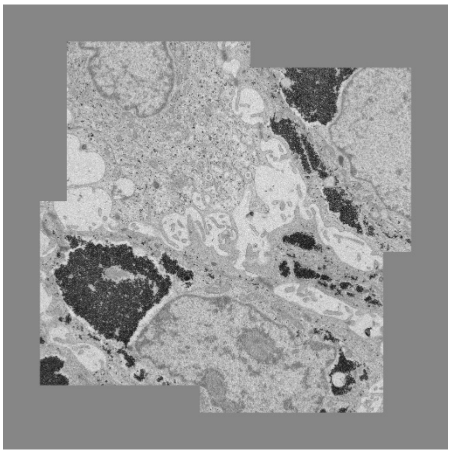
***crop*** = With a value of 1 the final montage will be cropped "inside".
With a value of 2 the final montage will be cropped "outside".
With a value of 0 the final montage will not be cropped.



***crop=1***



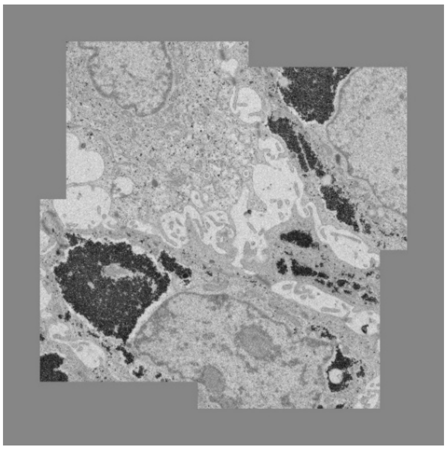
***crop=2***



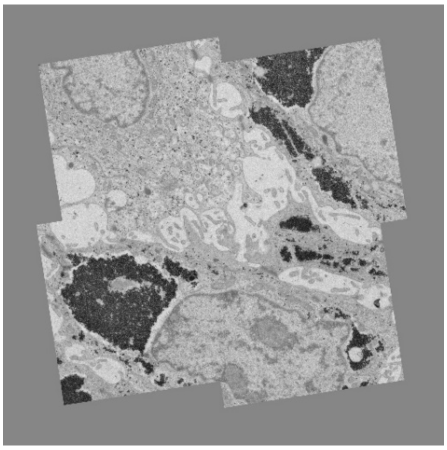
***crop=0***

***slopecorrection*** = With a value of 1 the program will correct the slope.
With a value of 0 there will be no slope correction.

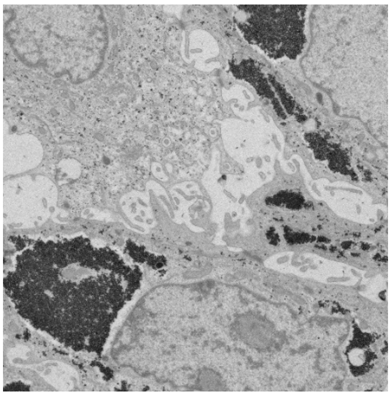
The slope correction will be applied only if the calculated slope is above 1% (slope is calculated as the mean between slope on each side of the montage).



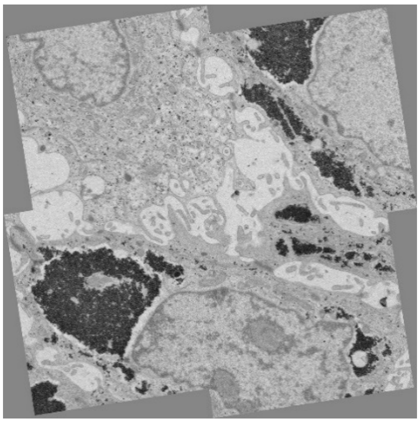
***slopecorrection=0***
***crop=0***



***slopecorrection=1***
***crop=0***



***slopecorrection=0***
***crop=1***



***slopecorrection=0***
***crop=2***

## Program sequence:

**1)** Find .tif, .mrc or .dm3 .dm4 files in **Montage** folder. Four different programs can be invoked depending on the finding.

***tiletif*** program is invoked (.tif from TVIPS EMMenu software) for .tif filenames ending with ***[0-9].tif** and not ending with ***_?-000.tif.***

***tilemrc*** program is invoked (.mrc files from SerialEM software) for filenames ending with *.**mrc*** or *.**st***.

***tiledm*** program is invoked (.dm3, dm4 files from Gatan Digital Micrograph) for filenames ending with ***.dm3*** or ***.dm4***.

