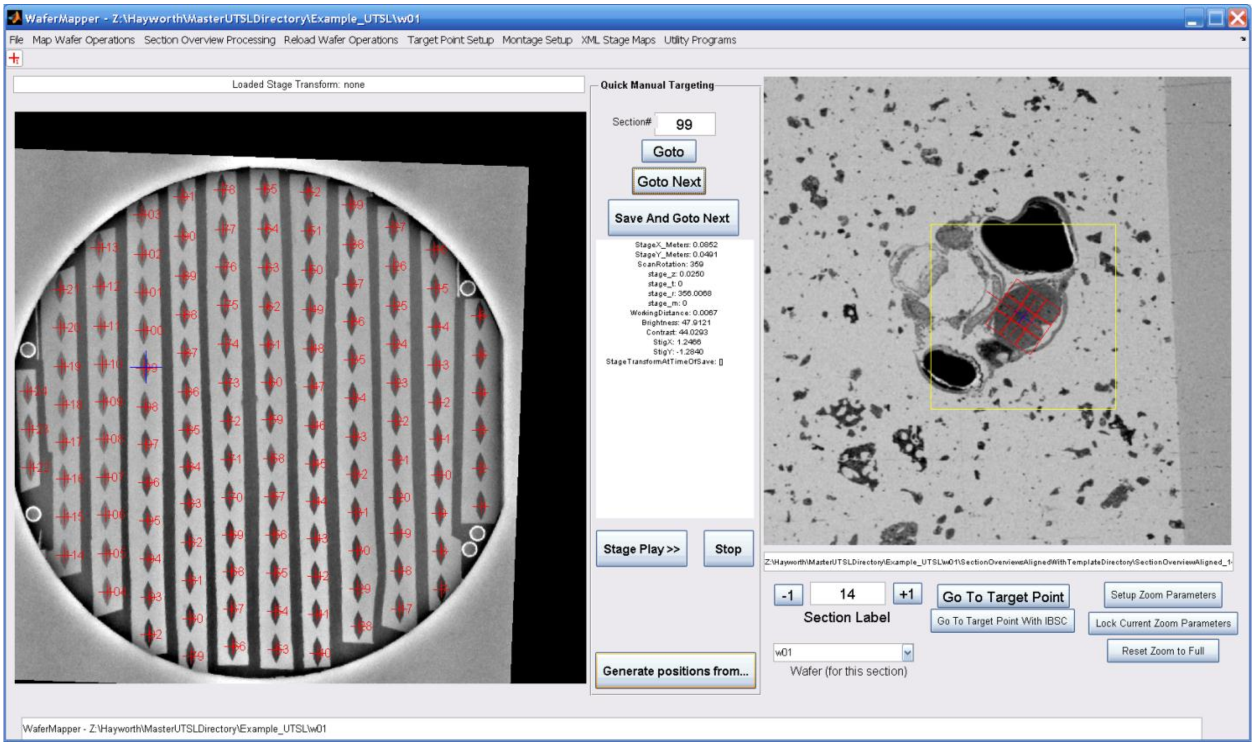
WaferMapper User’s Guide

(Beta Release 1.0, July 18th, 2011)

Contact: Kenneth Hayworth (braincad@hotmail.com)



Loaded Stage Transform: none

Section# 99

StageX_Meters: 0.0852
StageY_Meters: 0.0491
ScanRotation: 359
stage_z: 0.0250
stage_t: 0
stage_r: 356.0068
stage_m: 0
WorkingDistance: 0.0067
Brightness: 47.9121
Contrast: 44.0293
StigX: 1.2466
StigY: -1.2840
StageTransformAtTimeOfSave: []

Z:\Hayworth\MasterUTSLDirectory\Example_UTSL\w01\SectionOverviewsAlignedWithTemplateDirectory\SectionOverviewAligned_1

Section Label: 14

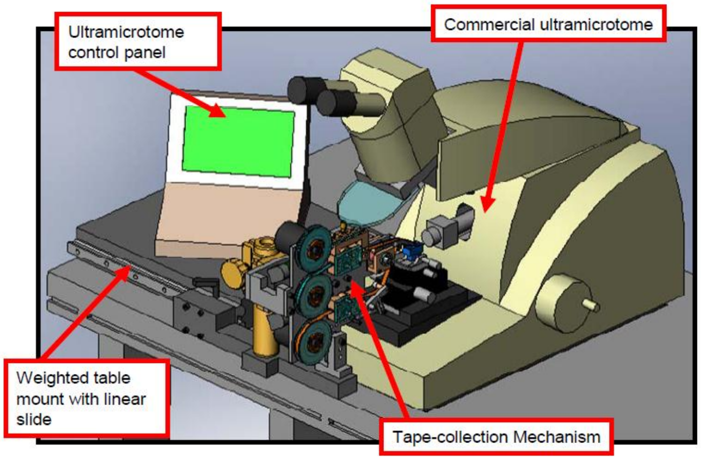
Wafer (for this section): w01

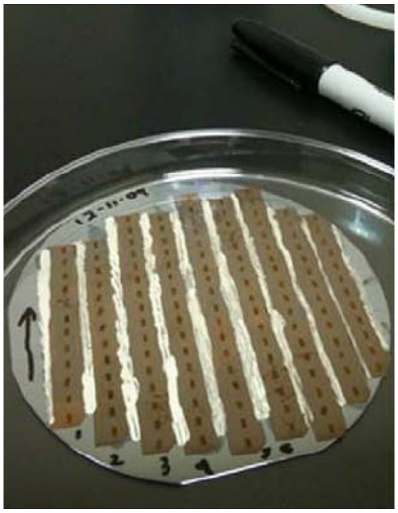
WaferMapper - Z:\Hayworth\MasterUTSLDirectory\Example_UTSL\w01

Table of Contents

1.0 Introduction

A single run of the ATUM (Automatic Tape-collecting UltraMicrotome) machine can produce tens of meters of tape containing many thousands of ultrathin sections. This tape is cut into strips and adhered to the surface of silicon wafers like the one shown below:

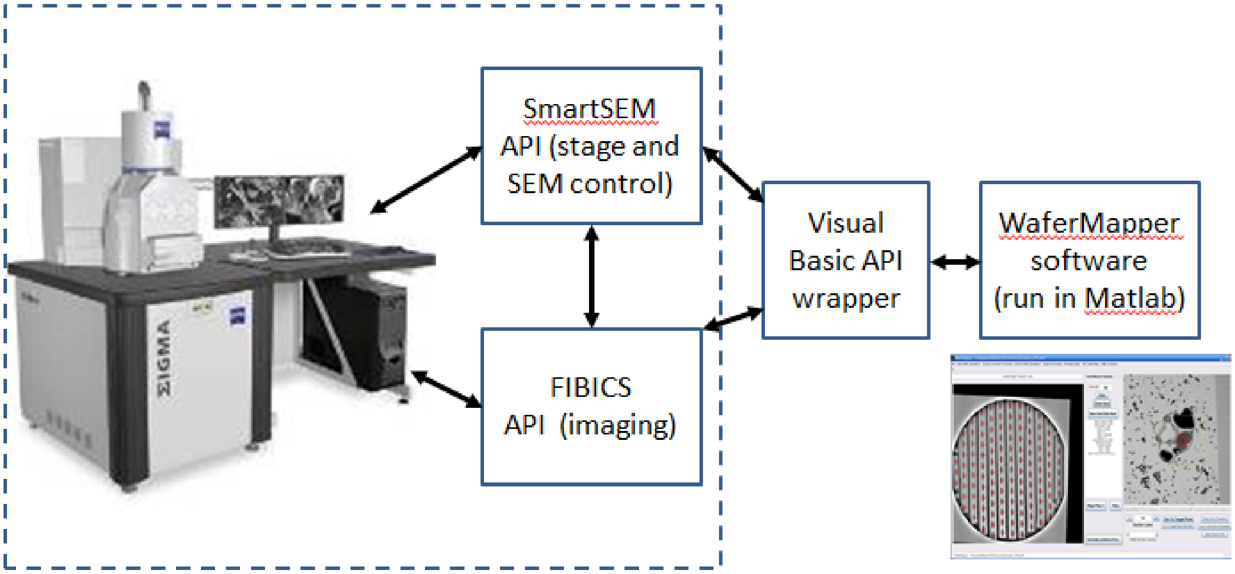

Ultramicrotome control panel
Commercial ultramicrotome
Weighted table mount with linear slide
Tape-collection Mechanism



A long run of the ATUM machine will produce dozens of these wafers. The collection of wafers from a single ATUM run is called an UltraThin Section Library (UTSL). To produce a volume image from this set of wafers each individual wafer needs to be loaded into the Scanning Electron Microscope (SEM) and then the SEM stage needs to be steered to the same spot of each of the hundred or more sections on the wafer and a montage of high-resolution tile images needs to be taken at this point.

To efficiently and effectively perform this imaging task software is needed that can automatically find all sections on each wafer and acquire section overview images of these. The software must then align these sections into one big low-resolution stack that the user can view to determine where to direct high-resolution imaging. The software should then allow the user to graphically define an imaging montage at a particular location within this aligned stack of sections. The software should then directly communicate to the SEM to accomplish this imaging in a precise and fully automated fashion. The WaferMapper software described here is designed to accomplish these goals.

The WaferMapper software is a Matlab GUI-based program that directly communicates with the Zeiss SmartSEM API and the Fibics imaging API as shown in the following diagram:

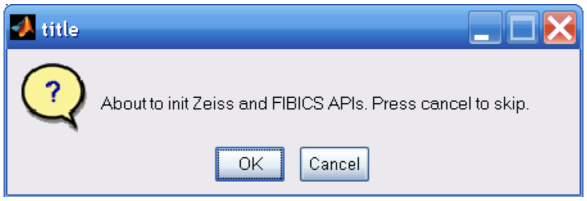

SmartSEM API (stage and SEM control)
FIBICS API (imaging)
Visual Basic API wrapper
WaferMapper software (run in Matlab)
ΣIGMA

1.1 Basic startup steps

Start Matlab on the Fibics computer. At the command prompt (>>) type the following:

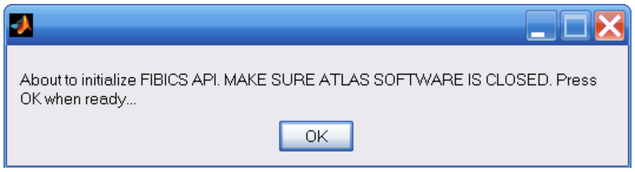
```
>> cd C:\WaferMapperCurrentReleaseDir
>> WaferMapper
```

The following window will pop up:



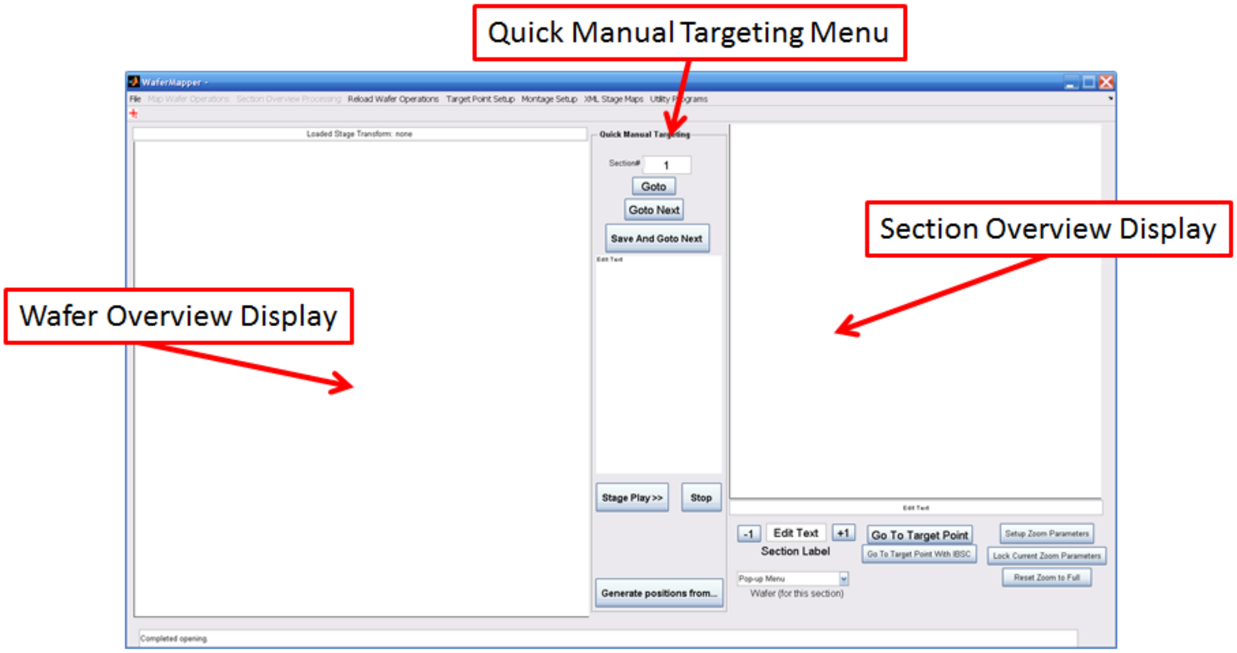
About to init Zeiss and FIBICS APIs. Press cancel to skip.

Press OK. (Note: If you wish to run the WaferMapper software on a computer not attached to the SEM then press cancel at the above prompt. All functionality not requiring the SEM will still be available.) Then the following message will pop up:



About to initialize FIBICS API. MAKE SURE ATLAS SOFTWARE IS CLOSED. Press OK when ready...

You must make sure that any previous instances of the ATLAS software are closed, and then press OK. A series of messages will pop up waiting for ATLAS to initialize, and then the main WaferMapper GUI will appear. Below is a diagram showing the main regions of the WaferMapper GUI:



Important program note: WaferMapper is a Matlab program built using the 'guide' framework. As such, if you click on something prematurely (e.g. if you choose a feature before the proper files have been created) and get an error message in the Matlab command window this will not stop the GUI. Usually you can simply ignore the problem generated by a wrong click and move on as if nothing occurred. Also, if you start a process and wish to end it before it naturally completes just press 'ctrl+c' while in the Matlab command window. This will stop the current process and generate an error message, but it will not stop the GUI and you can continue as before.

1.2 Creating a new UTSL and new wafers

WaferMapper is designed to organize your UTSL into a single directory – the "MasterUTSLDirectory". All sessions must begin by choosing (or creating) this directory. Only after this is chosen can you create a UTSL directory and only after that can you create directories for individual wafers.

In the WaferMapper GUI choose:

File-> **Open Master Dir for UTSLs**

Navigate to an existing MasterUTSLDirectory or create a new one called ‘MasterUTSLDirectory’. Now you can create a UTSL by:

File -> New UTSL

Call it ‘Example_UTSL’. Finally, you can create the first wafer in your UTSL with:

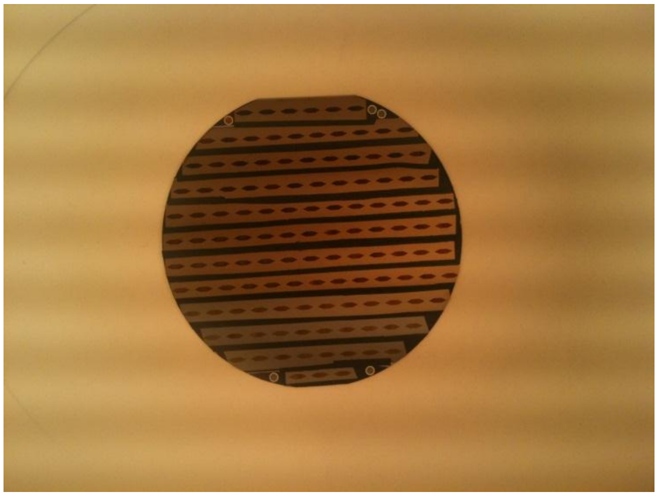
File –> New Wafer

Call it ‘w01’.

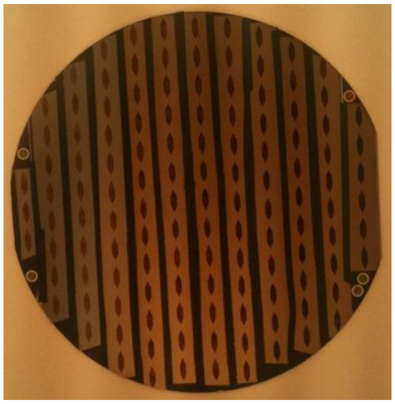
This will enable the ‘**Map Wafer Operations**’ menu.

1.3 Mapping a wafer using an optical image

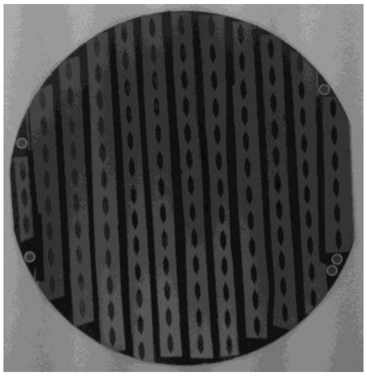
Using WaferMapper you can register an optical image of a wafer to the SEM stage coordinates and thus make navigation to the wafer’s sections as fast and intuitive as clicking the mouse on the image. The first step in this process is to take an image with an iPhone using the ‘top down pin hole box’ in the lab. This will create an image like this:



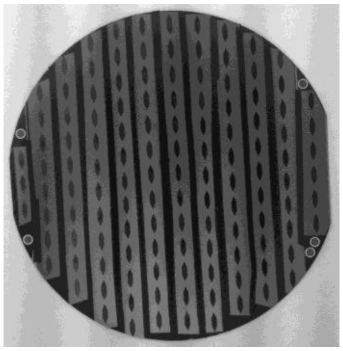
Use Fiji to crop and rotate this image as follows:



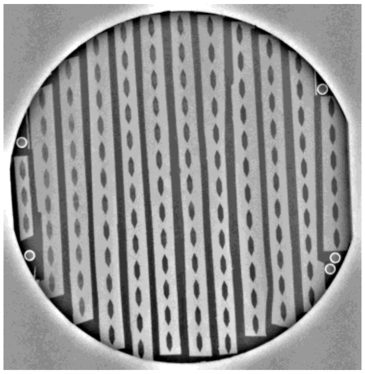
Then convert to a grayscale image using Fiji: **Image->Color->RGB to Luminance**



Adjust contrast using Fiji: **Image->Adjust->Brightness/Contrast…**

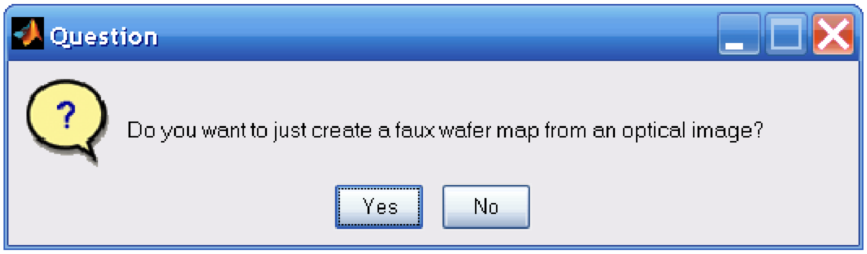


Then perform a mild highpass filtering to the image to reduce the shading differences across the image. The goal is to make sure the section's grayscale pixel values are all darker than the tape's grayscale values. Use Fiji: **Process->FFT->Bandpass Filter…** with parameters 'Filter large structures down to 50 pixels' and 'Filter small structures up to 1 pixel'. The result is:



This image must be saved as a ‘tif’ image file in order for WaferMapper to read it.

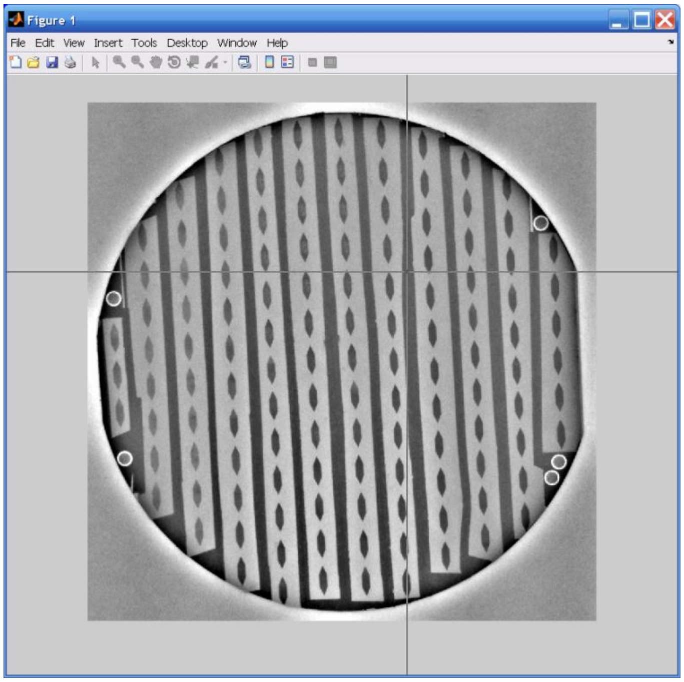
Now in the WaferMapper GUI choose **Map Wafer Operations -> Acquire Full Wafer Montage**. The following dialog box will pop up:



Question

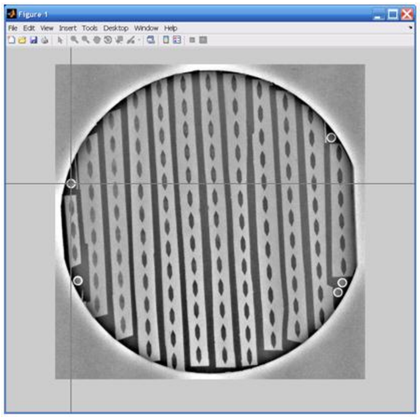
Do you want to just create a faux wafer map from an optical image?

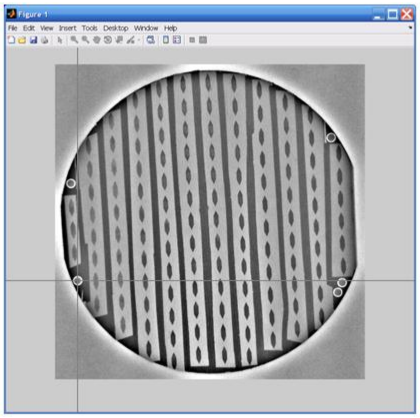
Choose ‘Yes’. A file dialog box will pop up for you to choose the filtered optical image you just created. Once you choose this file the image will pop up in a new Matlab figure window and a message box will inform you that you are to “Drive microscope to a fiducial then click that fiducial location in the image.” Clicking OK on the dialog box produces a set of crosshairs on the image like this:

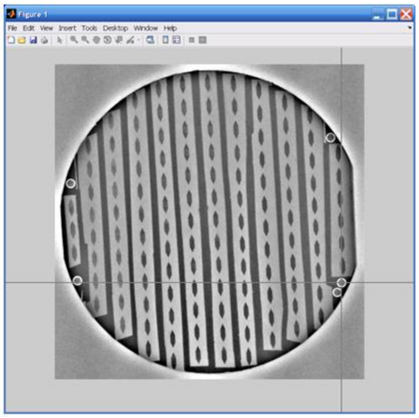


Move the mouse over to the SEM screen and drive the SEM using the joystick to the position of the first fiducial. Use a ctrl+tab with backlash to get the SEM right dead center of the first fiducial. Then move the mouse back to the Matlab figure and carefully position the crosshairs over the same fiducial in the optical image and click. A dialog box will pop up asking for confirmation, click yes.

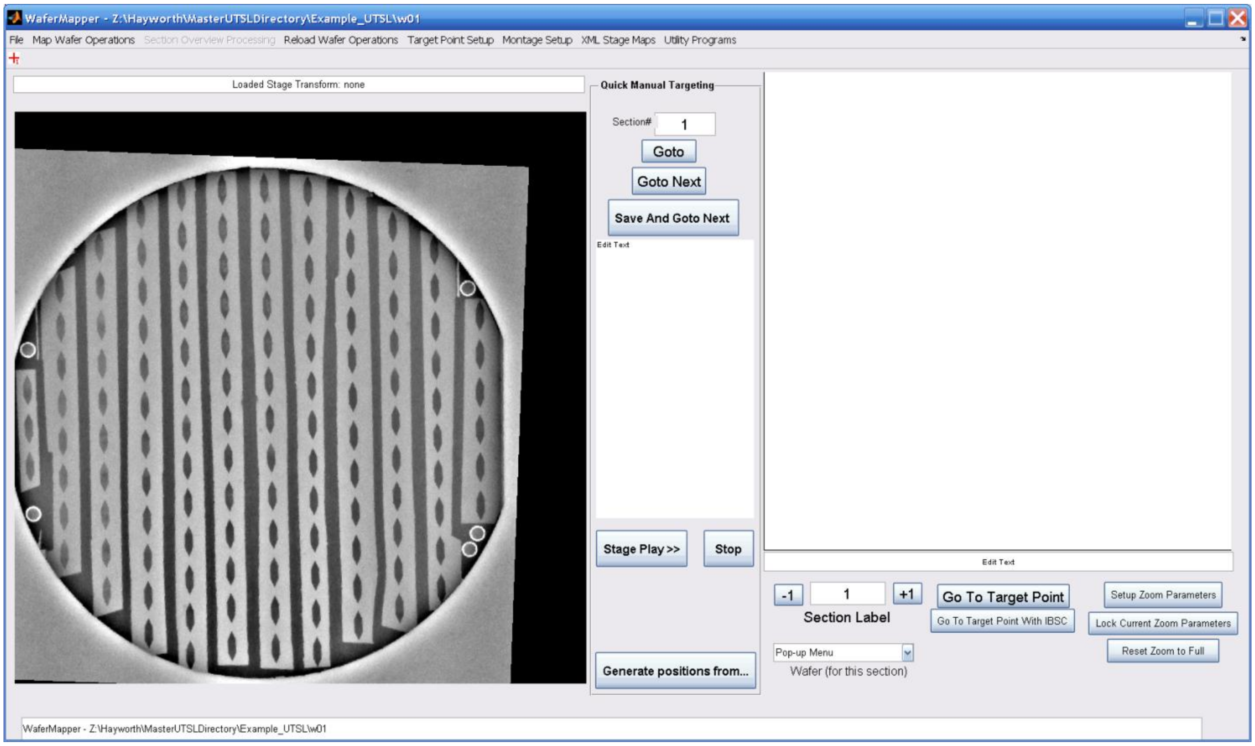
Repeat this for a total of at least three fiducials as follows:







Press ‘esc’ after the final fiducial is done. The program will pause for a moment to generate a transformed and aligned image and then display this in the WaferMapper GUI’s Wafer Overview display as follows:



WaferMapper - Z:\Hayworth\MasterUTSLDirectory\Example_UTSL\w01

Loaded Stage Transform: none

Section# 1

Section Label 1

WaferMapper - Z:\Hayworth\MasterUTSLDirectory\Example_UTSL\w01

To verify that this mapping actually worked choose **Map Wafer Operations -> Free View** . This will present crosshairs in the WaferMapper GUI's Wafer Overview Figure. Left click the crosshairs anywhere within the Wafer Overview Figure and the SEM should immediately drive the stage to that position. When you are finished verifying the mapping press 'esc'.

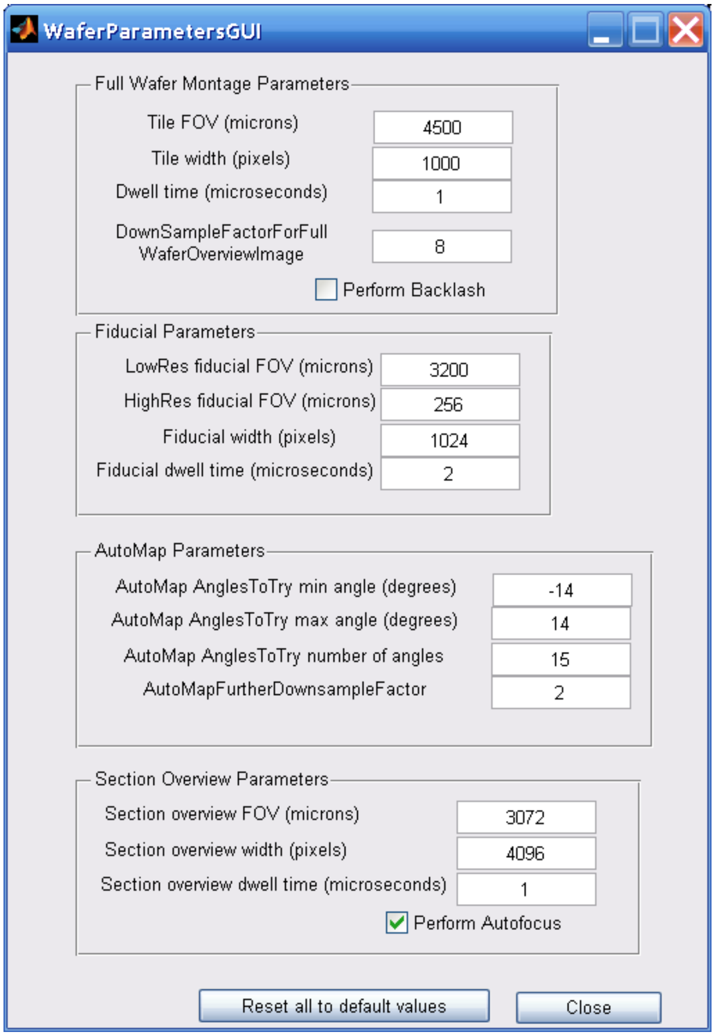
Note: If you need to redo this or any other step in the wafer mapping process it usually requires deleting (or renaming) directories in the wafer subdirectory you are working in. Then simply choose the menu option again. This prevents you from having to restart the whole process.

1.4 Mapping a wafer using an SEM montage

Instead of importing and registering an optical image of the wafer, you can also use WaferMapper to take a full wafer montage directly within the electron microscope. Advantages of this method are that it provides direct contrast of the tissues within sections and it provides better resolution for all subsequent steps. The main disadvantage is speed.

Make sure that the UTSL and the wafer directories are created as described in section 1.2 above. Also make sure that the microscope is put at a working distance appropriate for large field of view imaging with the backscatter detector. 9.5mm working distance works well in the Zeiss Sigma as it allows for montage tiles to be up to 5mm wide and these images are relatively free of edge brightness artifacts that occur at shorter working distances.

Choose **Map Wafer Operations -> Wafer Parameters**. This opens up the following dialog box:



WaferParametersGUI

Full Wafer Montage Parameters

Tile FOV (microns) 4500

Tile width (pixels) 1000

Dwell time (microseconds) 1

DownSampleFactorForFull WaferOverviewImage 8

☐ Perform Backlash

Fiducial Parameters

LowRes fiducial FOV (microns) 3200

HighRes fiducial FOV (microns) 256

Fiducial width (pixels) 1024

Fiducial dwell time (microseconds) 2

AutoMap Parameters

AutoMap AnglesToTry min angle (degrees) -14

AutoMap AnglesToTry max angle (degrees) 14

AutoMap AnglesToTry number of angles 15

AutoMapFurtherDownsampleFactor 2

Section Overview Parameters

Section overview FOV (microns) 3072

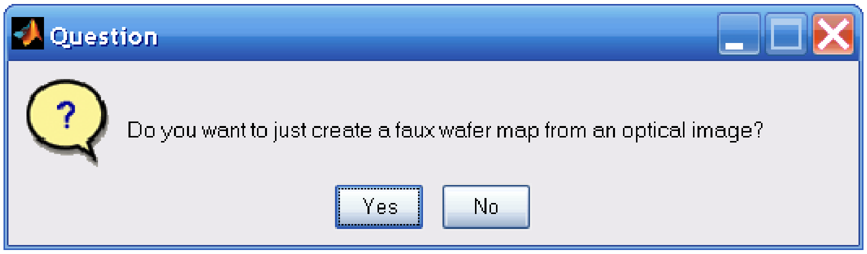
Section overview width (pixels) 4096

Section overview dwell time (microseconds) 1

☑ Perform Autofocus

We are interested here in the first group of edit boxes labeled Full Wafer Montage Parameters. The default value for Tile FOV (field of view) is 4.5mm (4500 microns) and the default tile width in pixels is 1000. These are appropriate values to use for the full wafer montage – greater pixel density may cause memory problems later on, and smaller tiles FOV will take too long to image. Close the dialog box.

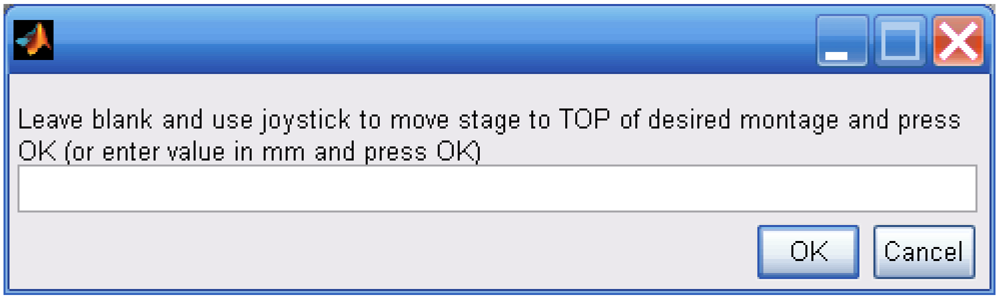
Choose **Map Wafer Operations -> Acquire Full Wafer Montage**. The following question box pops up:



Question

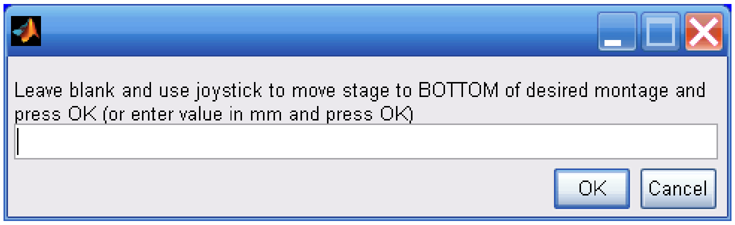
Do you want to just create a faux wafer map from an optical image?

Choose 'No'. A warning pops up saying to check the backscatter detector to make sure that it does not produce white corners at the chosen field of view. A few other warning messages pop up after this. Make sure to adjust the focus and contrast so that the sections show up clearly against the tape background. After these the following dialog pops up:

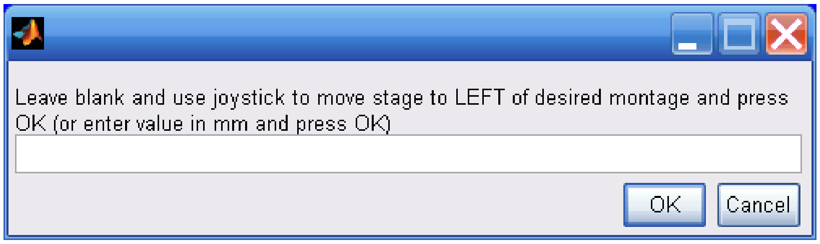


Leave blank and use joystick to move stage to TOP of desired montage and press OK (or enter value in mm and press OK)

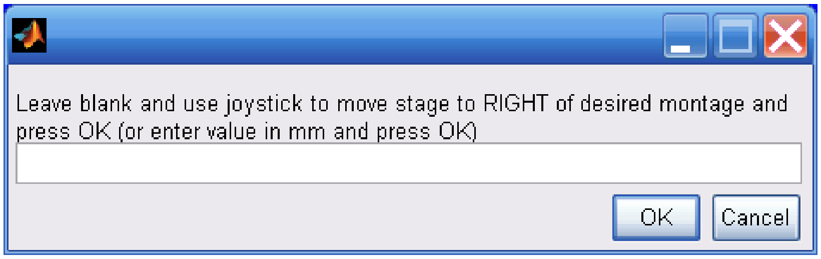
This dialog is asking you to specify the topmost extent of the full wafer montage (to make sure that all sections are contained within the final montage). Using the magnification dial, set the field of view of the SEM to around 4.5mm and use the joystick to drive the stage to the top of the wafer. Go to the strip that sticks out the highest and make sure that its top edge is visible in the view then press the 'OK' button on this dialog. Additional dialogs pop up prompting you to repeat this process for the Bottom, Left, and Right sides as well.



Leave blank and use joystick to move stage to BOTTOM of desired montage and press OK (or enter value in mm and press OK)

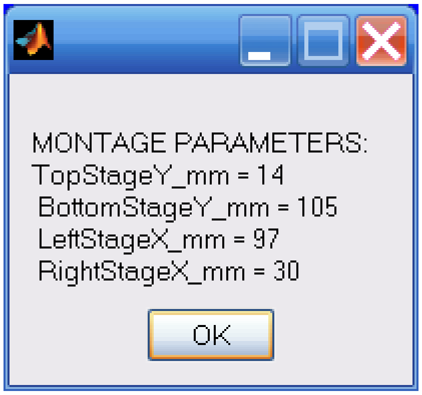


Leave blank and use joystick to move stage to LEFT of desired montage and press OK (or enter value in mm and press OK)



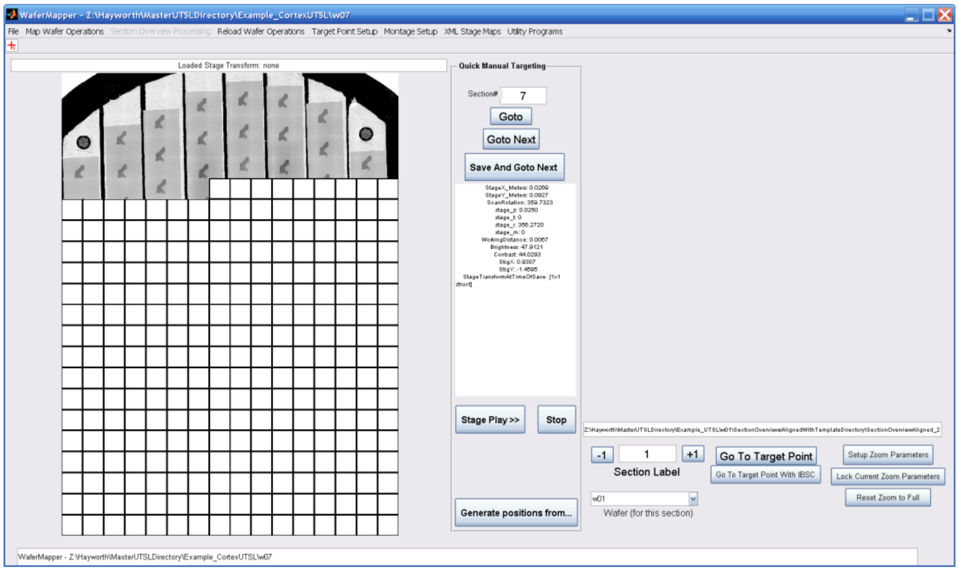
Leave blank and use joystick to move stage to RIGHT of desired montage and press OK (or enter value in mm and press OK)

After these are complete the following message pops up:



MONTAGE PARAMETERS:
TopStageY_mm = 14
BottomStageY_mm = 105
LeftStageX_mm = 97
RightStageX_mm = 30

Click ‘OK’. The full wafer montage process will now proceed automatically for about an hour and its progress will be displayed graphically in the WaferMapper GUI:

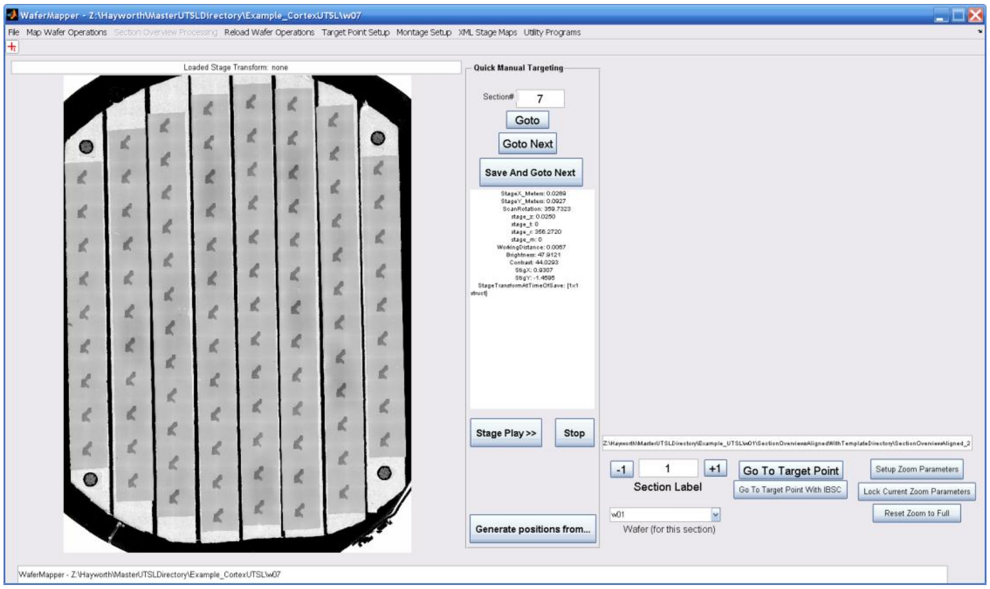


StageX_Meters: 0.0269
StageY_Meters: 0.0927
ScanRotation: 359.7323
stage_z: 0.0250
stage_t: 0
stage_r: 356.2720
stage_m: 0
WorkingDistance: 0.0067
Brightness: 47.9121
Contrast: 44.0293
StigX: 0.9307
StigY: -1.4695
StageTransformAtTimeOfSave: [1x1 struct]

Z:\Hayworth\MasterUTSLDirectory\Example_UTSL\w01\SectionOverviewsAlignedWithTemplateDirectory\SectionOverviewAligned_2

WaferMapper - Z:\Hayworth\MasterUTSLDirectory\Example_CortexUTSL\w07

At the end of this process the WaferMapper GUI will look like this:



Loaded Stage Transform: none

Section# 7

```
StageX_Meters: 0.0289
StageY_Meters: 0.0927
ScanRotation: 359.7323
stage_z: 0.0250
stage_t: 0
stage_r: 356.2720
stage_m: 0
WorkingDistance: 0.0067
Brightness: 47.9121
Contrast: 44.0293
StigX: 0.9307
StigY: -1.4595
StageTransformAtTimeOfSave: [1x1
struct]
```

Z:\Hayworth\MasterUTSLDirectory\Example_UTSL\w01\SectionOverviewsAlignedWithTemplateDirectory\SectionOverviewsAligned_2

Section Label: 1

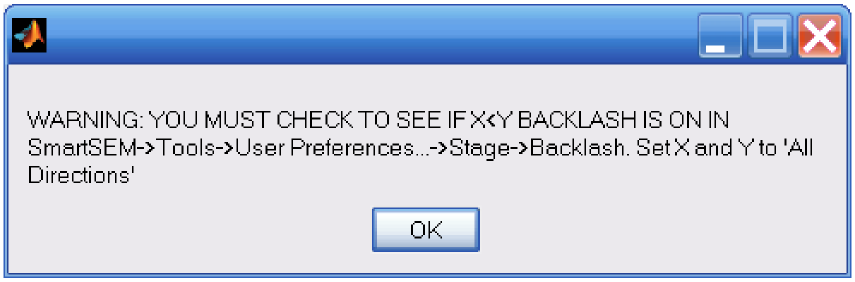
Wafer (for this section): w01

WaferMapper - Z:\Hayworth\MasterUTSLDirectory\Example_CortexUTSL\w07

1.5 Acquiring SEM fiducial images

The next step in the wafer mapping process is to acquire a set of low resolution and high resolution fiducial images. These are SEM images of fiducials (typically TEM grids that are mounted on the wafer) where the exact stage position of the image is recorded along with the image itself to be used for automatically reloading and realigning the wafer for use in another session (see section 1.15 below).

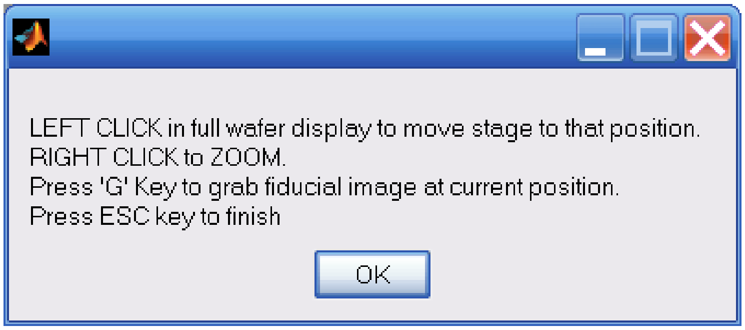
Choose **Map Wafer Operations -> Acquire Low Res. Fiducials** . The following message box will appear:



WARNING: YOU MUST CHECK TO SEE IF X<Y BACKLASH IS ON IN SmartSEM->Tools->User Preferences...->Stage->Backlash. Set X and Y to 'All Directions'

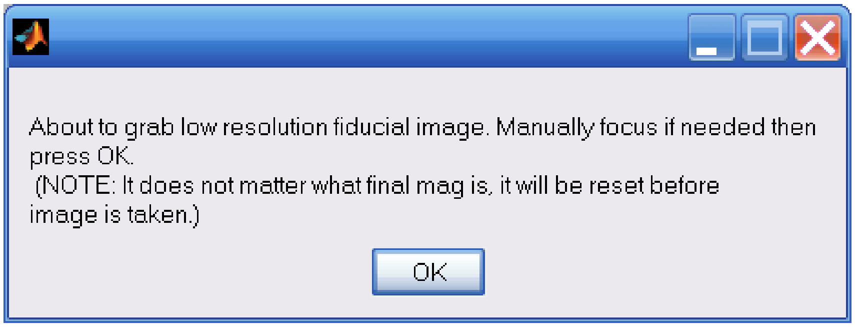
This is to remind you that the SEM backlash must be enabled to provide accurate positioning of the stage which is crucial during the acquisition of fiducials. To check and see if backlash is operating do a ctrl+tab click in the SmartSEM window. If it does not do a backlash following this ctrl+tab click then turn on backlash in SmartSEM Tools->User Preferences-> Stage -> Backlash. Also check Stage -> Navigation -> More Stage Functionality -> Backlash (make sure 'On' is checked).

Clicking OK to the message box produces the following message box:



LEFT CLICK in full wafer display to move stage to that position.
RIGHT CLICK to ZOOM.
Press 'G' Key to grab fiducial image at current position.
Press ESC key to finish

Click OK. Left click with the crosshairs on the first fiducial. This will drive the stage to approximately the correct position. Use ctrl+tab in the SmartSEM window to get to the exact center of the fiducial (there should be a backlash following this movement). Now go back to the WaferMapper GUI's Wafer Overview Display (such that the crosshairs reappear) and press 'G'. The following message box will pop up:



About to grab low resolution fiducial image. Manually focus if needed then press OK.
(NOTE: It does not matter what final mag is, it will be reset before image is taken.)

Check focus and then press OK. Fibics will have acquired this image and the image and a position data file will have been saved in the wafer subdirectory called 'LowResFiducialsDirectory'.

Repeat this process three to four times and then press 'esc' to remove crosshairs.

Now choose **Map Wafer Operations -> Acquire High Res. Fiducials** . And obtain three to four high res fiducial images via the same process.

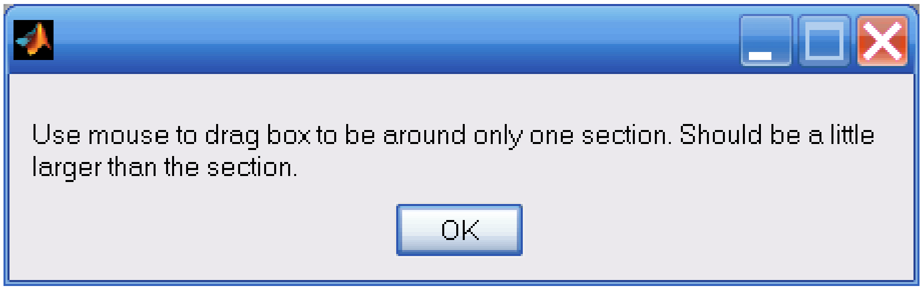
Note: The default image size for low res fiducials is 3.2mm wide FOV, and for high res fiducials is 256micron wide FOV. Make sure that there is sufficient contrast and detail in the fiducials chosen that they can be aligned robustly during reloading operations.

1.6 AutoMapping of sections

The next step in the wafer mapping process is to tell the program what a section looks like. This step is different depending on whether the full wafer overview was obtained optically or via the SEM montage.