***tilemaps*** program is invoked (.tif from FEI MAPS software) for .tif where filenames are structured like ***Tile_001-001-000_0-000.tif***.

**2)** Create list of tiles coordinates using values in ***parameters.sh*** (***dimx***, ***dimy***, ***overx*** and ***overy***)

***dimx***, ***dimy***, ***overx*** and ***overy*** are not necessary for .mrc files, the tiles coordinates and overlap are present in the .mrc extended header.

***dimx*** and ***dimy*** parameters are not necessary for .tif from FEI MAPS software, the tiles coordinates are present in the .tif filenames.

**3)** Create .mrc stack file from tiles sequence.

**4)** Run Blendmont program (IMOD) using .mrc file as input and .txt file as list of tiles coordinates. Blendmont produce a .mrc montage and a .err file which give the mean error of the blended montage and produce .ecd file with the edge correlation displacement.

**5)** Create automatically the resulting montage as a .tif file if mean error in the .err file is below the ***meanerror*** threshold from ***parameters.sh***

**6)** If the ***meanerror*** from .err file is above the threshold, the .mrc montage is loaded in 3dmod to visually check the montage quality.

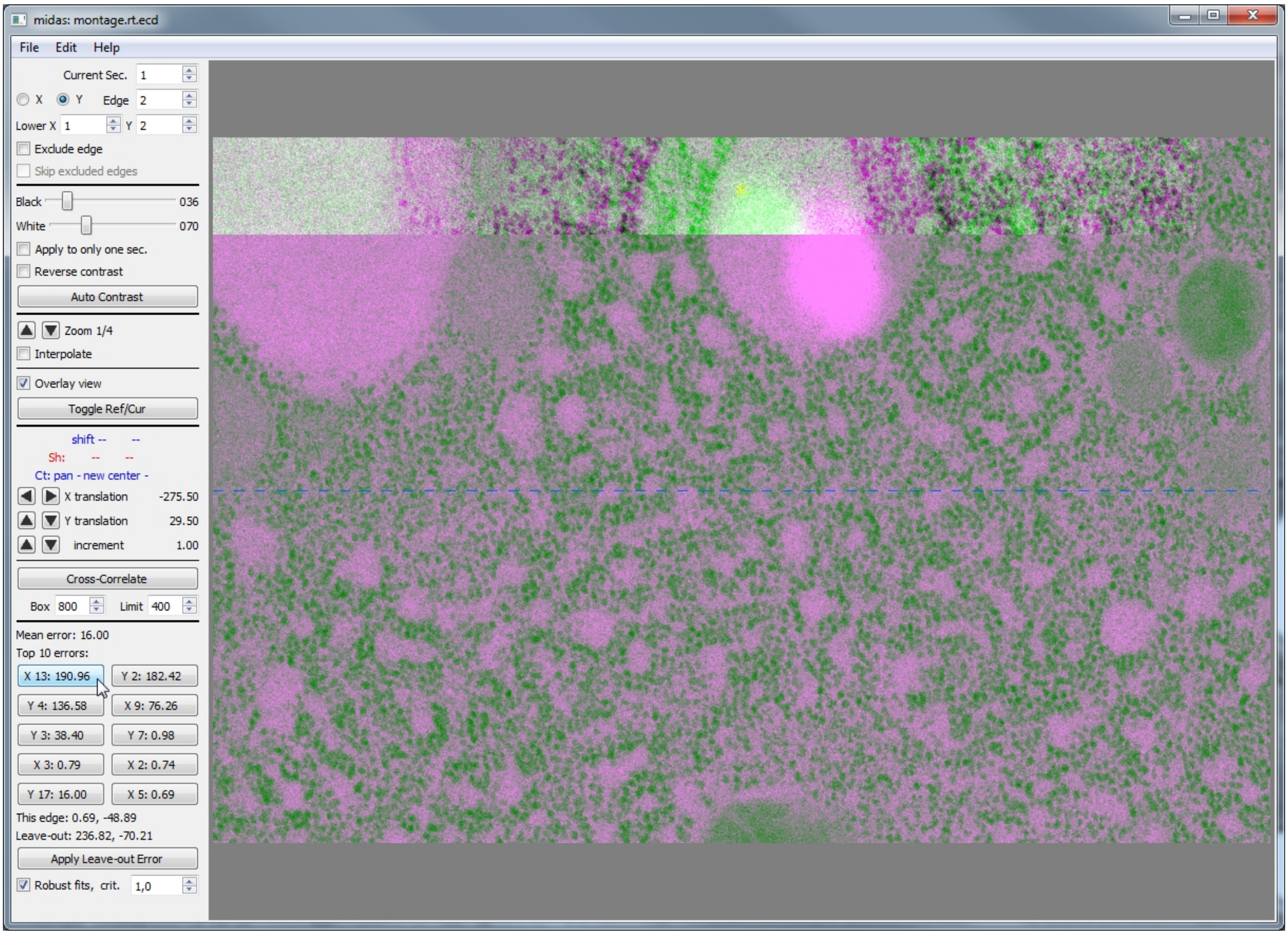
Answer to "***Do you want to adjust the montage alignments with Midas (y/n)?***"