If optical wafer image:

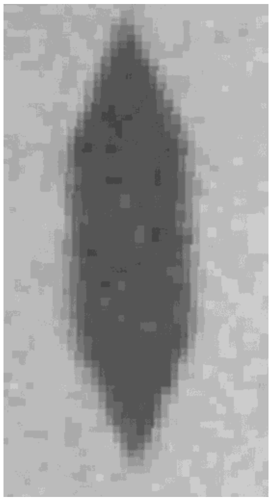
Choose **Map Wafer Operations -> Acquire Example Section Image** . If using an optical wafer map the new Matlab figure will popup containing the optical wafer image, and the following message box will appear:



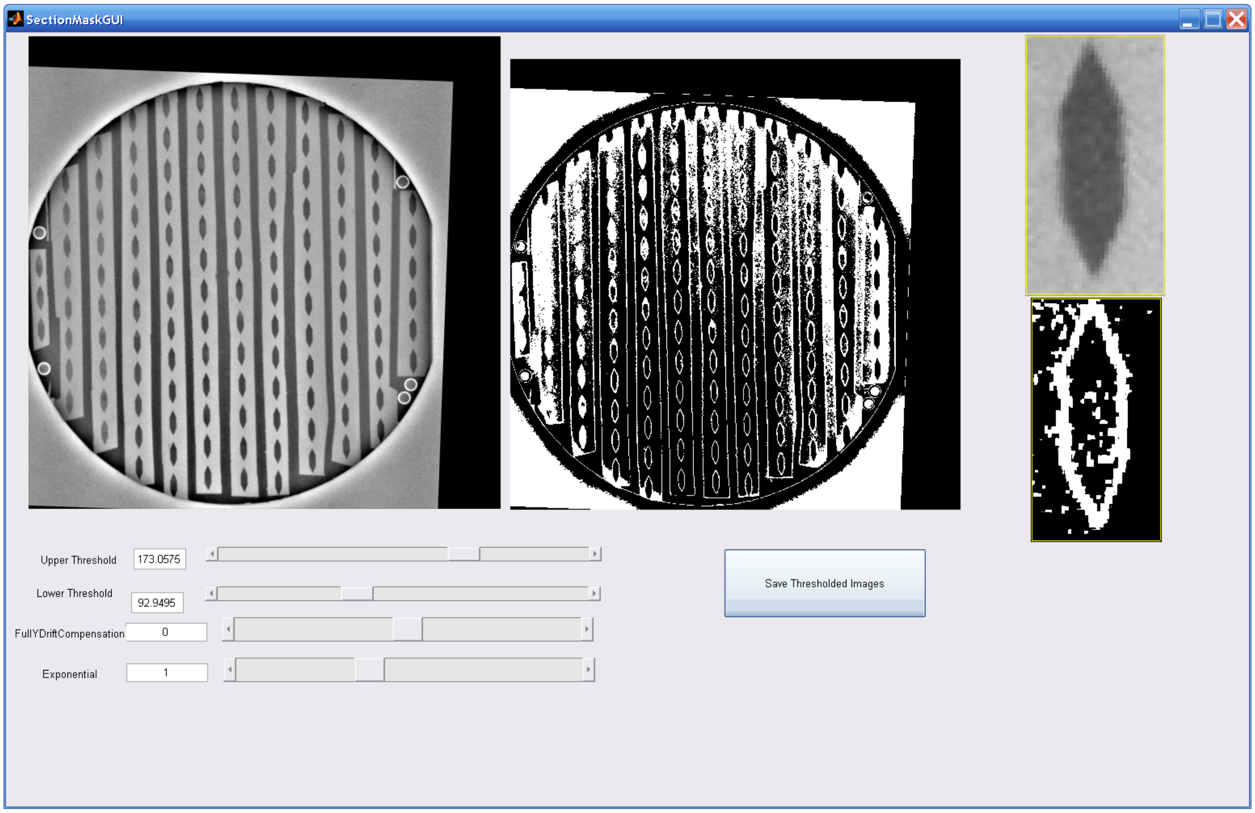
Use mouse to drag box to be around only one section. Should be a little larger than the section.

Press OK and then use your mouse to drag a box around one of the sections. When completed the cropped section is displayed. This will be used as a template to help automatically find all the other sections.

Choose **Map Wafer Operations -> Crop Example Section Image** . This is just an additional chance to crop the image in an expanded view. Make sure that the end image has the entire section and sufficient surrounding blank tape to make a good template. Here is the resulting example section image for this optical wafer map:

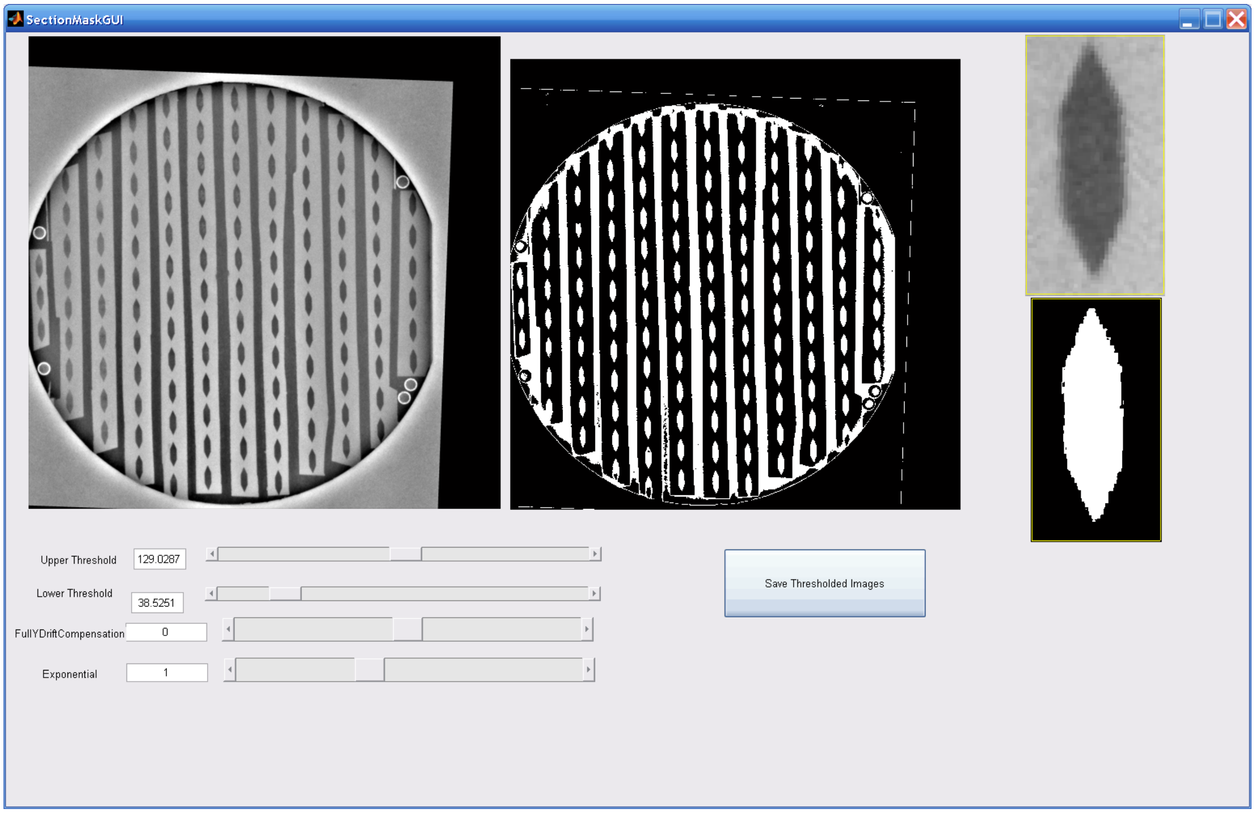


Choose **Map Wafer Operations -> Threshold Images**. This opens the SectionMaskGUI that will be used to threshold the wafer overview image and the example sections image to allow for later template matching. When it first opens it might look like this:



Upper Threshold 173.0575
Lower Threshold 92.9495
FullYDriftCompensation 0
Exponential 1

You should adjust the 'upper threshold' and 'lower threshold' scroll bars at the bottom of the GUI such that the pixels belonging to sections in the wafer overview and pixels belonging to the section in the example section image are all white, and pixels belonging to tape are black. The success of this step depends greatly on the quality of the image going in, but one should be able to get pretty close to segmenting the sections. Here is what the SectionMaskGUI looks like after proper adjustment:



Upper Threshold 129.0287

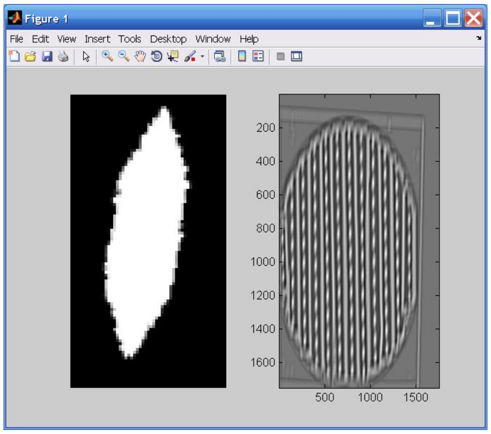
Lower Threshold 38.5251

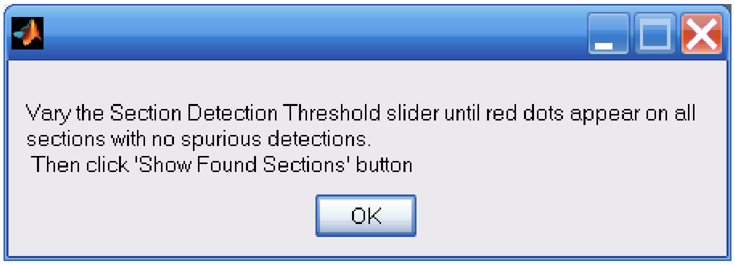
FullYDriftCompensation 0

Exponential 1

Press the ‘Save Thresholded Images’ button when done.

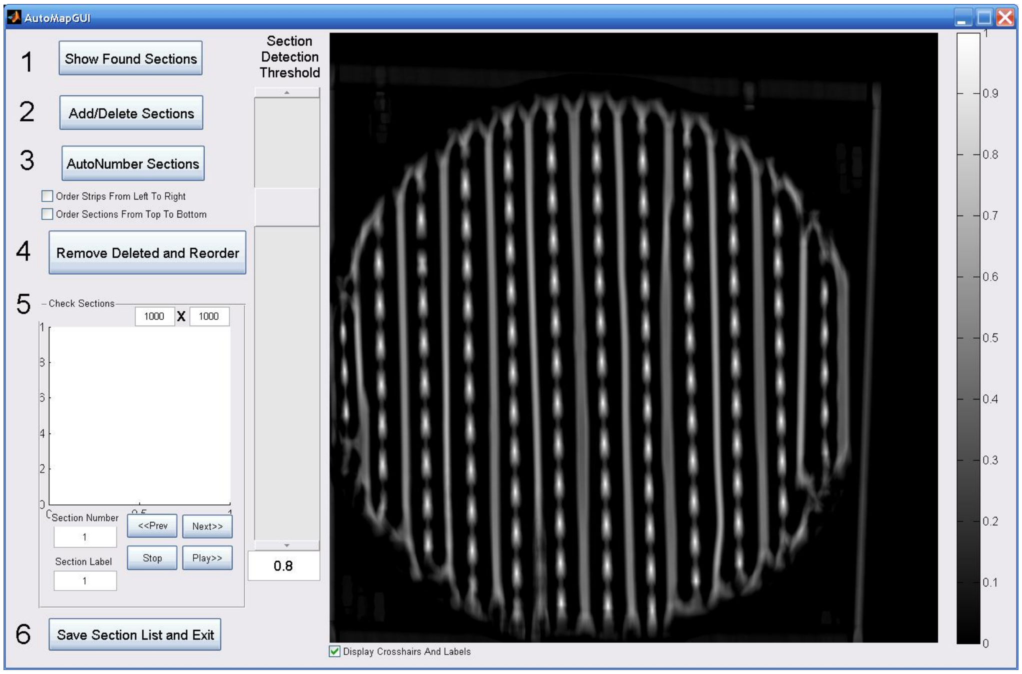
Choose **Map Wafer Operations -> Auto Map All Sections**. This starts a multi-angle template matching routine and pops up a window that looks like this, followed by the following message box:





Vary the Section Detection Threshold slider until red dots appear on all sections with no spurious detections.
Then click 'Show Found Sections' button

Pressing ‘OK’ you are presented with the AutoMapGUI with an image of the wafer whose brightness is a representation of how well the section template matched.



Section Detection Threshold: 0.8

☐ Order Strips From Left To Right
☐ Order Sections From Top To Bottom

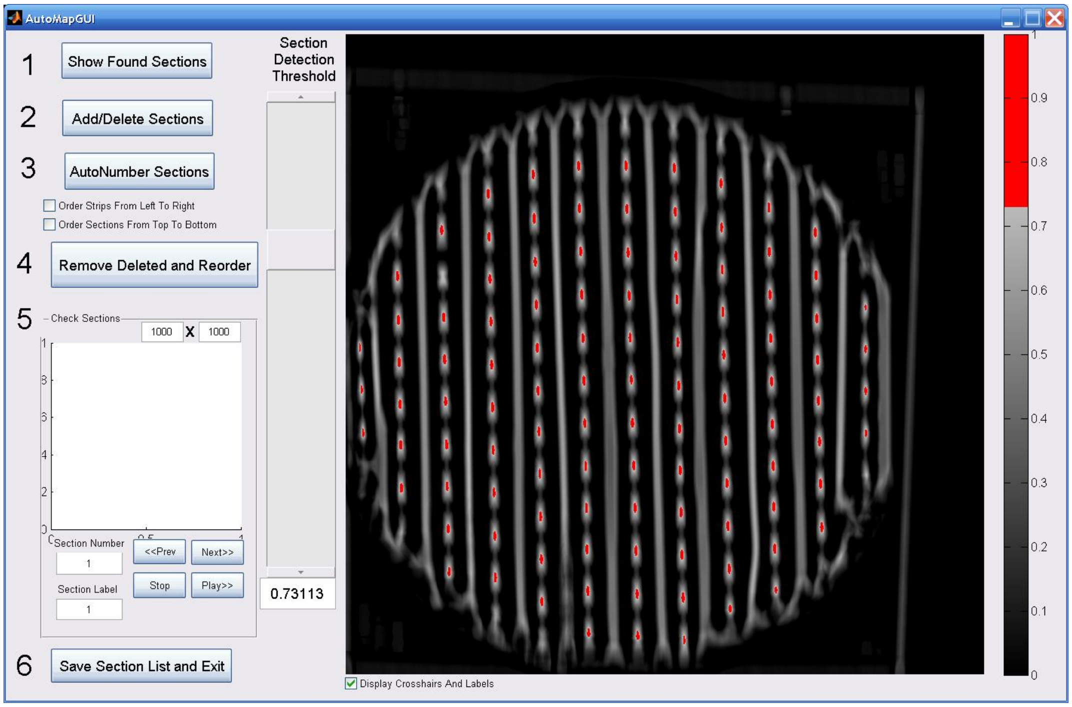
1000 X 1000

Section Number: 1

Section Label: 1

☑ Display Crosshairs And Labels

Vary the Section Detection Threshold slider up and down until red dots appear on most of the sections and no red dots appear anywhere else:



Section Detection Threshold: 0.73113

☐ Order Strips From Left To Right
☐ Order Sections From Top To Bottom

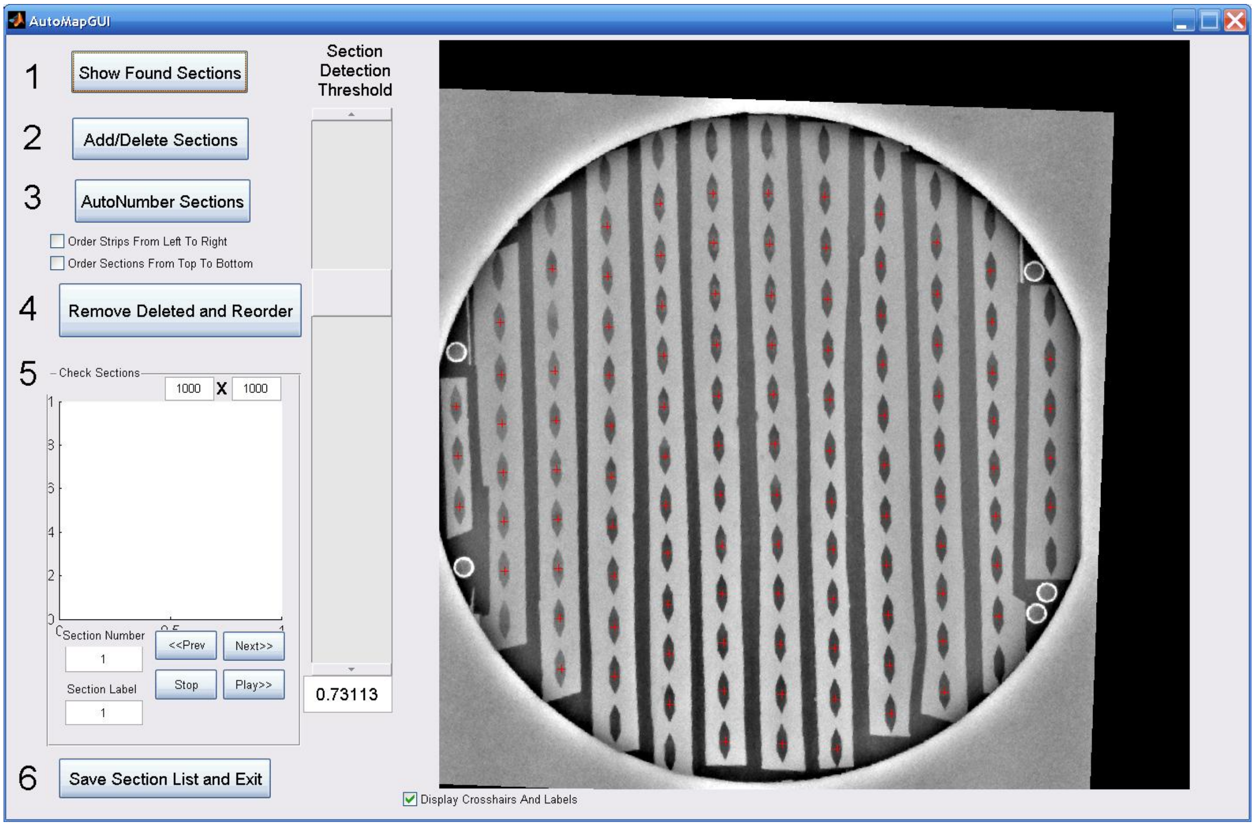
1000 X 1000

Section Number: 1

Section Label: 1

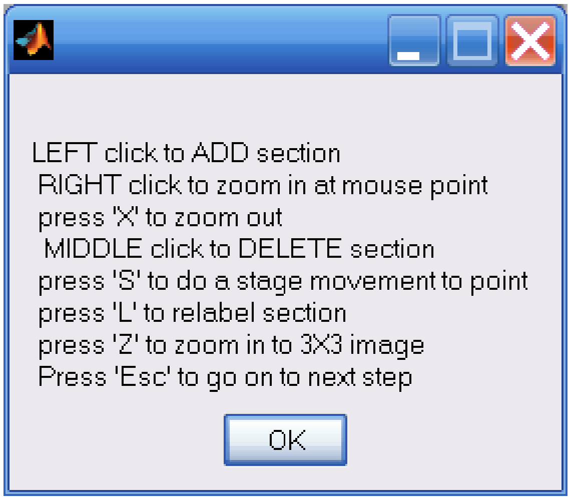
☑ Display Crosshairs And Labels

Then press the ‘Show Found Sections’ button. The image will revert to the original optical image and red crosses will start to appear one by one on the sections. A pop up will appear when the program has completed finding sections. Here is the result:


AutoMapGUI
1
Show Found Sections
2
Add/Delete Sections
3
AutoNumber Sections
□ Order Strips From Left To Right
□ Order Sections From Top To Bottom
4
Remove Deleted and Reorder
5
Check Sections
1000
X
1000
Section Number
1
Section Label
1
<<Prev
Next>>
Stop
Play>>
6
Save Section List and Exit
Section Detection Threshold
0.73113
☑ Display Crosshairs And Labels

As you can, see a few sections toward the edges were not found automatically. This is due to the excessive filtering that was needed to be applied to the original image due to poor lighting uniformity.

To add (or delete) sections press the button 'Add/Delete Sections'. The following directions pop up:



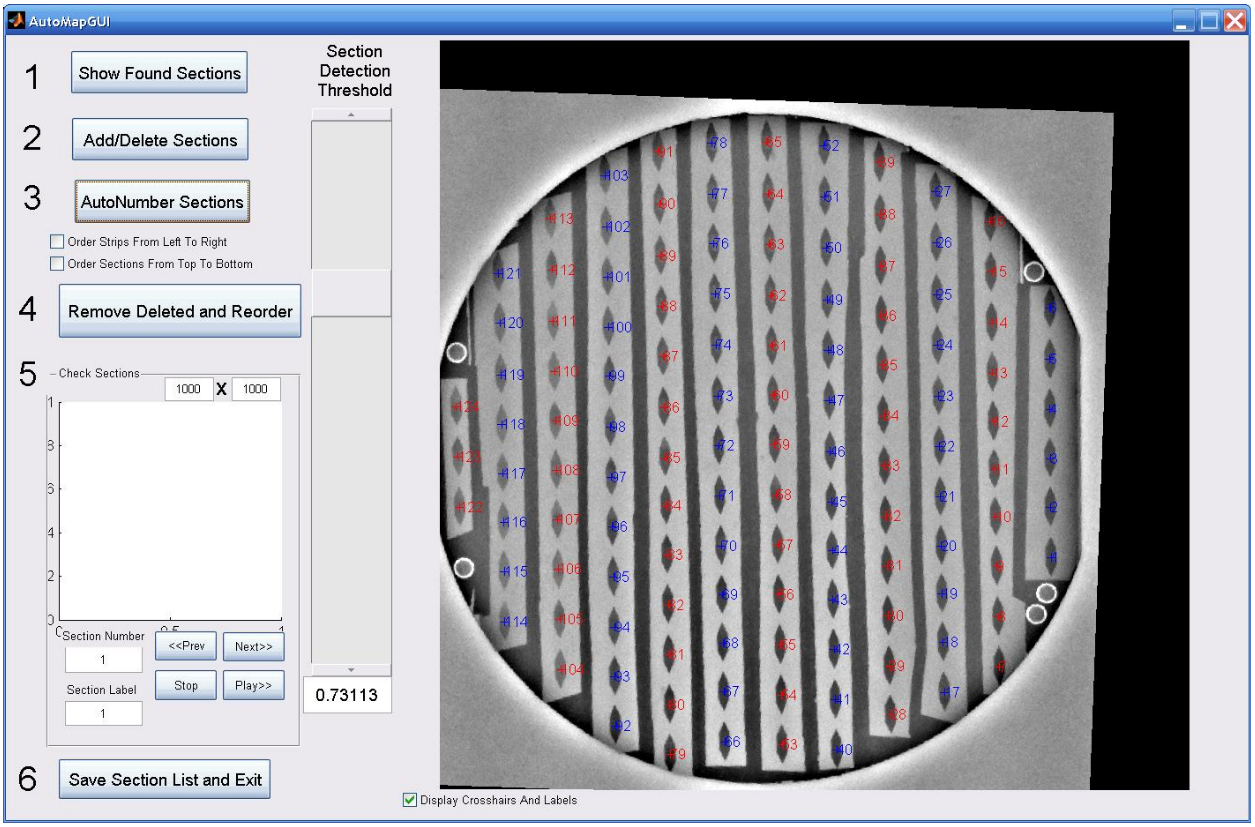
LEFT click to ADD section
RIGHT click to zoom in at mouse point
press 'X' to zoom out
MIDDLE click to DELETE section
press 'S' to do a stage movement to point
press 'L' to relabel section
press 'Z' to zoom in to 3X3 image
Press 'Esc' to go on to next step

This lists all the ways you can navigate. Crucially you can left click to add a section at the crosshairs. You can right click to zoom in at the crosshairs and then continue to left click with more accuracy. To zoom back out press ‘X’. To delete a section, move the crosshairs close near the red cross you want to delete and then do a middle mouse click.

It is sometimes difficult to determine if a section should be added by looking at the full wafer image alone (especially for small sections or runs with damaged sections). To help in the circumstances one can press ‘S’ and the SEM stage will move to the corresponding location so that you can check out that location in the SEM.

When you have put red crosses on all the sections and deleted any false positives press ‘esc’ to remove the crosshairs.

Now press the ‘**AutoNumber Sections**’ button. This numbers all sections in the ordering specified by the check boxes, it also color codes odd and even strips to make it easier to verify that the ordering is correct.



□ Order Strips From Left To Right
□ Order Sections From Top To Bottom

Check Sections: 1000 X 1000

Section Number: 1

Section Label: 1

Section Detection Threshold: 0.73113

☑ Display Crosshairs And Labels

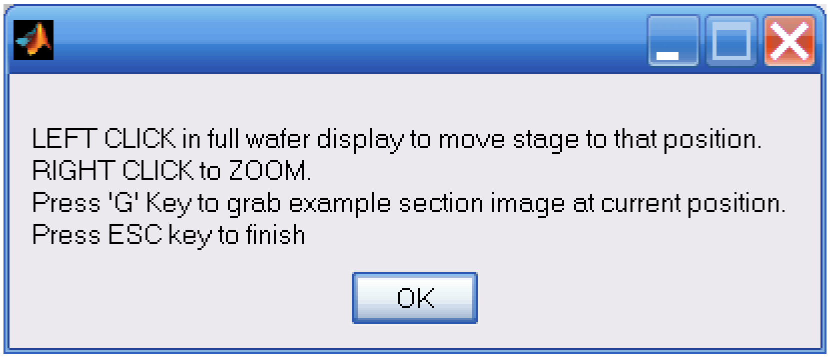
Ignore the ‘Check Sections’ menu items as this will not work on optical image data. Its functionality is for wafers that are mapper with an SEM full wafer montage as described next.

Press the button ‘**Remove Deleted and Reorder**’. This does an internal consistency check to make sure that any manually deleted sections were actually deleted. Then press ‘Save Section List and Exit’. Section numbers will now appear in the WaferMapper GUI’s Wafer Overview Figure.

If an SEM montage wafer image:

If the full wafer image was obtained via performing a full wafer montage using WaferMapper (as described in section 1.4 above) these are the steps to follow to map all of the sections (many are very similar to the optical method described above).

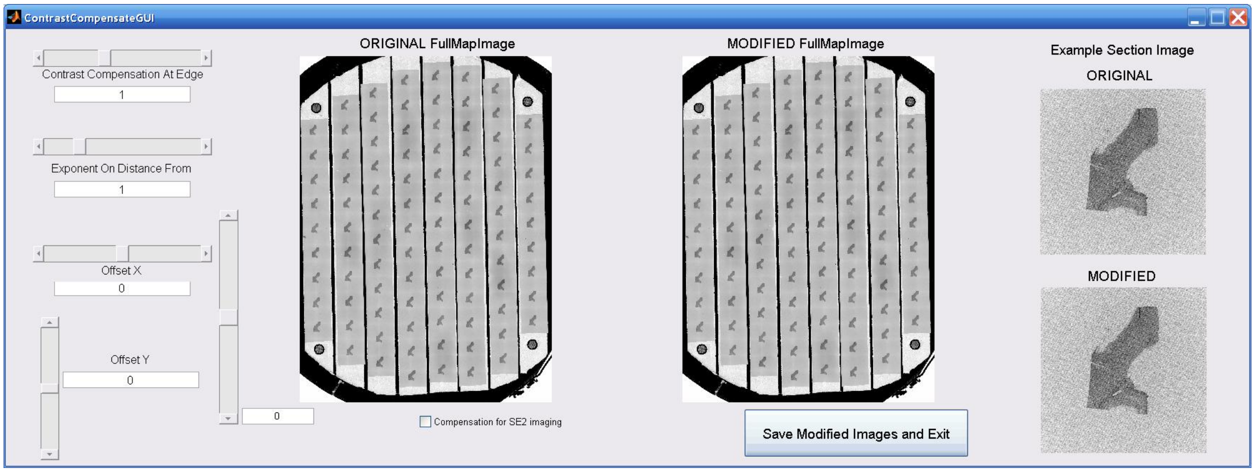
Choose **Map Wafer Operations -> Acquire Example Section Image**. The following message box will appear:



LEFT CLICK in full wafer display to move stage to that position.
RIGHT CLICK to ZOOM.
Press 'G' Key to grab example section image at current position.
Press ESC key to finish

Use the crosshairs to click on one of the sections on the wafer and the microscope stage will move to that section. Then press 'G' to grab the image of this section. This image will be taken at the same mag used for the full wafer montage tiles and it will serve as the template to automatically find all the sections on this wafer. For this reason it is important not to change the brightness or contrast of this image relative to the conditions under which the full wafer montage tiles were taken. When done acquiring press the 'esc' key to remove crosshairs.

Choose **Map Wafer Operations -> Contrast Compensate**. The following GUI appears:



Contrast Compensation At Edge 1

Exponent On Distance From 1

Offset X 0

Offset Y 0

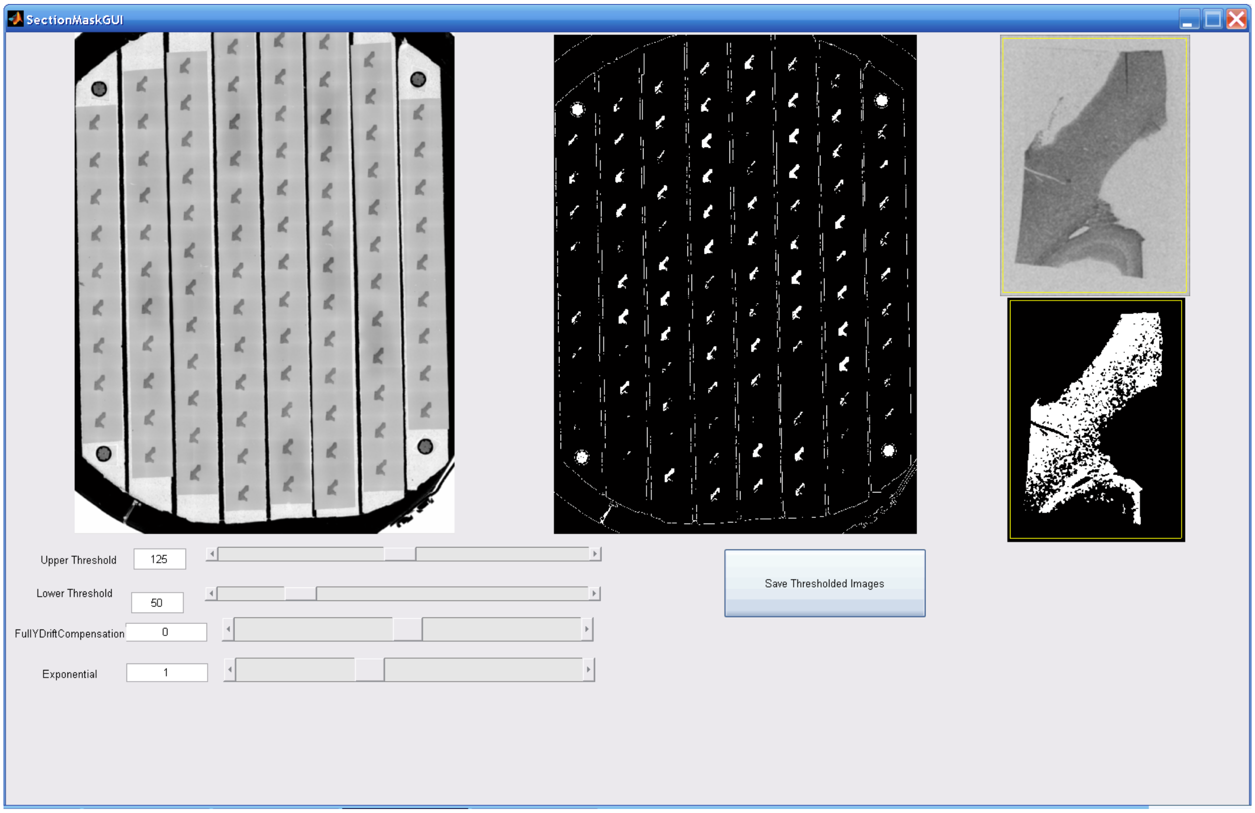
0

□ Compensation for SE2 imaging

This GUI is designed to help compensate for contrast artifacts at the edges of the montage tiles that are produced when a too short working distance is used for the full wafer montage acquisition. The user simply dials the sliders at the left until the right hand full wafer image looks smooth. In this case there are no edge artifacts because the full wafer montage was obtained at 9.5mm working distance. Simply press the 'Save Modified Images and Exit' button to go on to the next step.

Choose **Map Wafer Operations -> Crop Example Section Image** , and drag a box around the section and a bit of the surrounding tape. This will be used to make a template for matching the rest of the sections.

Choose **Map Wafer Operations -> Threshold Images**. This opens the SectionMaskGUI that will be used to threshold the wafer overview image and the example sections image to allow for later template matching. When it first opens on an SEM obtained wafer image it might look like this:



SectionMaskGUI

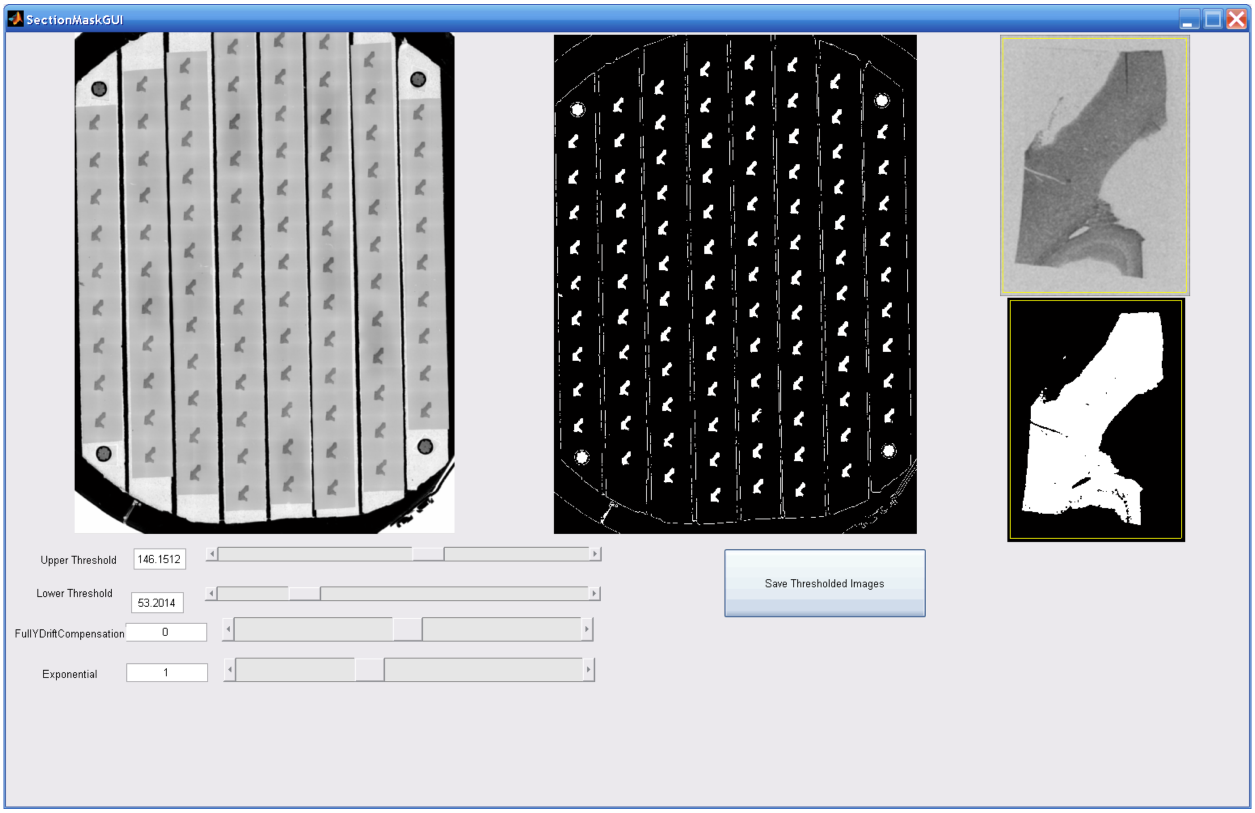
Upper Threshold 125

Lower Threshold 50

FullYDriftCompensation 0

Exponential 1

You should adjust the 'upper threshold' and 'lower threshold' scroll bars at the bottom of the GUI such that the pixels belonging to sections in the wafer overview and pixels belonging to the section in the example section image are all white, and pixels belonging to tape are black. Here is what the SectionMaskGUI looks like after proper adjustment:



Upper Threshold 146.1512

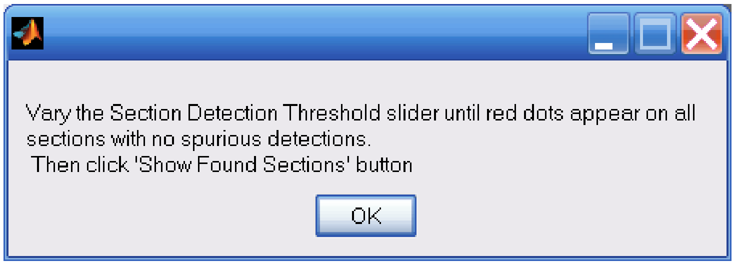
Lower Threshold 53.2014

FullYDriftCompensation 0

Exponential 1

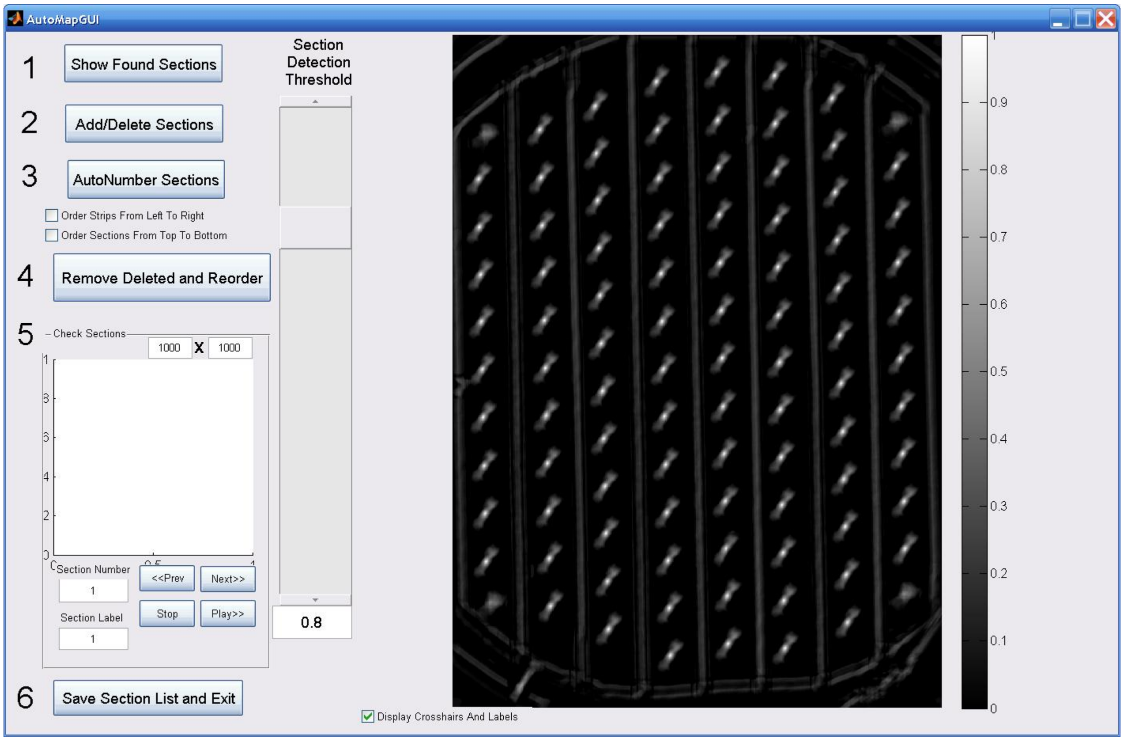
Press the ‘Save Thresholded Images’ button when done.

Choose **Map Wafer Operations -> Auto Map All Sections**. This starts a multi-angle template matching routine, followed by the following message box:



Vary the Section Detection Threshold slider until red dots appear on all sections with no spurious detections.
Then click 'Show Found Sections' button

Pressing ‘OK’ you are presented with the AutoMapGUI with an image of the wafer whose brightness is a representation of how well the section template matched.



☐ Order Strips From Left To Right
☐ Order Sections From Top To Bottom

1000 X 1000

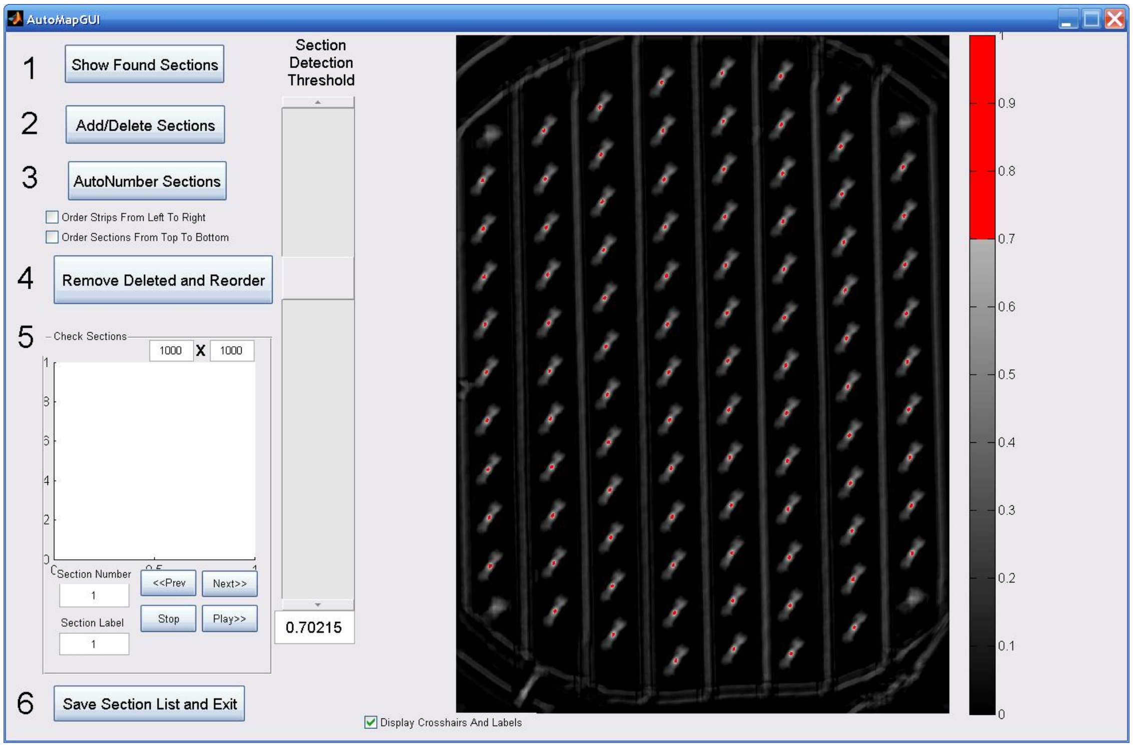
Section Number: 1

Section Label: 1

Section Detection Threshold: 0.8

☑ Display Crosshairs And Labels

Vary the Section Detection Threshold slider up and down until red dots appear on most of the sections and no red dots appear on anywhere else:



☐ Order Strips From Left To Right
☐ Order Sections From Top To Bottom

1000 X 1000

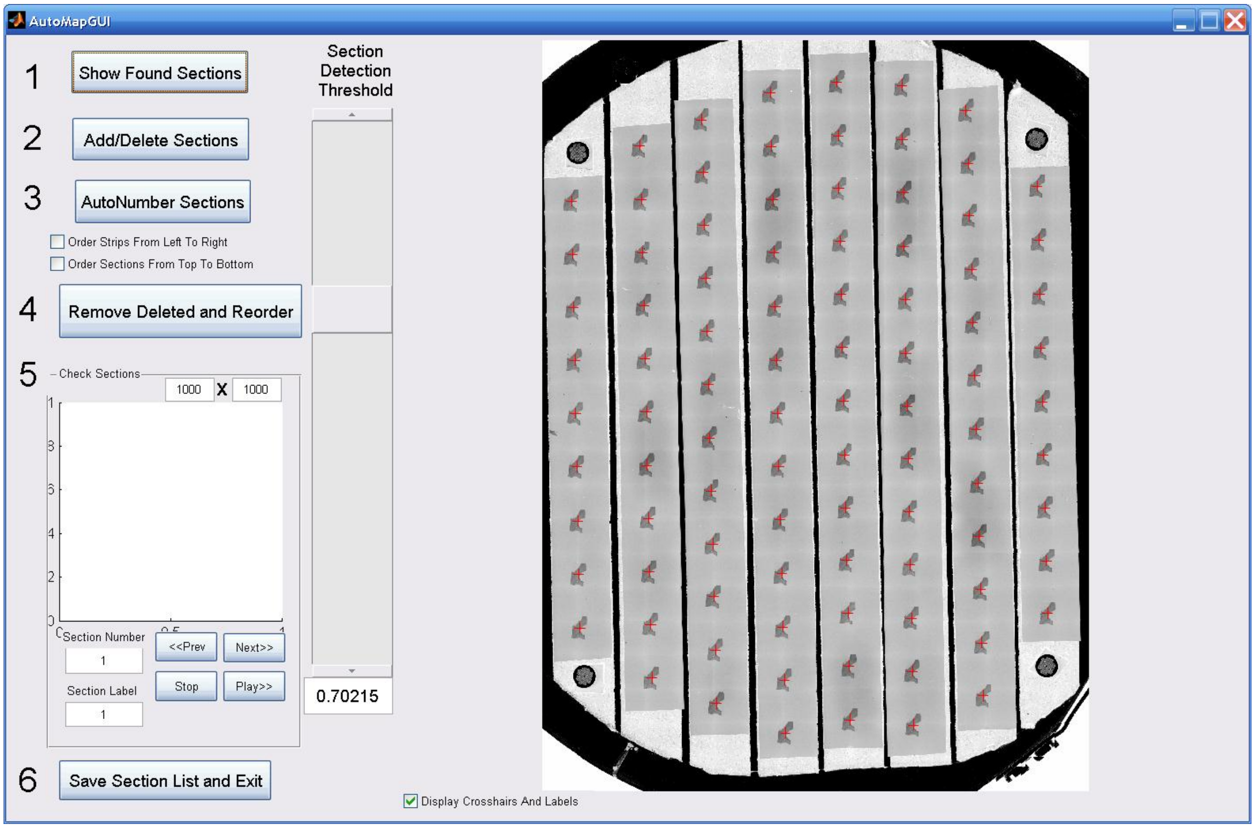
Section Number: 1

Section Label: 1

Section Detection Threshold: 0.70215

☑ Display Crosshairs And Labels

Then press the '**Show Found Sections**' button. The image will revert to the original SEM montage image and red crosses will start to appear one by one on the sections. A pop up will appear when the program has completed finding sections. Here is the result:



☐ Order Strips From Left To Right
☐ Order Sections From Top To Bottom

1000 X 1000

Section Number: 1

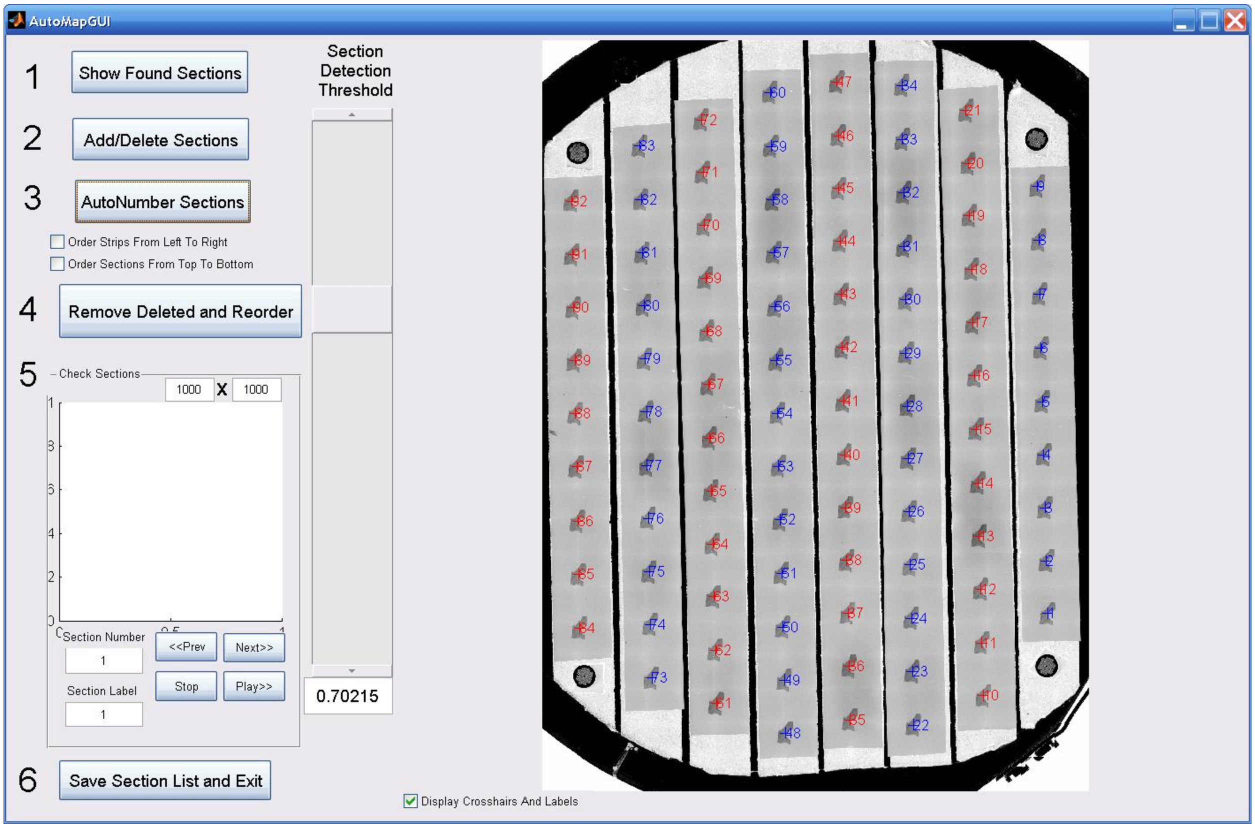
Section Label: 1

Section Detection Threshold: 0.70215

☑ Display Crosshairs And Labels

To add (or delete) sections press the button 'Add/Delete Sections'. This procedure is described in more detail above in the optical section.

Now press the '**AutoNumber Sections**' button. This numbers all sections in the ordering specified by the check boxes, it also color codes odd and even strips to make it easier to verify the ordering is correct.


AutoMapGUI
1
Show Found Sections
2
Add/Delete Sections
3
AutoNumber Sections
□ Order Strips From Left To Right
□ Order Sections From Top To Bottom
4
Remove Deleted and Reorder
5
Check Sections
1000 X 1000
Section Number
1
<<Prev
Next>>
Section Label
1
Stop
Play>>
6
Save Section List and Exit
Section Detection Threshold
0.70215
☑ Display Crosshairs And Labels

Press the button '**Remove Deleted and Reorder**'. This does an internal consistency check to make sure that any manually deleted sections were actually deleted.