If you answer "***n***": the resulting montage is created automatically as a .tif file.

If you answer "***y***": the montage is loaded in the Midas program (IMOD).

**7)** Midas will display the overlap zone between two tiles (named edges) and allow you to navigate and manually adjust the alignment between adjacent tiles in the montaged image. It give you different tools to help you in finding and fixing bad displacements.

a) First, adjust the contrast to clearly see your image using the ***Auto Contrast*** button or the ***Black*** and ***White*** sliders.

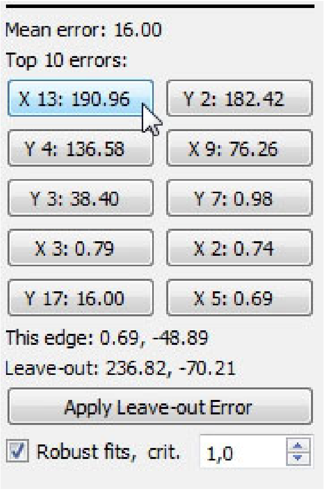


Current Sec. 1
○ X ● Y Edge 2
Lower X 1 Y 2
□ Exclude edge
□ Skip excluded edges
Black 036
White 070
□ Apply to only one sec.
□ Reverse contrast
Zoom 1/4
□ Interpolate
☑ Overlay view
shift -- --
Sh: -- --
Ct: pan - new center -
X translation -275.50
Y translation 29.50
increment 1.00
Box 800 Limit 400
Mean error: 16.00
Top 10 errors:
X 13: 190.96 Y 2: 182.42
Y 4: 136.58 X 9: 76.26
Y 3: 38.40 Y 7: 0.98
X 3: 0.79 X 2: 0.74
Y 17: 16.00 X 5: 0.69
This edge: 0.69, -48.89
Leave-out: 236.82, -70.21
☑ Robust fits, crit. 1,0

b) At the lower left of Midas window you have the ***Top 10 errors*** buttons which helps you to find the biggest errors. By shifting the image with the left mouse button you can manually fix the bad displacement and you will see that the ***Mean error*** value will decrease.

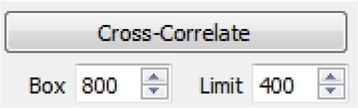
If an edge with a good alignment shows up as having a large error, use the following strategy:

- Continue to fix or exclude other edges with high errors. The error of the well-aligned edge may drop as other errors around it are resolved.
- If the well-aligned edge still has a high error after other edges with high errors have been fixed, examine all of the edges around the two pieces below and above this edge (using the buttons describe in part d), see below). You may find a small misalignment.
- If you still cannot find a misaligned edge, you can raise the fitting criterion (***Robust fits, crit.***) until the error becomes normal, or ignore the large error.



Mean error: 16.00
Top 10 errors:
X 13: 190.96 Y 2: 182.42
Y 4: 136.58 X 9: 76.26
Y 3: 38.40 Y 7: 0.98
X 3: 0.79 X 2: 0.74
Y 17: 16.00 X 5: 0.69
This edge: 0.69, -48.89
Leave-out: 236.82, -70.21
☑ Robust fits, crit. 1,0

c) Instead of manually shift the image, or in combination with, you can use the ***Cross-Correlate*** button (keyboard shortcut C) to help in finding the best shift. In that case, you may increase the box size in which the cross correlation take place.

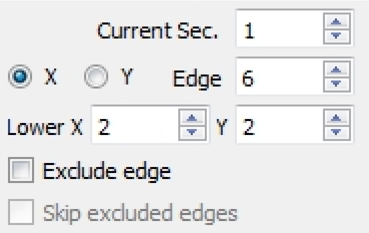


Box 800 Limit 400

d) One last possibility (the longer one) is to navigate between each edges, one by one, using the top left buttons and check every edges.