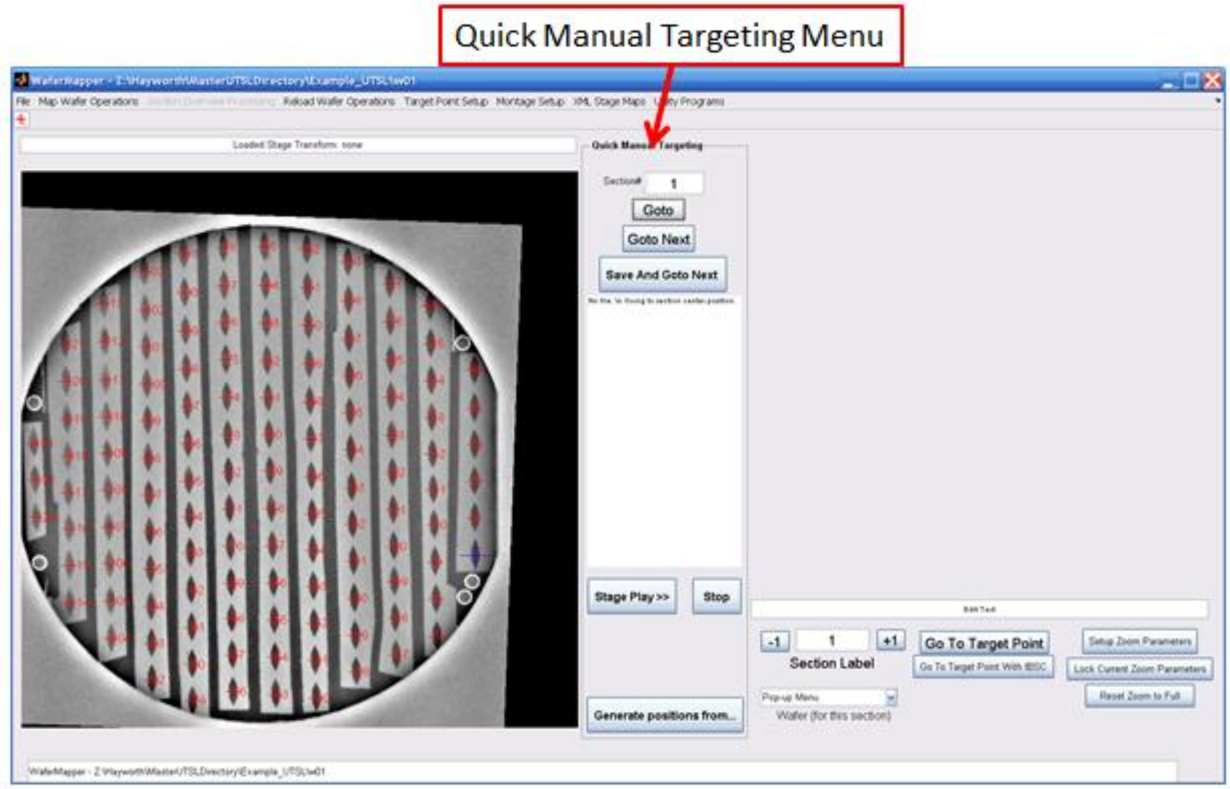
You can press the '**Play>>**' button to get a zoomed in version of each section in order to help verify that all sections are labeled correctly.

Then press '**Save Section List and Exit**'. Section numbers will now appear in the WaferMapper GUI's Wafer Overview Figure.

1.7 Quick manual targeting and XML stage map creation

At this point the wafer is not fully mapped (this requires section overviews to be acquired, see the next section). Full mapping including acquisition of section overviews is required for automatic targeting of imaging within each section. However, the wafer is now mapped sufficiently to allow for quick manual targeting on each section for the creation of XML stage maps for use in the Fibics ATLAS software. This quick manual targeting can also be used to define the positions for montages acquired using the WaferMapper software.

In the middle of the WaferMapper GUI is the 'Quick Manual Targeting' menu:



At this stage in the mapping process you can type in a section number and press the '**Goto**' button and the SEM stage will move to the center of the section. (Note: this also works after a wafer reload, described below, using the fiducial-based stage correction to correctly compensate for reload errors.) You can also press the '**Goto Next**' button and the section number will be incremented and the stage will move to that section.

More importantly, you can press the '**Save And Goto Next**' button. This will save (in the ManuallyCorrectedStagePositionsDirectory) a 'CorrectedStagePostion_x.mat' file that contains the stage position (and other microscope parameters) that are current just before the button is pressed. This button is used to enable the user to as quickly and efficiently as possible perform manual targeting on each section. Here is how it works:

First the user types '1' into the section# box and presses '**Goto**'. Then the user moves their mouse to the SmartSEM window and presses ctrl+tab to move the stage to the correct position for this section. Then the user moves their mouse back to the WaferMapper window and presses the '**Save And Goto Next**' button. The user repeats this process for all sections she wants to manually target. If she wishes to skip a section then she just presses '**Goto Next**' instead of '**Save And Goto Next**'. If the user goes back to a section that already has a saved 'CorrectedStagePostion_x.mat' file then that saved position is the one that the stage will be driven to. (Note: the user can also simply delete a saved 'CorrectedStagePostion_x.mat' file for a particular section and the program will revert to using that section's center as its goto point).

When all desired sections have been manually targeted using this method, the user can type in the number of the first saved section and press '**Goto**'. Then she can press the '**Stage Play>>**' button to have the microscope stage move as quickly as possible automatically going through all saved positions in order until it reaches a section not yet saved or reaches the last section on the wafer. Pressing the '**Stop**' button will stop this Stage Play sequence.

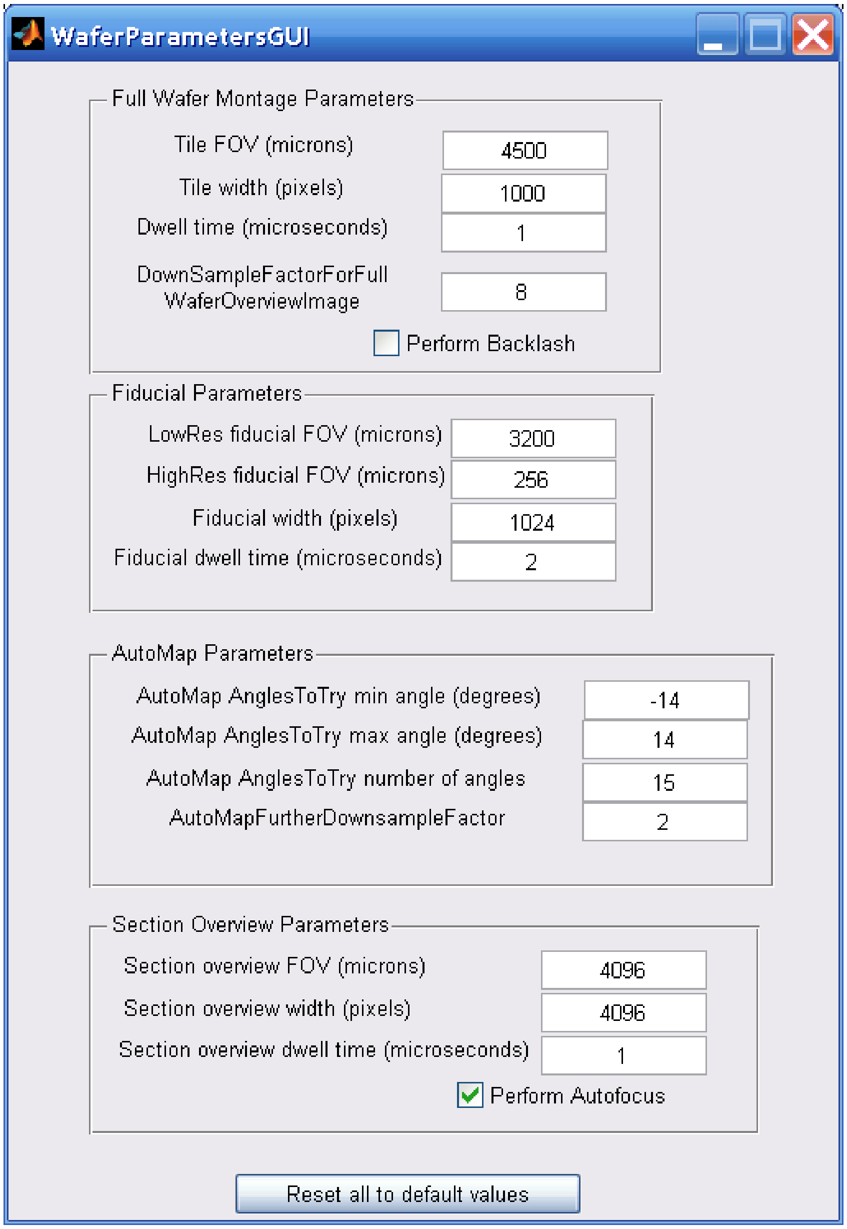
To convert the list of saved 'CorrectedStagePostion_x.mat' files into an XML stage map that is readable by the Fibics ATLAS software choose **XML Stage Maps -> Generate XML From Manually Corrected Stage Pos Files**. This will create an XML file containing only those sections that the user has created 'CorrectedStagePostion_x.mat' files for in the ManuallyCorrectedStragePositionsDirectory. Currently the program will generate an XML file having the same WD, stig, brightness, contrast, etc. for all sections, and these will be set to whatever the microscope value is when the **XML Stage Maps -> Generate XML From Manually Corrected Stage Pos Files** menu item is chosen.

To generate 'CorrectedStagePostion_x.mat' files for all sections at once press the '**Generate Positions From…**' button and choose the '**Section List**' option. This (and using the **XML Stage Maps -> Generate XML From Manually Corrected Stage Pos Files**) allows a quick way to export all the sections center positions to the ATLAS software.

You can also use WaferMapper to take montage stacks at the positions of these 'CorrectedStagePostion_x.mat' files using the tools in the **Montage Setup** menu. This process will be described later.

1.8 Acquiring Section Overviews

The next step is to perform a pixel to stage calibration to determine the precise number of microns of stage movement corresponding to a certain number of pixels at the image parameters used for taking the section overviews. First choose **Map Wafer Operations -> Wafer Parameters**. This opens up the following dialog box:



Full Wafer Montage Parameters

Tile FOV (microns): 4500

Tile width (pixels): 1000

Dwell time (microseconds): 1

DownSampleFactorForFullWaferOverviewImage: 8

☐ Perform Backlash

Fiducial Parameters

LowRes fiducial FOV (microns): 3200

HighRes fiducial FOV (microns): 256

Fiducial width (pixels): 1024

Fiducial dwell time (microseconds): 2

AutoMap Parameters

AutoMap AnglesToTry min angle (degrees): -14

AutoMap AnglesToTry max angle (degrees): 14

AutoMap AnglesToTry number of angles: 15

AutoMapFurtherDownsampleFactor: 2

Section Overview Parameters

Section overview FOV (microns): 4096

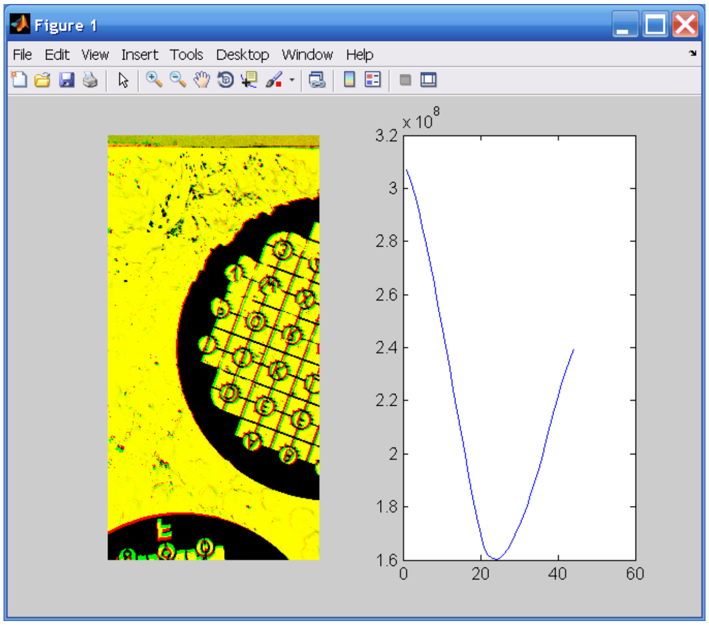
Section overview width (pixels): 4096

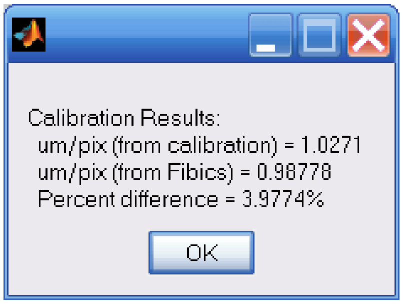
Section overview dwell time (microseconds): 1

☑ Perform Autofocus

At the bottom are the imaging parameters for taking the section overviews. Make sure these are properly filled out BEFORE doing the pixel to stage calibration (since this calibration will be for these parameters only)!!!

Now choose **Map Wafer Operations -> Perform Pixel To Stage Calibration**. This presents you with crosshairs in the WaferMapper GUI's Wafer Overview Display and a dialog that says to right click to move the stage and to press 'G' to grab an image. Right click to one of the fiducials. Check the focus in the SmartSEM display and then and press 'G' in the Wafer Overview Display. The program will now automatically acquire two images of this fiducial, the second offset by 1mm of stage movement. When complete the crosshairs will reappear, press 'esc' to remove them. Immediately following this a Matlab figure will show up that will slowly slide one image across the other while plotting the difference between the images. At the end a message will pop up displaying the calibration results which are stored to a file:




Calibration Results:
um/pix (from calibration) = 1.0271
um/pix (from Fibics) = 0.98778
Percent difference = 3.9774%

Now go to the SmartSEM window and check the focus, stig, contrast, and brightness in preparation for taking section overviews. Then choose **Map Wafer Operations -> Acquire Section Overviews.** The microscope will automatically go to each of the sections and acquire the section overview images which will be stored in a subdirectory called SectionOverviewsDirectory along with imaging data files readable by Matlab.

When the program finishes taking images you should check the thumbnails in the SectionOverviewsDirectory to make sure there were no problems. If there are bad images you can delete them (or better yet rename them with a _removed at the end) and then choose **Map Wafer Operations -> Acquire Section Overviews** again. The program will only retake those images that have been renamed/deleted.

Once the section overviews have been acquired the wafer is officially ‘fully mapped’ which means that you can feel free to take it out of the microscope and know that it can be reloaded and put back into registration allowing for multiple imaging sessions. It is important not to remove the wafer from the chamber before this section overview imaging is completed if you wish to use the full functionality of the WaferMapper software during later imaging sessions.

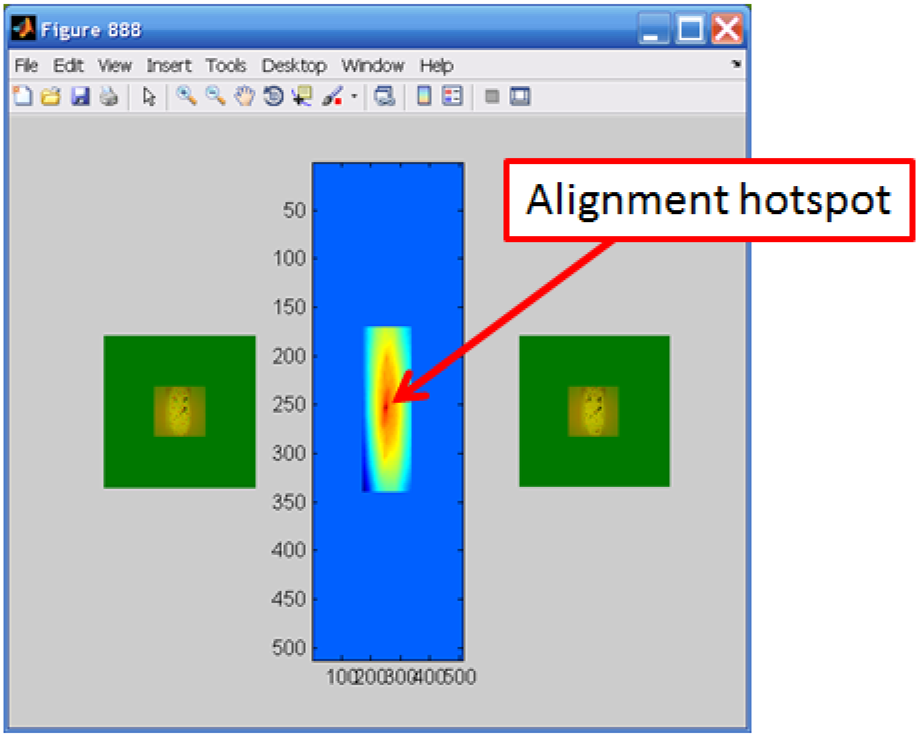
1.9 Aligning Section Overviews

After the section overviews are acquired the **Section Overview Processing** menu becomes enabled. (Note: Section overview processing can be performed offline on a different computer running a separate copy of WaferMapper, just open up the fully mapped wafer.)

The WaferMapper software does not aligned each section to the previous section, the reason being that this would present a cascading problem if one section was damaged. Instead, the user chooses one section on the current wafer or the pervious wafer to provide the template for aligning all sections. If you are aligning overviews in the first wafer in a UTSL then choose **Section Overview Processing -> Choose Section Template Image (for First Wafer)**, if you are aligning overviews in the Nth wafer in a UTSL then choose **Section Overview Processing -> Choose Section Template Image (for Subsequent Wafer)**. You will be presented with a list box containing all the section overviews. Look at the thumbnails in the directory to find one that will make a good template and choose it.

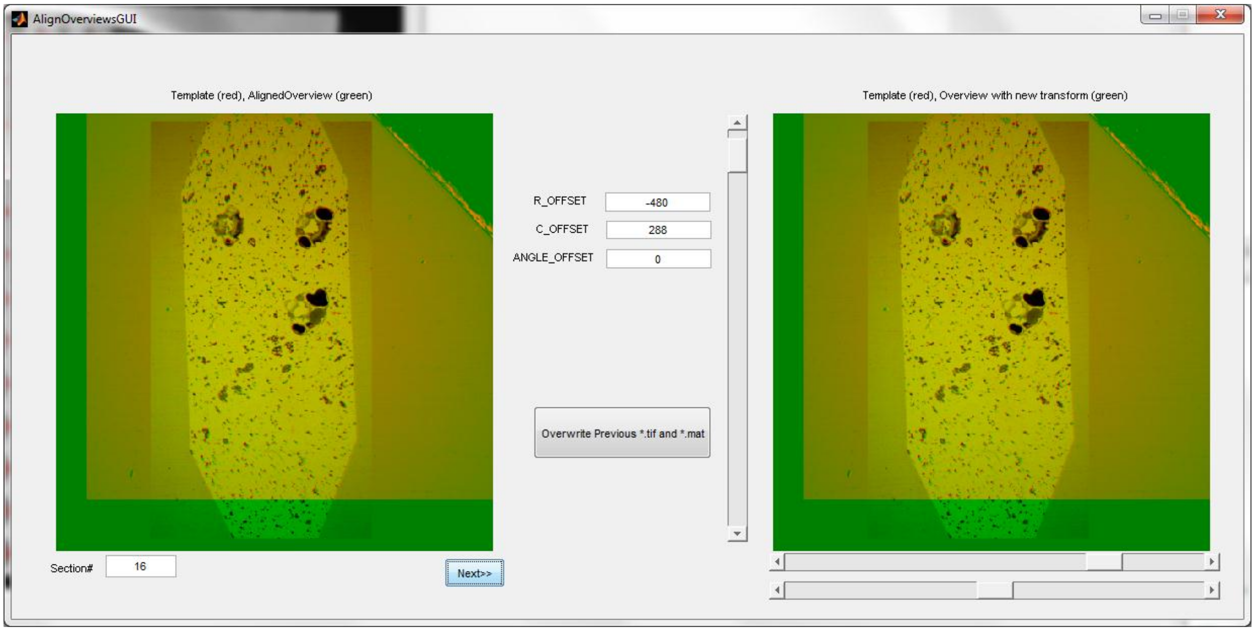
Then choose **Section Overview Processing -> Crop Section Template Image**. This presents the section overview template in a Matlab figure. Drag a box around the section, pixels outside this box will be set to the average value of its perimeter.

Finally choose **Section Overview Processing -> Align Section Overviews**. This starts a process where each overview image is aligned to the template. If everything is working correctly there should be a relatively clear 'hotspot' of correct alignment in each of the processing image as shown below:



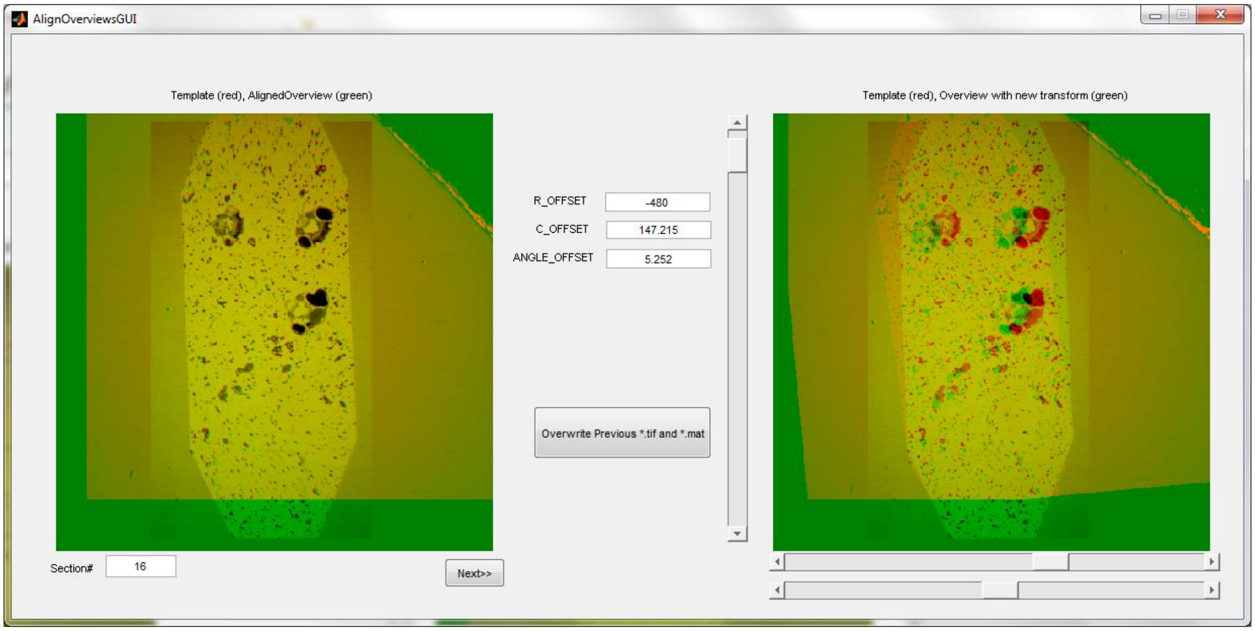
The aligned overview sections are saved in the SectionOverviewsAlignedWithTemplateDirectory subdirectory.

Once this alignment process is complete choose **Section Overview Processing -> Check and Correct Alignment GUI**. This GUI allows you to walk through each section to check its alignment vs. the template, and the GUI provides X, Y, and rotation sliders to correct a section's alignment if necessary. Below is a screen shot of the AlignOverviewsGUI.


AlignOverviewsGUI
Template (red), AlignedOverview (green)
Template (red), Overview with new transform (green)
Overwrite Previous *.tif and *.mat
Next>>

R_OFFSET -480
C_OFFSET 288
ANGLE_OFFSET 0
Section# 16

Moving the sliders translates and rotates the section relative to the template as shown below:


AlignOverviewsGUI
Template (red), AlignedOverview (green)
Template (red), Overview with new transform (green)
Overwrite Previous *.tif and *.mat
Next>>

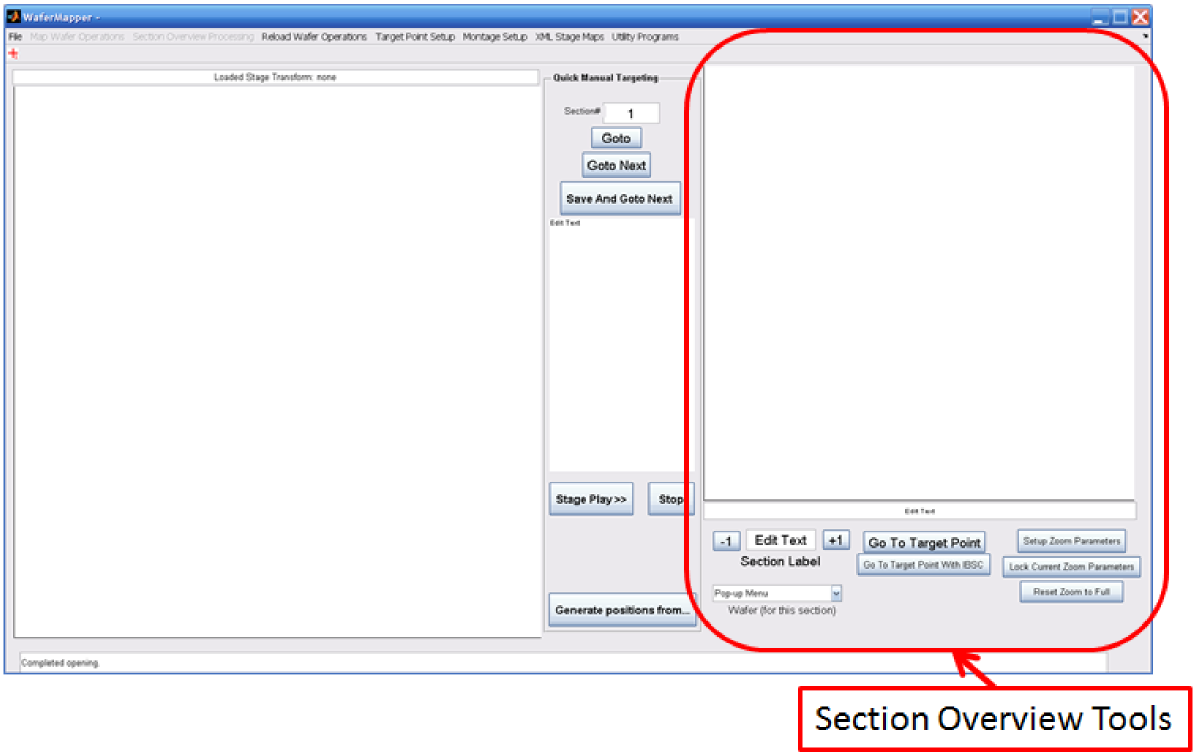
R_OFFSET -480
C_OFFSET 147.215
ANGLE_OFFSET 5.252
Section# 16

To save a manually corrected image simply press the '**Overwrite Previous *.tif and *.mat**' button. Clicking the '**Next>>**' button displays the next section, and you can type in a particular section number in the **Section#** box and go to it directly.

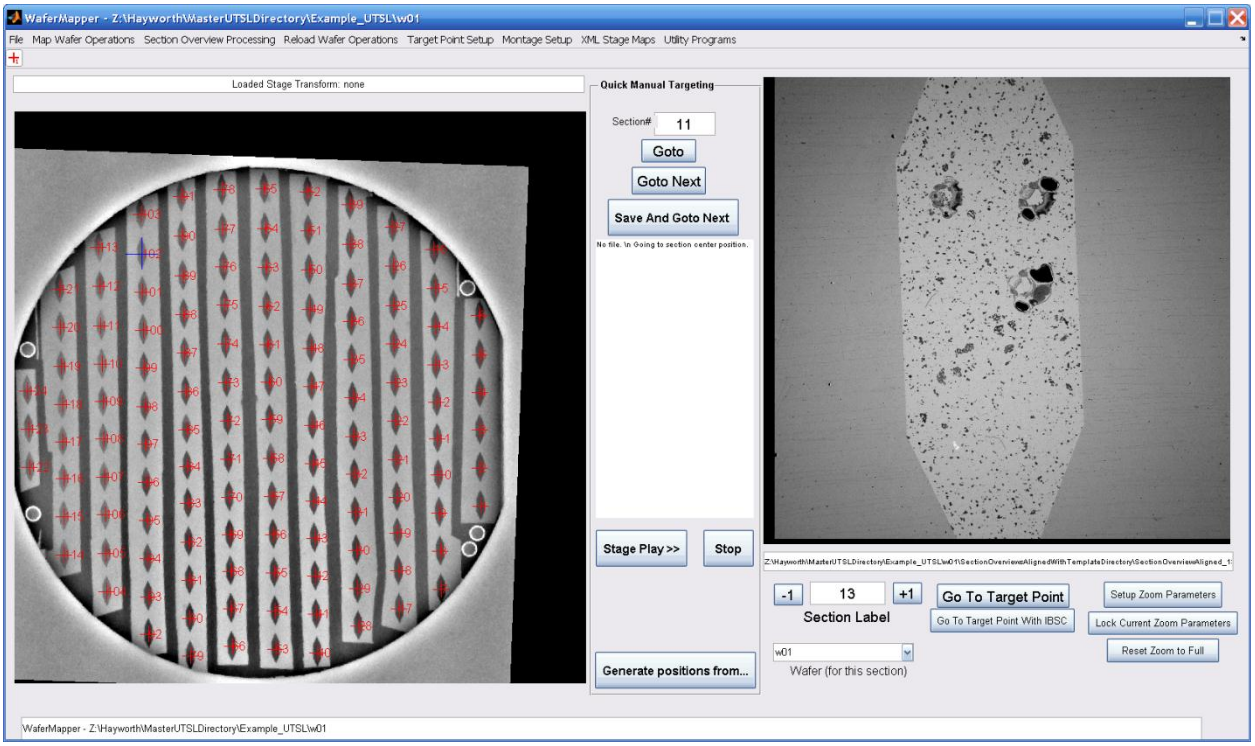
When you are satisfied that all section overviews are aligned properly close the AlignOverviewsGUI.

1.10 Choosing a target point

Once the section overviews are aligned you can now use the Section Overview tools on the right side of the WaferMapper GUI. These tools allow you to navigate through the entire stack of aligned section overview images across all wafers in the UTSL. They allow you to zoom in on any part of the stack, and most importantly they allow you to graphically define the positioning of your imaging montage directly on the aligned stack of section overviews.



First choose **File -> Open UTSL** and reopen the current UTSL. This is required to refresh the list of wafers that the program stores for the UTSL. Then choose **File -> Open Wafer** and reopen the current wafer. Now go to the '**Section Label**' edit box at the bottom of the Section Overview Tools and enter 13 and press enter. The aligned 13th section of this wafer will be displayed in the Section Overview Display:



Loaded Stage Transform: none

Section# 11

No file. \n Going to section center position.

Z:\Hayworth\MasterUTSLDirectory\Example_UTSL\w01\SectionOverviewsAlignedWithTemplateDirectory\SectionOverviewAligned_1:

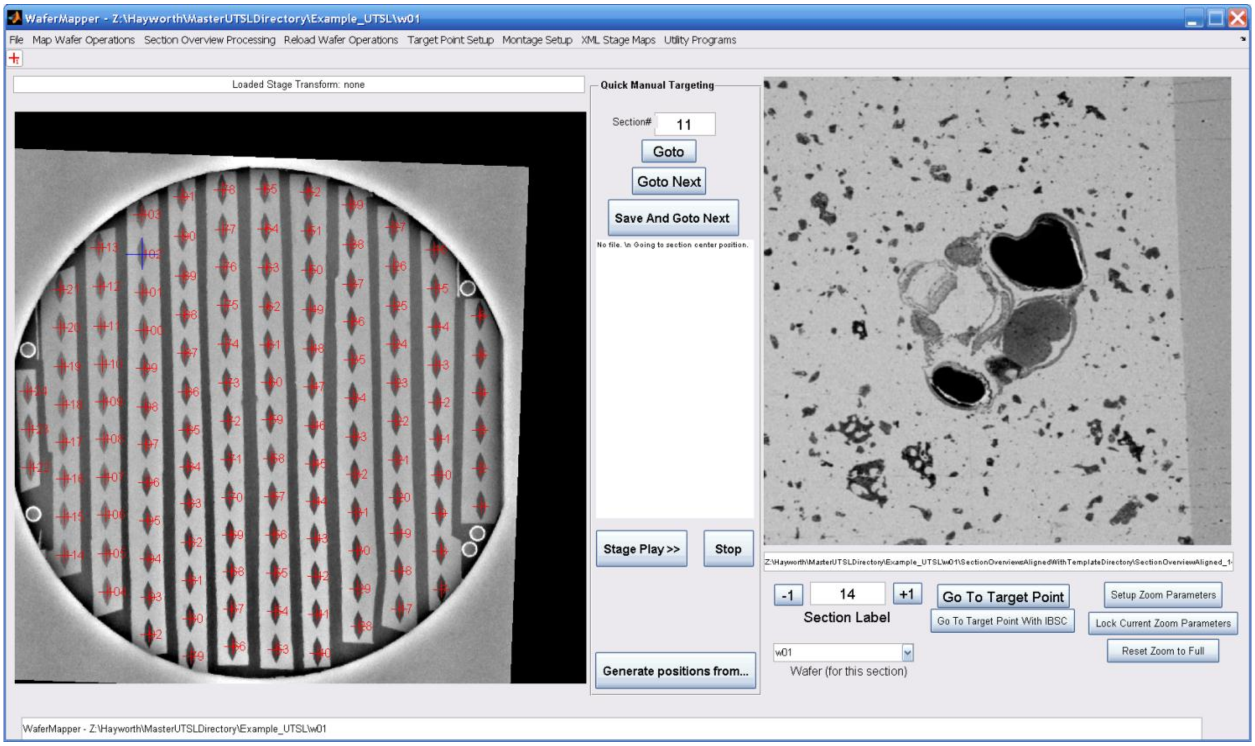
Section Label: 13

Wafer (for this section): w01

WaferMapper - Z:\Hayworth\MasterUTSLDirectory\Example_UTSL\w01

Now press the '**+1**' button. This will increment to the next section and display it. If there are multiple wafers in the UTSL then you can choose a different wafer using the **Wafer (for this section)** list box and then entering in a new section in the **Section Label** box – once you press enter that aligned section overview will be displayed.

Now press the "**Setup Zoom Parameters**" button. The mouse will change to a magnifying glass over the figure. This is the Matlab figure zoom tool and you can use it just like you would in any Matlab figure (e.g. right clicking on the figure will popup a menu for zoom out etc.). Zoom into see the region of the section overview that you want to image and then press the "**Lock Current Zoom Parameters**" button. This is important since it removes the magnifying glass mouse icon and lets the program record the current zoom level for subsequent viewing of other sections in the stack. Here is what the WaferMapper GUI looks like after such a zooming operation:


WaferMapper - Z:\Hayworth\MasterUTSLDirectory\Example_UTSL\w01

Loaded Stage Transform: none

Section# 11

No file. \n Going to section center position.

Z:\Hayworth\MasterUTSLDirectory\Example_UTSL\w01\SectionOverviewsAlignedWithTemplateDirectory\SectionOverviewAligned_1-

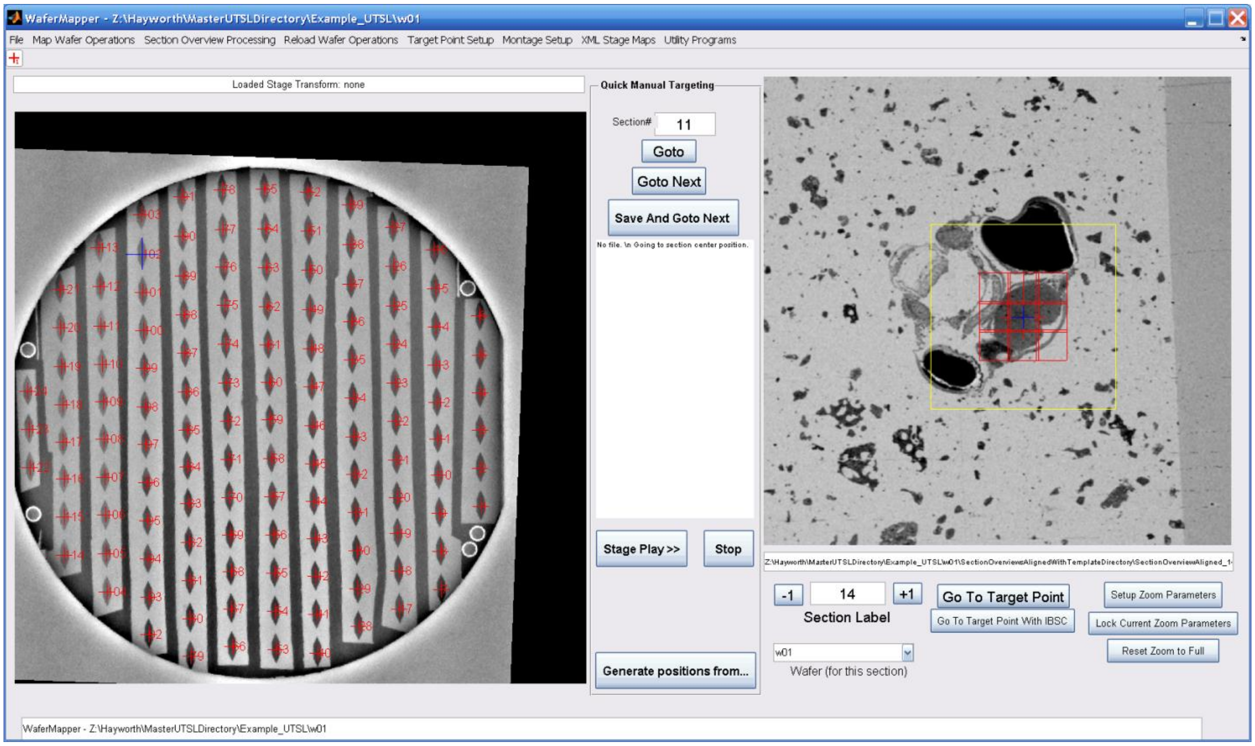
Section Label 14

Wafer (for this section) w01

WaferMapper - Z:\Hayworth\MasterUTSLDirectory\Example_UTSL\w01

Using the '**+1**' and '**-1**' buttons, flip through a few sections to see how precise the alignment is now that you have zoomed up on the region you are interested in imaging. For some runs and for some imaging tasks this section overview alignment may be sufficient, for others where there is significant distortion across the sections an additional alignment step is needed. This will be described later.

Choose **Target Point Setup -> Choose Target In Aligned Section Overviews**. This will display crosshairs in the Section Overview Display. Click in the center of the tissue region you want to take an image of. This will create a graphical representation of the default 3x3 montage at that point:


WaferMapper - Z:\Hayworth\MasterUTSLDirectory\Example_UTSL\w01

Loaded Stage Transform: none

Section# 11

No file. \n Going to section center position.

Z:\Hayworth\MasterUTSLDirectory\Example_UTSL\w01\SectionOverviewsAlignedWithTemplateDirectory\SectionOverviewAligned_1

Section Label 14

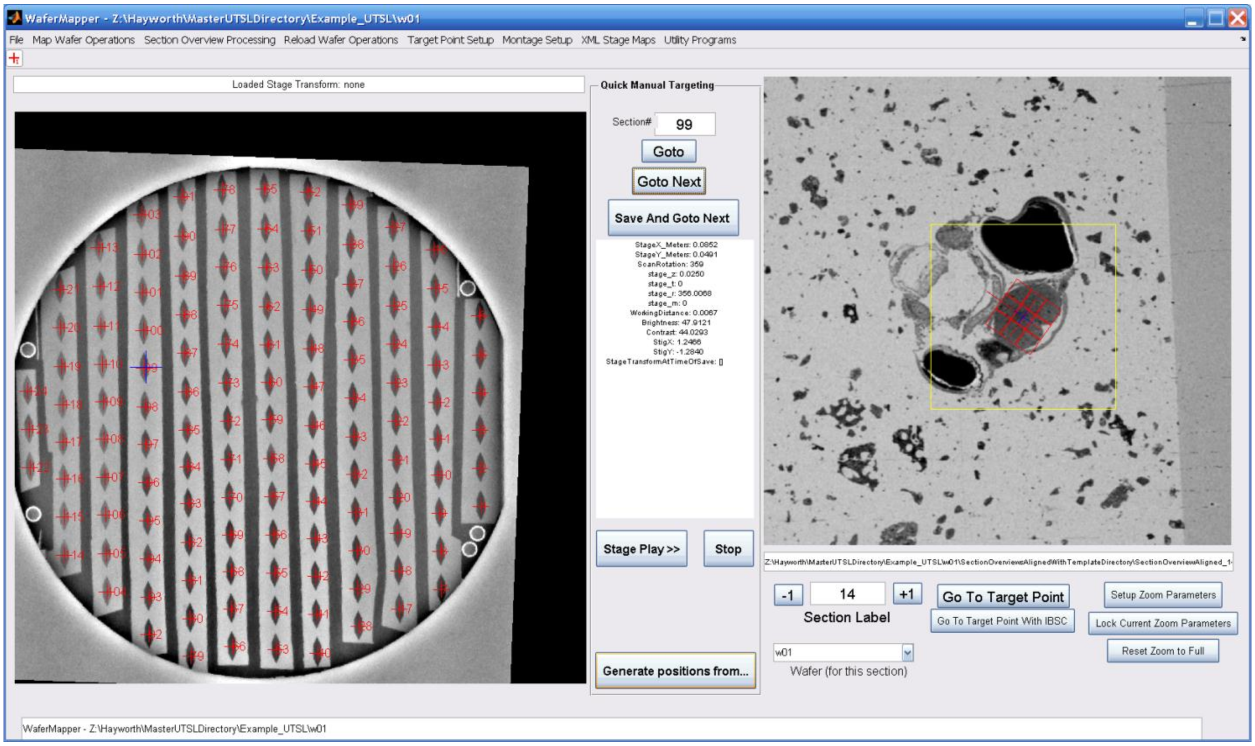
Wafer (for this section) w01

WaferMapper - Z:\Hayworth\MasterUTSLDirectory\Example_UTSL\w01

We will learn how to adjust the montage parameters in the next section, for now the only things that are important are the position of the red cross in the center of the montage, the rotation angle of the red cross, and the size of the yellow rectangle (the 'Low Res Box' used for image-based stage correction).

Just by placing the target point in the aligned section overviews you are now able to generate a set of 'CorrectedStagePostion_x.mat' files in the ManuallyCorrectedStragePositionsDirectory that are centered correctly on that region of the tissue. To see this, press the '**Generate Positions From…**' button in the Quick Manual Targeting menu and choose the '**Target Point**' option. This will overwrite any existing files with ones that are centered on this particular point in the tissue. Verify this by using the '**Goto**', '**Goto Next**', and '**Stage Play>>**' buttons. Note that these positions have adopted the scan rotation that was determined from the overview alignment step.

Now choose **Montage Setup -> Set Montage Parameters**. This opens up a complicated dialog that will be described in the next section. For now we just want to vary the 'North Angle' of the montage. Look for the edit box labeled 'North Angle (degrees)' and change this to 305 degrees and press the '**Update Display**' button. This will rotate the target point cross to be aligned with the main tissue axis. Close the dialog box. The result is shown below:



Loaded Stage Transform: none

Section# 99

StageX_Meters: 0.0852
StageY_Meters: 0.0491
ScanRotation: 359
stage_z: 0.0250
stage_t: 0
stage_r: 356.0068
stage_m: 0
WorkingDistance: 0.0067
Brightness: 47.9121
Contrast: 44.0293
StigX: 1.2466
StigY: -1.2840
StageTransformAtTimeOfSave: []

Z:\Hayworth\MasterUTSLDirectory\Example_UTSL\w01\SectionOverviewsAlignedWithTemplateDirectory\SectionOverviewAligned_1-

Section Label 14

Wafer (for this section) w01

WaferMapper - Z:\Hayworth\MasterUTSLDirectory\Example_UTSL\w01

Now if you press the '**Generate Positions From…**' button in the Quick Manual Targeting menu and choose the '**Target Point**' option the 'CorrectedStagePostion_x.mat' files that are generated have a scan rotation that includes this montage north angle correction. If you choose **XML Stage Maps -> Generate XML From Manually Corrected Stage Pos Files** the XML file that is generated will also include these correct scan rotations.

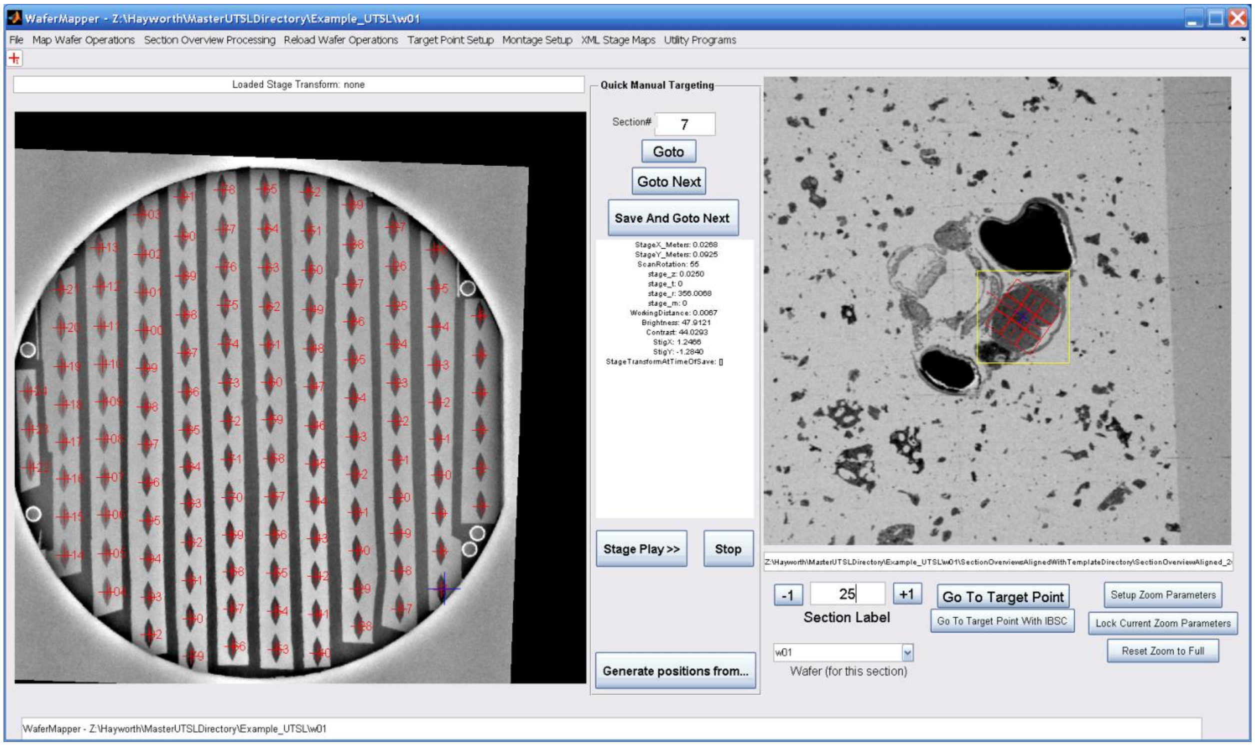
You can save a target point by choosing **Target Point Setup -> Save Target Point**.

1.11 Generating a list of aligned target points

As mentioned earlier, section overview alignment may not be precise enough for some imaging tasks especially in cases where there is significant distortion across the sections. To handle this, the WaferMapper program uses a form of local alignment performed around a region of interest centered on the target point.