You can start with the X edges and start at Edge 1. Then press the keyboard shortcut A or B (shift + A, shift + B) to navigate between the X edges and fix the bad displacements.

Do the same with Y edges.



Current Sec. 1

● X ○ Y Edge 6

Lower X 2 Y 2

□ Exclude edge

□ Skip excluded edges

After fixing all the bad displacements, go to ***File***, then ***Save transforms*** and just close Midas.

**8)** The program run Blendmont again using .mrc file as input, .txt file as list of tiles coordinates and .ecd file to read new displacements in the overlap zones instead of computing correlations again.

**9)** The resulting montage is created as a .mrc file.

**10)** Depending on the stage displacement accuracy (or image shift-beam shift depending on the montage acquisition strategy), the overlap between tiles can be different from the overlap written in ***parameters.sh***.

If the real overlap is far lower than the overlap in ***parameters.sh*** the resulting montage area can be bigger than the area calculated by Blendmont, and the montage will be cropped, resulting in a loss of data on the edges of the montage.

To avoid this loss of data, a free space around the montage is allocated (by default 10% of the width and 10% of the height). This value can be change in /usr/local/IMOD/bin/montage or montage3d if needed, by changing the ***freespace***=10 variable at the beginning of the script.

At the end of the computation, this free space is cropped.

The option to control how the program crop the free space is given with the "crop" value in parameters.sh.

***crop=0*** disabled the cropping.

***crop=1*** is a tight cropping (every pixels from the freespace are removed around the montage, removing a small amount of the montage on the edges).

***crop=2*** is a lose cropping (all pixels from the montage are kept).

**11)** Depending on the stage displacement accuracy (or image shift-beam shift depending on the montage acquisition strategy), the resulting montage can have a "slope". ***slopecorrection*** is the option to correct this inclination.

***slopecorrection=0*** disabled the correction.

***slopecorrection=1*** activate the correction. The slope correction is only done if the slope is above 1% (this limit can be changed in autocrop file).

**12)** Create the resulting montage as a .tif file with filename ending with ***–Tiled.tif*** with or without a scalebar-infobox.

***scalebar=0*** no scalebar-infobox

***scalebar=1*** add the scalebar-infobox

***13)*** Remove temporary files.

## Chapter 2 - Z Alignment:

The ***alignz*** command will create semi-automatically a Z aligned stack from a set of files (like TEM serial sections or SEM serial block face images). The command recognize TIFF, or DM3-DM4 file format. It create an aligned stack in MRC format. It give the possibility to exclude some sections from the stack, to perform linear or warping transformation alignment (with or without boundary model), to invert the contrast and to crop the stack. At the end of the process if the result is not correct, one can choose between other alignment modes like local fitting, remove translation, rotation or size for the linear or warping transform. The resulting Z stack keep the pixel size in its header. The program also creates a .log files.

*Email bug reports to jean dot daraspe at unil dot ch*

*And send the .log file.*

Place your stack files in a folder. Place this folder in ***AlignZ*** directory. Ex:

***/ AlignZ /Mouse-Liver/Mouse_001.tif***
***/ AlignZ /Mouse-Liver/Mouse_002.tif***
***/ AlignZ /Mouse-Liver/Mouse_003.tif***
***etc…***

You can have many folders in ***/ AlignZ /***.

***/ AlignZ /Mouse-Liver/***
***/ AlignZ / Mouse-Brain /***
***/ AlignZ / Human-Muscle /***
***etc…***

Finally, open the terminal (Cygwin terminal in Windows) and just enter:

***alignz***

The Z aligned stack will be created semi-automatically as .mrc file format in every folders.

An example of files are installed in ***/AlignZ/Serial-Cells/.*** After installation you can try the alignZ program by running the ***alignz*** command.

## Program sequence:

**1)** Find .tif files in **AlignZ** folder and invoke ***alignslice*** program for .tif filenames ending with ****[0-9].tif***

**2)** Create .mrc stack file from file sequence and load it into 3dmod to check if sections has to be excluded.

**3)** Ask **"Do you want to exclude some sections from the stack (y,n)?"**

- If yes: **Enter sections numbers to be excluded. Ranges may be entered.**

Ex: **If you want to exclude sections from 1 to 5 and sections 6, 8 and 10, enter: 1-5,6,8,10**

- If No: just continue

**4)** Ask "**Do you want to perform alignment with linear transformation (1) or warping (2)?**"

Most images can be aligned adequately by applying general linear transformations. These transformations can include a shift, a rotation, a magnification (size change), and a stretch, all of which occur uniformly across an image. When the changes between two images are nonlinear (e.g., if distortions occurs differently in different areas), then it is necessary to use a nonlinear or warping transformation to align them. (After the first stack is produced, if the result is not correct, the program will ask you if you want to choose the other mode of alignment: linear if you choose first the warping, or warping if you choose first the linear)

- If **(1) Linear alignment**: the program will make a linear alignment by running **xfalign** to align images automatically using **xfsimplex** by searching for the full transformation as represented by semi-natural parameters (rotation, magnification, difference between X and Y axis stretch, difference between X and Y axis rotation.)