The user chooses a target point in one section overview as described above. Then the program does a local alignment of all sections centered on that target point starting from the current section. It does this by cropping out a small region of the current aligned section overview (the region defined by the yellow box graphic). This cropped image is then compared to the next section to obtain the corresponding aligned region, and this cropping and alignment process proceeds (both forward and backward from the current section) across all sections and all wafers in the UTSL. The result is a new directory in the UTSL called ‘AlignedTargetListDirectory’ with a subdirectory (that you name) containing a set of images with labels like LowResAligned_w07_Section33.tif. All of these images are just cropped regions of the aligned section overviews, but they have been cropped such that they are all aligned with each other. A single data file ‘AlignedTargetList.mat’ is also generated in this directory. It contains the offsets needed to align each of these cropped regions. The WaferMapper program can use this offset information to help better direct the stage positioning of the SEM, it can also use the image files to do Image Based Stage Correction allowing the stage positioning to be corrected just before the final montage is imaged. Here is how it works:

The yellow box shown in the section overview display is called the ‘Target Point Low Res Box’. This box is typically set to 200 – 400 micron width by choosing **Target Point Setup -> Target Point Parameters**. Let’s go to section 25 of the stack by entering ‘25’ into the Section Label edit box and pressing enter. Then choose **Target Point Setup -> Target Point Parameters**, and set the Low Res Box width to 200microns. The resulting display looks like this:



Loaded Stage Transform: none

Section# 7

StageX_Meters: 0.0268
StageY_Meters: 0.0925
ScanRotation: 55
stage_z: 0.0250
stage_t: 0
stage_r: 356.0068
stage_m: 0
WorkingDistance: 0.0067
Brightness: 47.9121
Contrast: 44.0293
StigX: 1.2466
StigY: -1.2840
StageTransformAtTimeOfSave: []

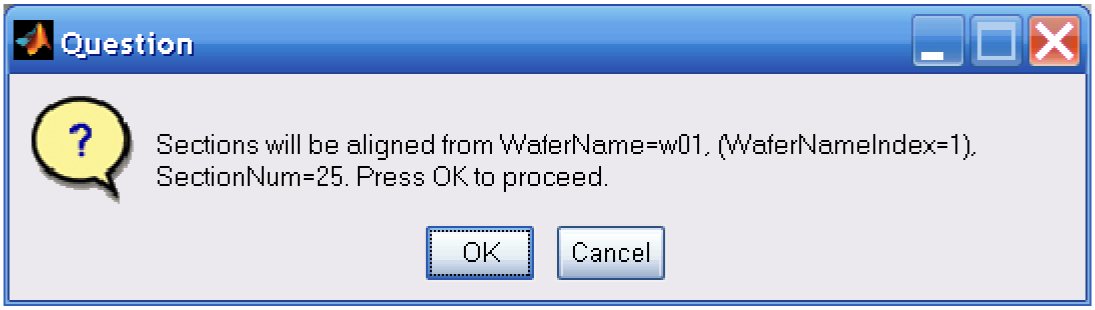
Z:\Hayworth\MasterUTSLDirectory\Example_UTSL\w01\SectionOverviewsAlignedWithTemplateDirectory\SectionOverviewAligned_2

Section Label: 25

Wafer (for this section): w01

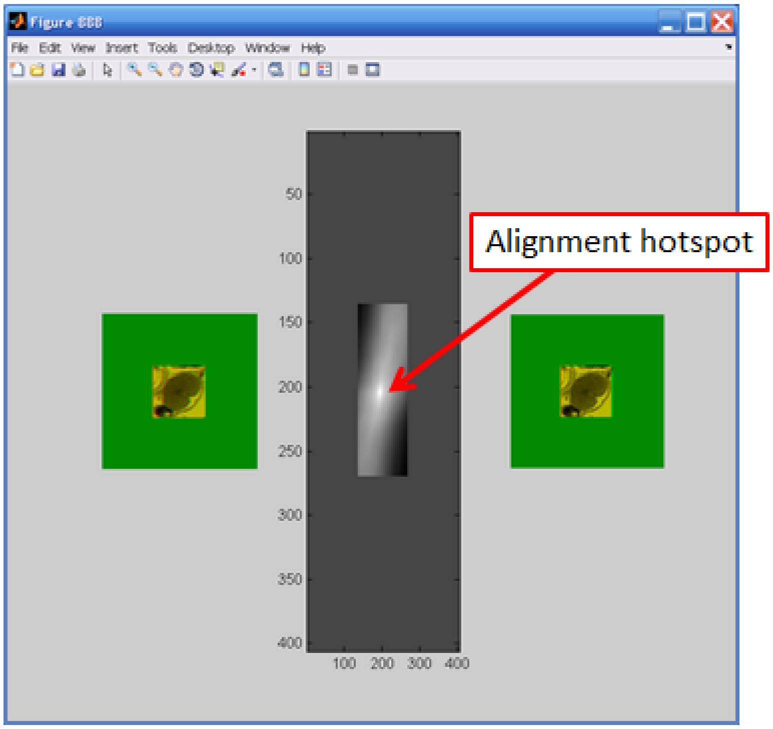
WaferMapper - Z:\Hayworth\MasterUTSLDirectory\Example_UTSL\w01

Now choose **Target Point Setup -> Generate and Save List Of Aligned Target Points**. The program will display the following message box:

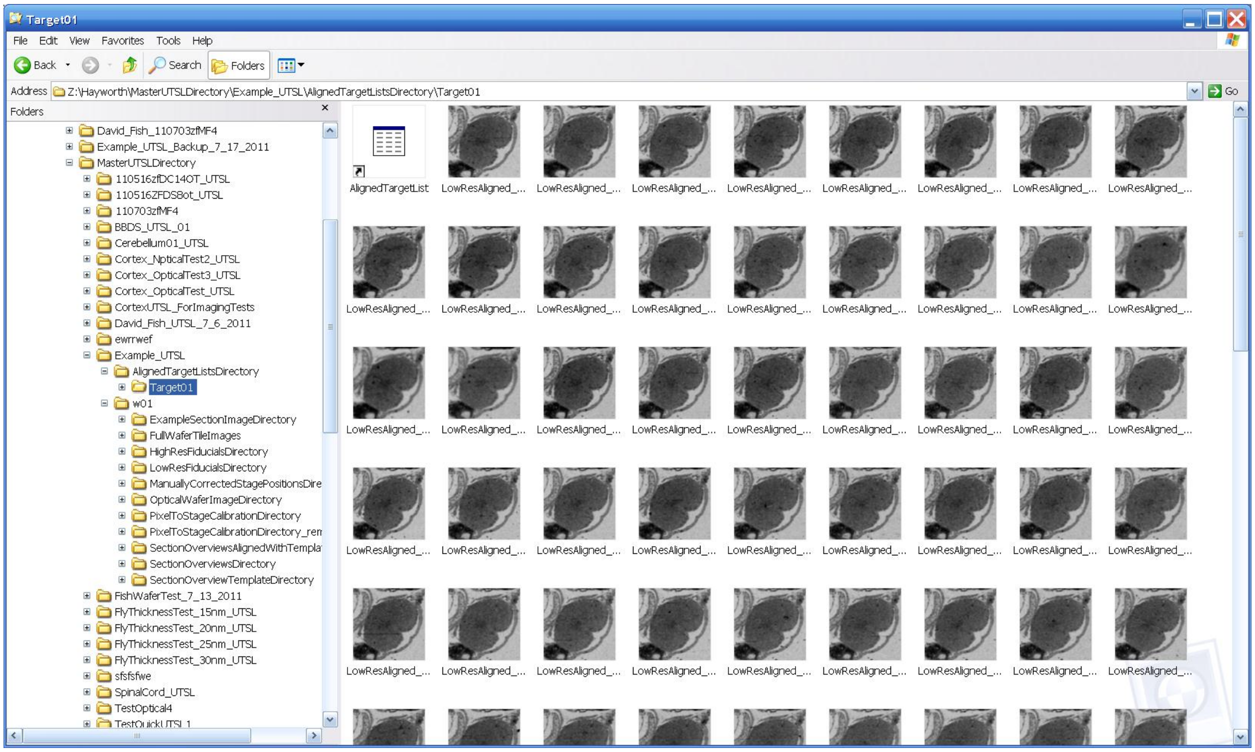


Sections will be aligned from WaferName=w01, (WaferNameIndex=1), SectionNum=25. Press OK to proceed.

This says that the program will align the target points starting from section#25 of wafer w01 (the current section) and proceeding in both the forward and backward directions. Press OK. A dialog box opens asking for a name for this AlignedTargetLIst – this will be the name of the subdirectory under the 'AlignedTargetListDirectory'. Enter 'Target01' and press OK. This starts an alignment process displaying Matlab figures like this:

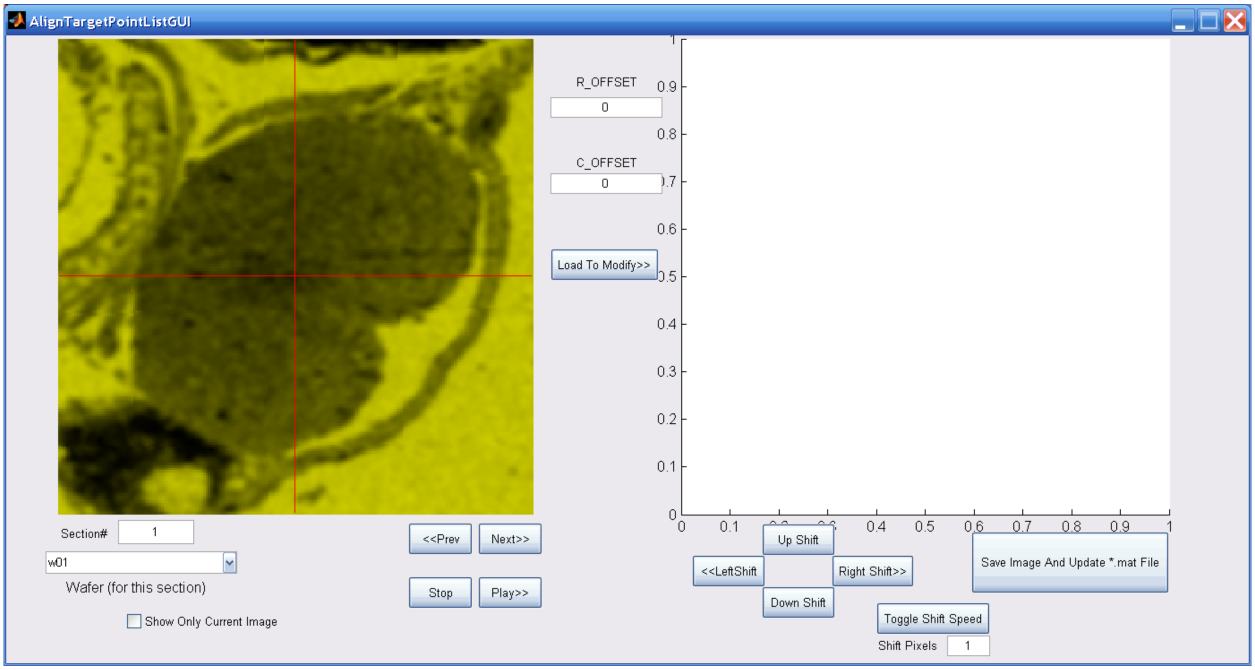


Again, a hotspot roughly in the center of the display means that the alignment process is working well. Once the alignment process is complete the Target01 directory looks like this:


Target01
File Edit View Favorites Tools Help
Back
Search
Folders
Address
Go
Folders
David_Fish_110703zfMF4
Example_UTSL_Backup_7_17_2011
MasterUTSLDirectory
110516zfDC14OT_UTSL
110516ZFDS8ot_UTSL
110703zfMF4
BBDS_UTSL_01
Cerebellum01_UTSL
Cortex_NpticalTest2_UTSL
Cortex_OpticalTest3_UTSL
Cortex_OpticalTest_UTSL
CortexUTSL_ForImagingTests
David_Fish_UTSL_7_6_2011
ewrrwef
Example_UTSL
AlignedTargetListsDirectory
Target01
w01
ExampleSectionImageDirectory
FullWaferTileImages
HighResFiducialsDirectory
LowResFiducialsDirectory
ManuallyCorrectedStagePositionsDire
OpticalWaferImageDirectory
PixelToStageCalibrationDirectory
PixelToStageCalibrationDirectory_rem
SectionOverviewsAlignedWithTempla
SectionOverviewsDirectory
SectionOverviewTemplateDirectory
FishWaferTest_7_13_2011
FlyThicknessTest_15nm_UTSL
FlyThicknessTest_20nm_UTSL
FlyThicknessTest_25nm_UTSL
FlyThicknessTest_30nm_UTSL
sfsfsfwe
SpinalCord_UTSL
TestOptical4
TestQuickUTSL1
AlignedTargetList
LowResAligned_...

Address: Z:\Hayworth\MasterUTSLDirectory\Example_UTSL\AlignedTargetListsDirectory\Target01

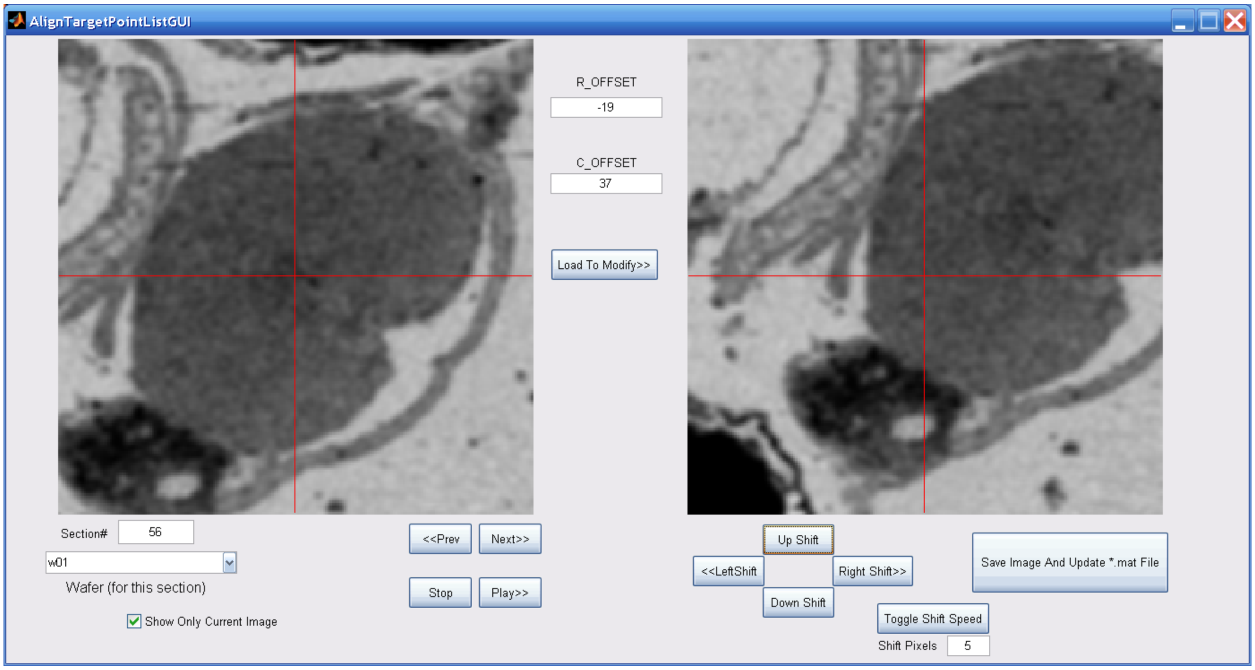
A quick look at the thumbnails shows that the alignment succeeded; however there is a tool that can be used to check and correct this alignment if needed. Choose **Target Point Setup -> Check and Correct Target Point Alignment**, and choose the just created ‘Target01’ aligned target point list from the list box that is displayed. This opens the AlignedTargetPointListGUI:



R_OFFSET: 0
C_OFFSET: 0
Section#: 1
w01
Shift Pixels: 1

With this GUI you can quickly (because they are small files) play through the entire aligned target list. You can also correct any misalignments that are seen.

First check the **Show Only Current Section** check box – this removes the bicolor display showing the current section and the previous averaged sections that it was aligned to. Now click the '**Play>>**' button. This will quickly play through all of the cropped images across all wafers. If a problem is seen then use the **Secton#** and **Wafer (for this section)** list box to display the problem section. Then press the '**Load To Modify>>**' button to load this image into the right hand part of the GUI.


AlignTargetPointListGUI

R_OFFSET: -19

C_OFFSET: 37

Section#: 56

w01

Wafer (for this section)

☑ Show Only Current Image

Shift Pixels: 5

Use the '**Up Shift**', '**Down Shift**', '**Left Shift**', and '**Right Shift**' buttons to correct any misalignment (Note: this is actually taking a new crop of the aligned section overview on the fly). This GUI does not allow for rotations because these should have been taken care sufficiently at the section overview alignment stage. Press the '**Toggle Shift Speed**' to switch between coarse and fine shift movements. When you are satisfied with the alignment press the '**Save Image And Update *.mat File**' button. When all images are aligned simply close the AlignTargetPointsListGUI.

When you are done checking the alignment choose **Target Point Setup -> Load Existing List Of Aligned Target Points** and select the just created aligned target point list. Now if you press the **'Go To Target Point'** button beneath the section overview display the stage will move to the current target point's stage position with the additional local alignment corrections.

You can use this new corrected alignment as the basis of a set of manual alignment files and/or as an XML stage map for Fibics by pressing the '**Generate positions from…**' button and choosing the '**Aligned Target Points**' option. You can use the '**Stage Play>>**' button to review these and use ctrl+tab in the SmartSEM window and the **'Save And Goto Next'** button to perform any manual corrections needed. Finally you can choose **XML Stage Maps -> Generate XML From Manually Corrected Stage Pos Files** to generate an XML stage map for Fibics from these aligned target points.

1.12 Image Based Stage Correction

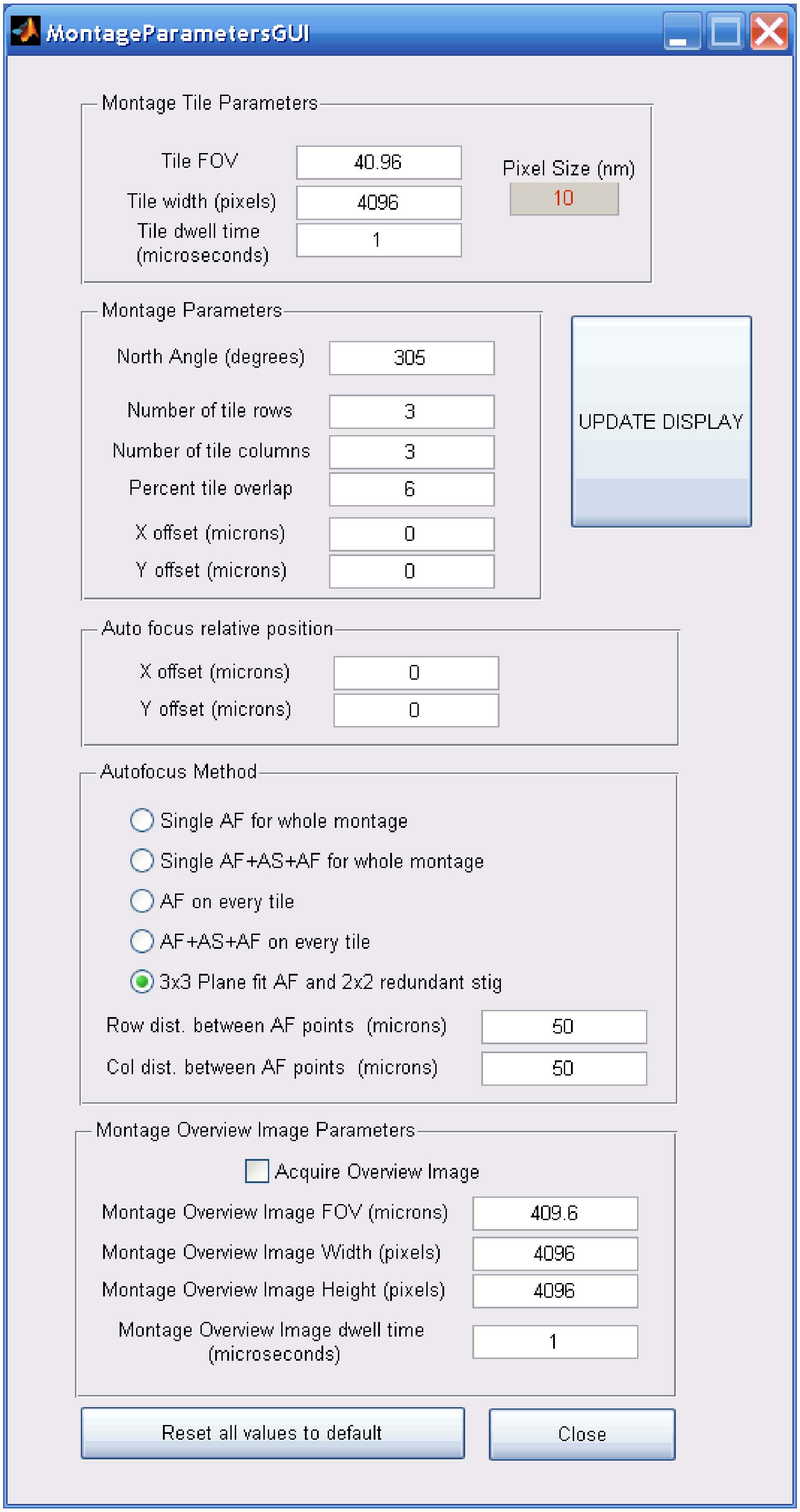
Once you have created and loaded a list of aligned target points (as done in the previous section) you can use Image Based Stage Correction (IBSC) to obtain the best stage positioning accuracy. Press the '**Go To Target Point With IBSC'** button beneath the section overview display. The stage will move to the current target position and then will acquire an image with the same field of view and pixel size as the cropped images in the aligned target list. It will software align these two images determining the offset, and then will move the stage to compensate for any offset. The stage is left in this IBSC corrected position.

Conditions under which IBSC may fail – T.B.D.

(Note: I plan to add a feature that allows IBSC to generate a set of manual alignment files, but this is not yet completed.)

1.13 Acquiring a montage

To graphically setup a montage choose **Montage Setup -> Set Montage Parameters**. The following dialog box pops up:



Montage Tile Parameters

Tile FOV	40.96
Tile width (pixels)	4096
Tile dwell time (microseconds)	1
Pixel Size (nm)	10

Montage Parameters

North Angle (degrees)	305
Number of tile rows	3
Number of tile columns	3
Percent tile overlap	6
X offset (microns)	0
Y offset (microns)	0

Auto focus relative position

X offset (microns)	0
Y offset (microns)	0

Autofocus Method

○ Single AF for whole montage

○ Single AF+AS+AF for whole montage

○ AF on every tile

○ AF+AS+AF on every tile

● 3x3 Plane fit AF and 2x2 redundant stig

Row dist. between AF points (microns)	50
Col dist. between AF points (microns)	50

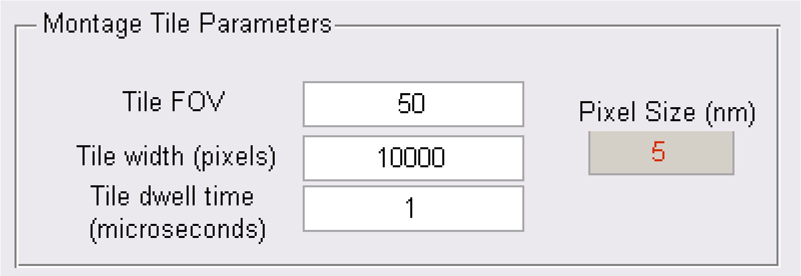
Montage Overview Image Parameters

□ Acquire Overview Image

Montage Overview Image FOV (microns)	409.6
Montage Overview Image Width (pixels)	4096
Montage Overview Image Height (pixels)	4096
Montage Overview Image dwell time (microseconds)	1

This Montage Parameters dialog is used to setup all the features of the montage. Let’s walk through a quick example.

First enter a **Tile FOV** of 50 microns and a **Tile width (in pixels)** of 10000. Pressing enter, the **Pixel Size (nm)** box will be updated to reflect the resulting pixel size of 5nm. Set the **Tile dwell time** to 1.0 microsecond. These will define the main imaging parameters of each tile image.



Montage Tile Parameters

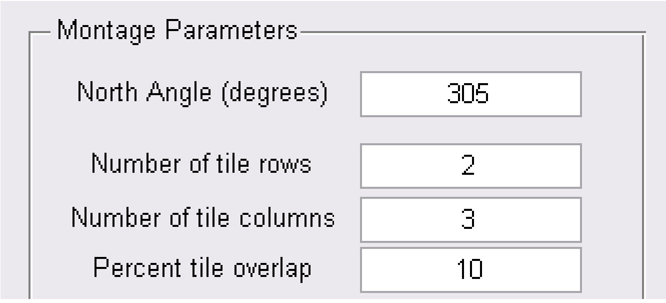
Tile FOV 50

Tile width (pixels) 10000

Tile dwell time (microseconds) 1

Pixel Size (nm) 5

Next setup the montage parameters. Set the North Angle (degrees) to 305, and the Number of tile rows to be 2, and the number of tile columns to be 3. Change the percent tile overlap to 10. Pressing the ‘**UPDATE DISPLAY**’ button will cause the Section Overview Display in the main GUI to update its graphic representation of the montage.