- If **(2) Warping** : the program will use **xfalign** to first find linear transforms with **xfsimplex** and then, find a warping transformation between successive images by running **tiltxcorr** to correlate overlapping image patches. To automatize the process the program calculate the size of the patches automatically. I limit the number of patches to 9. I admit that this automatic part can be a limit of the program, but if needed this value can be change in /usr/local/IMOD/bin/alignslice by changing the variable **warppatch=9** at the beginning of the script. The minimum number of patches should be 4.

Then the program ask "**Do you want to draw a boundary model (y) or continue without (n)?**" If the images contain areas not suitable for correlation, the best way to deal with this is with a boundary model. Draw one or more contours around the areas suitable for correlation. If this area changes, draw them on multiple sections as needed to capture the change in area.

(Look at http://bio3d.colorado.edu/imod/doc/3dmodguide.html, Modeling in the Zap Window paragraph to see how to draw contours in 3dmod)

- If Yes: the stack is loaded into 3dmod to allow you to draw your contours, then go to 'File' then 'Save Model' and close 3dmod."

- If No: just continue.

**5)** Then the program run **xftoxg** which takes a list of transformations (f) from each section to the previous one, and computes a list of xforms (g) to apply to each section to obtain a single consistent set of alignments using the global alignment mode. This mode will remove any consistent trends in the data stack, such as a progressive rotation, compression or shift. After the first stack is produced, if the result is not correct the program will ask you if you want to choose another mode (see paragraph 9)

**6)** Ask "**Do you want to inverse contrast (y,n)?**" (This option is usefull for data coming from serial block face imaging from scanning electron microscope).

**7)** Run **newstack** using .xg transformation file (from xftoxg) to apply to images and invert or not the contrast.

**8)** Ask "**Do you want to crop the stack (y,n)?**" and load the stack into 3dmod

If you want to crop the stack do the following:

- Select the rubberband (the dotted line rectangle in the Zap window)

- Position the rubberband by pressing the first mouse button at any corner of the desired area and dragging to the opposite corner.

The selection can be then repositioned and resized with first and second mouse buttons.

You can also crop the stack in Z:

- Go to the first slice that you want to keep and press the 'Lo' button.

- Then, go to the last slice that you want to keep and press the 'Hi' button.

- Finally, just exit 3dmod.... DON'T CLOSE THE ZaP WINDOW BUT ONLY THE 3DMOD WINDOW!

**9)** The first resulting stack is produced as an .mrc file and this message appear:

**If the alignment result doesn't seem to be correct, you can choose one of the following options or do nothing and quit (q):**

**- Local fitting (retain trends): align images to nearby ones but retain all progressive trends in position and size. Choose 1**

**- Remove trends in translation: Eliminate all shifts between images but retain other trends in the image. Choose 2**

**- Remove trends in translation and rotation: Eliminate shifts and rotations between images but retain trends in size. Choose 3**

**- Remove trends in translation, rotation and size: Eliminate shifts, rotations and size between images. Choose 4**

**- Proceed to a full realignment using warping transforms? Choose 5** (if your first choice was linear)

**- Proceed to a full realignment using linear transforms? Choose 6** (if your first choice was warping)

**- Do nothing and quit (q)**

- For answer 1 to 4 the program continue from paragraph 5 to the end.

- For answer 5 or 6 the program continue from paragraph 4 to the end.

**10)** The resulting .mrc stack is produced with the file name ending with:

- For linear transform only: **-linear.mrc**
- For linear transform with local fitting: **-linear-LF.mrc**
- For linear transform with removing translation: **-linear-RT.mrc**
- For linear transform with removing translation and rotation: **-linear-RTR.mrc**
- For linear transform with removing translation, rotation and size: **-linear-RTRS.mrc**
- For warping transform only: **-warping.mrc**
- For warping transform with local fitting: **-warping-LF.mrc**
- For warping transform with removing translation: **-warping-RT.mrc**
- For warping transform with removing translation and rotation: **-warping-RTR.mrc**
- For warping transform with removing translation, rotation and size: **-warping-RTRS.mrc**

**11)** Remove temporary files.

## Chapter 3 - Montage3D:

The ***montage3d*** command is a combination of ***montage*** followed by ***alignz***. First, it create automatically or semi-automatically a list of montage from tile files. Each created montage from the list are moved to the dedicated directory called ***MontageAligned***, and then the montages are processed for Z alignment. The process can be stopped at any time. At a new run, ***montage3d*** detect the presence of already created montages in ***MontageAligned*** and continue to the next one until the end. After that, the ***alignz*** command is called to make the Z alignment of montages. The resulting Z stack keep the pixel size in its header. The program also creates a .log files.

*Email bug reports to jean dot daraspe at unil dot ch*

*And send the .log file.*

Your list of montage has to be organized as follow. A list of folders containing your tiles has to be placed in the ***Montage3D*** directory. You only need one ***parameters.sh*** file placed in the ***Montage3D*** directory (and not in all subfolders). It means that the parameters in ***parameters.sh*** are the same for all montages, and <u>means that all the montages has to have the same number of tiles in X and Y</u>.

***/Montage3D/parameters.sh***
***/Montage3D/Mouse-Liver-001/tile_001.tif, tile_002.tif, tile_003.tif, etc…***
***/Montage3D/Mouse-Liver-002/tile_001.tif, tile_002.tif, tile_003.tif, etc…***
***/Montage3D/Mouse-Liver-003/tile_001.tif, tile_002.tif, tile_003.tif, etc…***
***/Montage3D/Mouse-Liver-004/tile_001.tif, tile_002.tif, tile_003.tif, etc…***
***etc…***

Finally, open the terminal (Cygwin terminal in Windows) and just enter:

***montage3d***

The montages will be created automatically or semi-automatically (depending on ***meanerror*** value in ***parameters.sh***, look at Parameters description in Montage in chapter 1) as tiff format in every folders and move them in ***MontageAligned*** directory.

Then, the ***alignz*** program will create the Z aligned stack semi-automatically as .mrc file format in the ***MontageAligned*** directory.

An example of tiles are installed in ***/Montage3D/Cells01, Cells02, Cells03/.*** After installation you can try the montage3d program by running the ***montage3d*** command.

## Program sequence:

The sequence is the same as ***montage*** and ***alignz***, look at chapter 1 and 2 for further details.

## Chapter 4 – List of program:

***alignlf***: Apply “local fitting” mode and call ***alignprompt.***

***alignlinear***: Align the image stack with linear transformation (with global alignment mode) and call ***alignprompt.***

***alignprompt***: Ask the user to choose between linear or warping transformation and keep or not trends in translations rotations or size if the previous result is not good.

***alignrt***: Apply “remove translation” mode and call ***alignprompt.***

***alignrtr***: Apply “remove translation and rotation” mode and call ***alignprompt.***

***alignrtrs***: Apply “remove translation, rotation and size” mode and call ***alignprompt.***

***alignslice***: Align the image stack with linear or warping transformation (with “global alignment” mode) and call ***alignprompt.***

***alignwarp***: Align the image stack with warping transformation (with “global alignment” mode) and call ***alignprompt.***

***alignz***: Find .tif or .dm3-dm4 files and call ***alignslice***.

***cleanupalign.py***: Remove temporary files in Alignz directory

***cleanupmontage.py***: Remove temporary files in Montage directory

***cleanupmontalign.py***: Remove temporary files in MontageAligned directory

***cleanuptilemont.py***: Remove temporary files in Montage3d directory

***montage***: Find .tif, .mrc or .dm3-dm4 files and call the corresponding tiling program (***tiledm***, ***tilemaps***, ***tilemrc*** or ***tiletif***)

***montage3d***: Find .tif, .mrc or .dm3-dm4 files and call ***tilemont***

***montpath***: PATH for Montage, Alignz, Montage3d and MontageAligned directories.

***scaleBar.py***: Add the scalebar and the infobox on montages.

***tiledm***: Tiling program for Gatan Digital Micrograph .dm3 or .dm4 files.

***tilemaps***: Tiling program for FEI MAPS .tif files.

***tilemont***: Tiling program for ***montage3d.***

***tilemrc***: Tiling program for SerialEM .mrc files (or ending with .st).

***tiletif***: Tiling program for TVIPS EMMenu .tif files.

***autocrop***: Crop and rotate the montage.