Montage Parameters

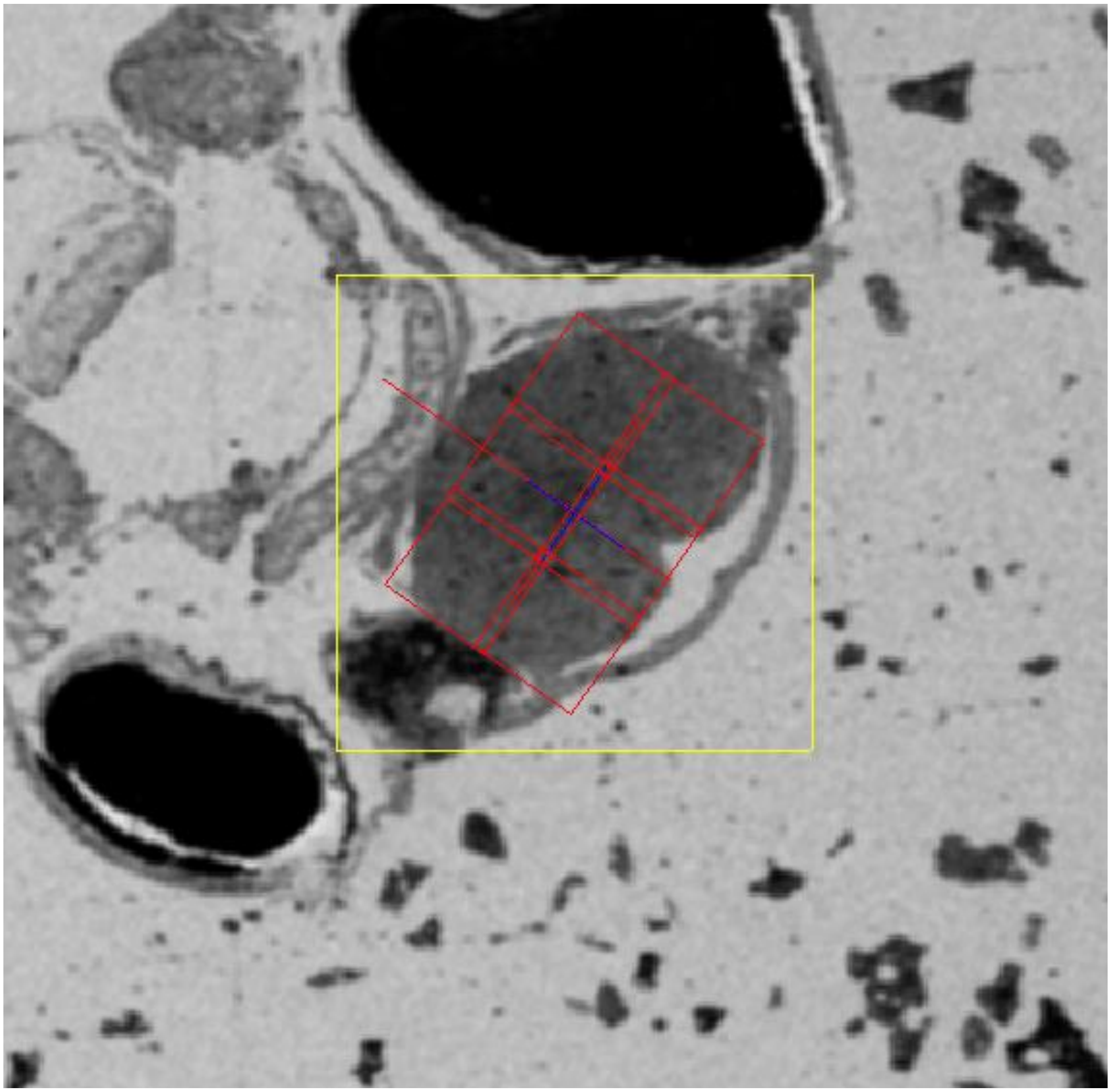
North Angle (degrees) 305

Number of tile rows 2

Number of tile columns 3

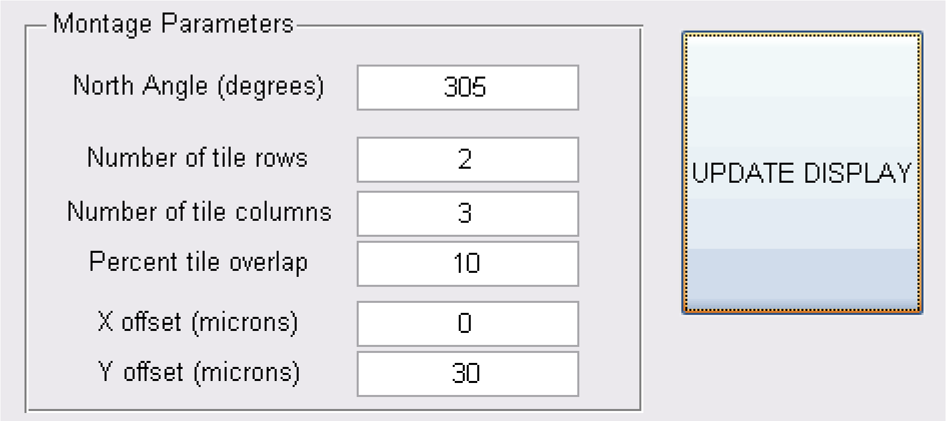
Percent tile overlap 10

Now it looks like this:



Note that the individual tiles of the montage are display in the positions they will be imaged and you can visualize the degree of overlap etc.

Sometimes one does not want the montage to be centered directly on the target point. To offset the entire montage in the Y direction type ‘30’ into the ‘**Y offset microns**’ edit box in the Montage Parameters bounding group.



Montage Parameters

North Angle (degrees) 305

Number of tile rows 2

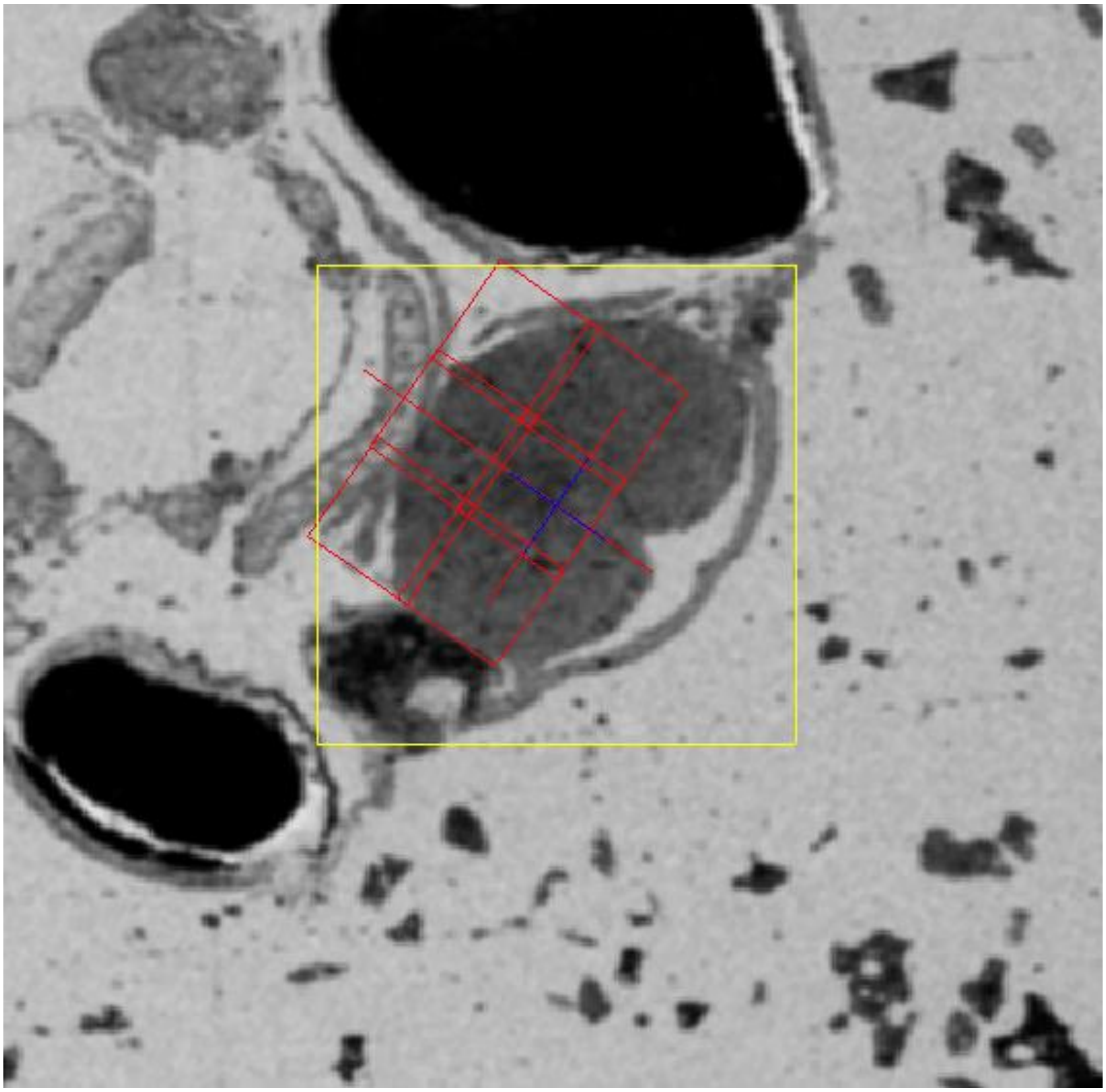
Number of tile columns 3

Percent tile overlap 10

X offset (microns) 0

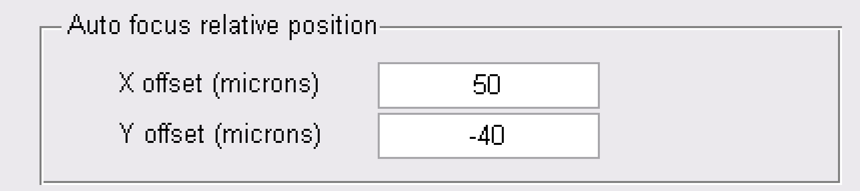
Y offset (microns) 30

The result is this:



You can see that the entire montage has been offset in the 'North' (i.e. Y) direction of the montage.

The blue cross in the center is the position that the main autofocus for the montage will be performed at. If your tissue has a hole in its center that would be problematic for the autofocus routine you can offset this autofocus position by typing values into the **X offset** and **Y offset** edit boxes in the **Auto focus relative position** group. Type in the following values:

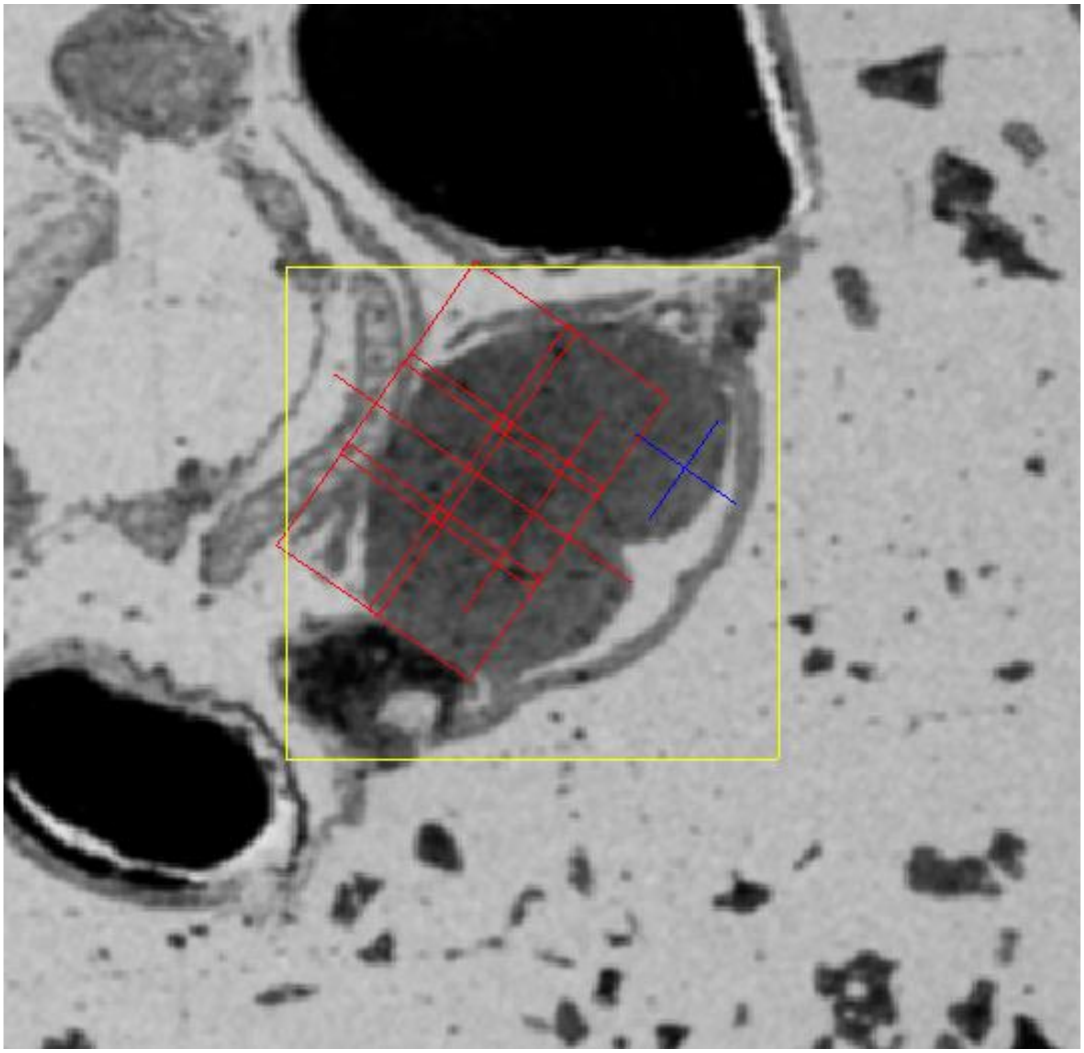


Auto focus relative position

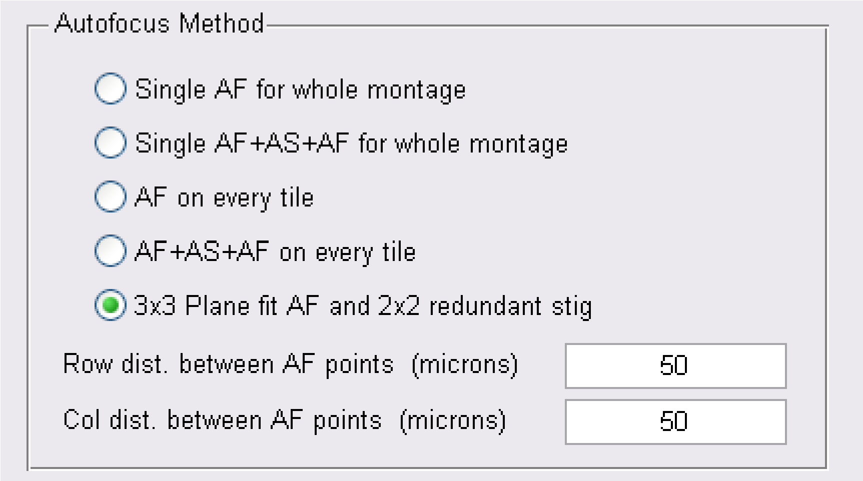
X offset (microns) 50

Y offset (microns) -40

The blue autofocus cross has now moved:



The next set of parameters lets the user determine how the autofocus will be performed on the montage.



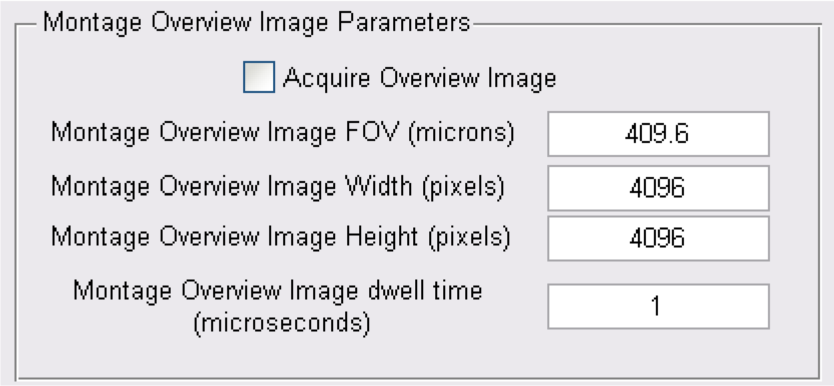
Autofocus Method

○ Single AF for whole montage
○ Single AF+AS+AF for whole montage
○ AF on every tile
○ AF+AS+AF on every tile
● 3x3 Plane fit AF and 2x2 redundant stig

Row dist. between AF points (microns)	50
Col dist. between AF points (microns)	50

If the user chooses '**Single AF for whole montage**' or '**Single AF+AS+AF for whole montage**' a single autofocus (or autofocus +autostig + autofocus) will be performed at the point of the blue cross and this WD and stig value will be used for all tiles of the montage. If the '**AF on every tile**' or '**AF+AS+AF on every tile**' is chosen then the blue cross position is ignored. If the '**3x3 Plane fit AF and 2x2 redundant stig**' option is chosen then the microscope will do a grid of 3x3 autofocus points centered on the blue cross position with distances between the autofocus points set by the edit boxes below. These 9 working distances (minus any outliers) will then be used to fit a plane from which the actual tile working distances will be set. The first 2x2 points of this 3x3 grid include autostig calls, and the median of these four values (of x and y sitg) is used to set the stigs for the entire montage.

Finally the user can request that a single overview image be taken for each montage with parameters set but the following boxes:



Montage Overview Image Parameters

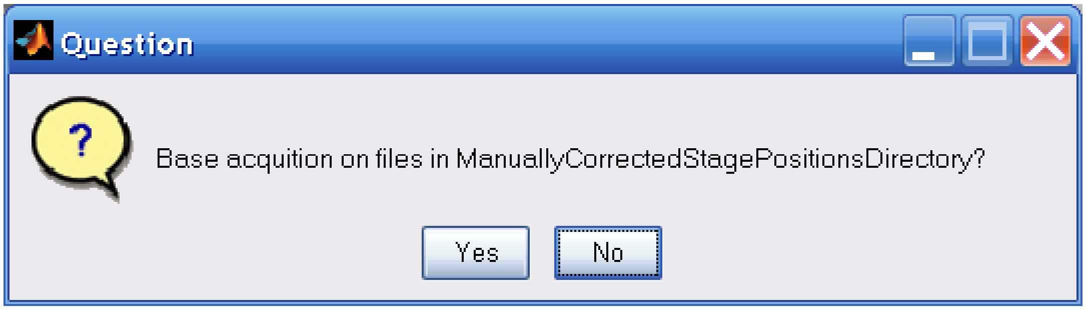
□ Acquire Overview Image

Montage Overview Image FOV (microns)	409.6
Montage Overview Image Width (pixels)	4096
Montage Overview Image Height (pixels)	4096
Montage Overview Image dwell time (microseconds)	1

To save these montage parameters choose **Montage Setup -> Save Montage Setup**.

Acquiring a montage:

To acquire a montage choose **Montage Setup -> Acquire Montage Stack Main**. The following question box pops up:

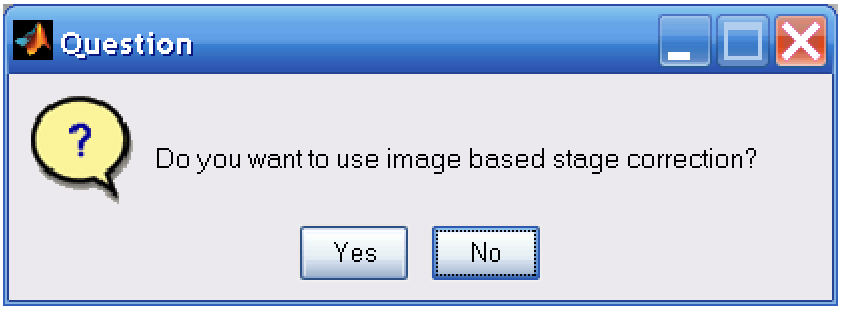


Question

Base acquition on files in ManuallyCorrectedStagePositionsDirectory?

If the user presses 'Yes' the montage will be performed at the points defined by the 'CorrectedStagePostion_x.mat' files in the ManuallyCorrectedStragePositionsDirectory. In this case only those sections that have 'CorrectedStagePostion_x.mat' files will be imaged.

If the user presses 'No' then the following question box pops up:

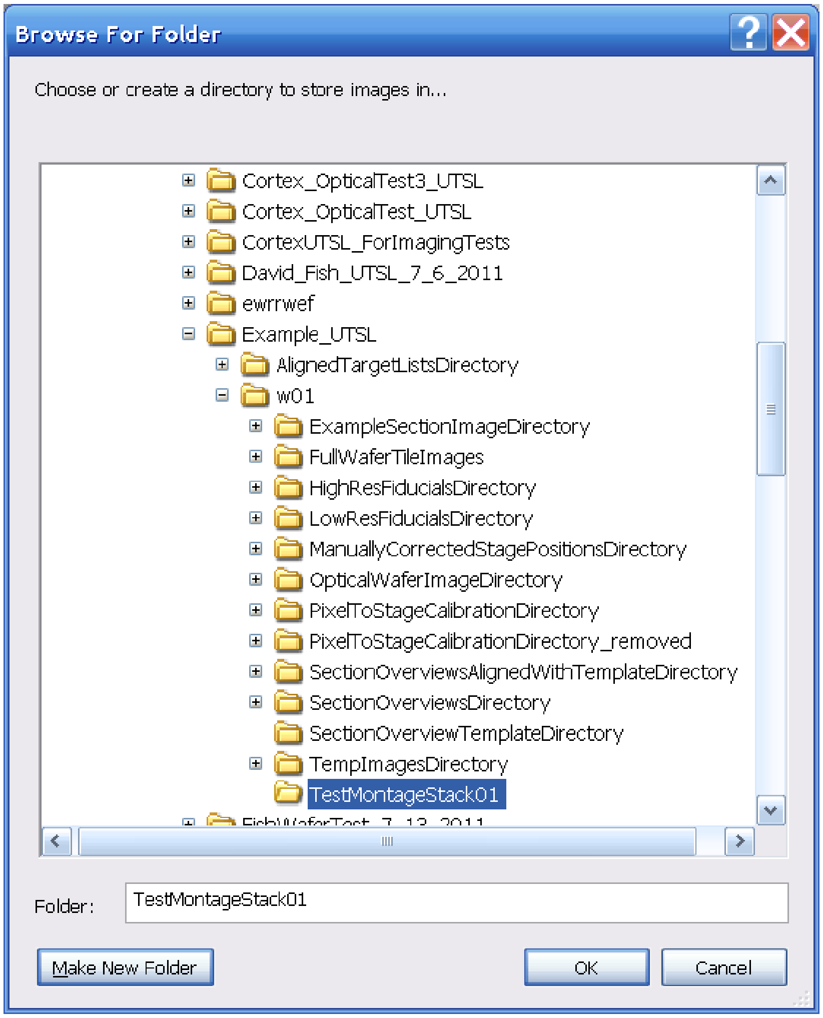


Question

Do you want to use image based stage correction?

This asks the user whether the stage position should be based solely on the 'open loop' positons determined by the currently loaded aligned target list, or whether there should be an IBSC correction performed at each section before the montage is acquired.

After these choices are made a popup displays the current wafer name (as a last check to the user to remind them what wafer the program thinks is loaded). Pressing OK then opens up a dialog box for the user to choose the directory where the montage images will be saved:

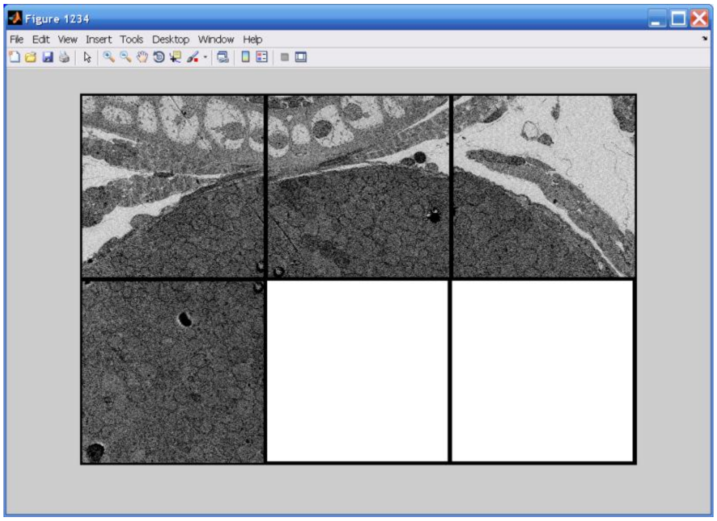

Browse For Folder
Cortex_OpticalTest3_UTSL
Cortex_OpticalTest_UTSL
CortexUTSL_ForImagingTests
David_Fish_UTSL_7_6_2011
ewrrwef
Example_UTSL
AlignedTargetListsDirectory
w01
ExampleSectionImageDirectory
FullWaferTileImages
HighResFiducialsDirectory
LowResFiducialsDirectory
ManuallyCorrectedStagePositionsDirectory
OpticalWaferImageDirectory
PixelToStageCalibrationDirectory
PixelToStageCalibrationDirectory_removed
SectionOverviewsAlignedWithTemplateDirectory
SectionOverviewsDirectory
SectionOverviewTemplateDirectory
TempImagesDirectory
TestMontageStack01
FishWaferTest_7_13_2011
Make New Folder
OK
Cancel

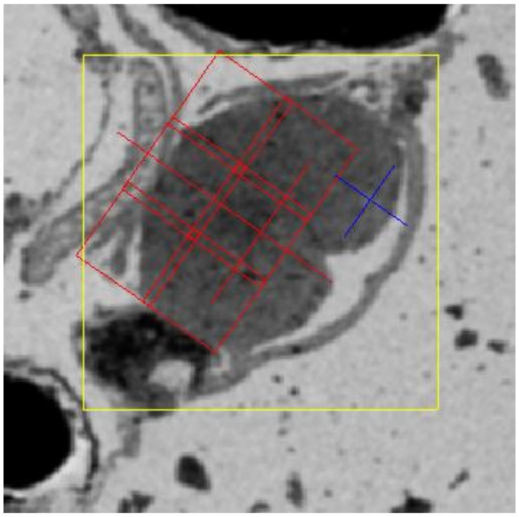
Choose or create a directory to store images in...

Folder: TestMontageStack01

Warning: If you use ‘**Make New Folder**’ to create a new directory to save the images in, make sure that it is really the one chosen before pressing OK (this Matlab dialog is a bit tricky to use).

Press OK. The program will now automatically direct imaging until all sections are complete. During imaging a Matlab figure will pop up which displays the montage tiles side by side with black borders between them. Those yet to be image (or those dropped out, see below) are displayed as solid white tiles:



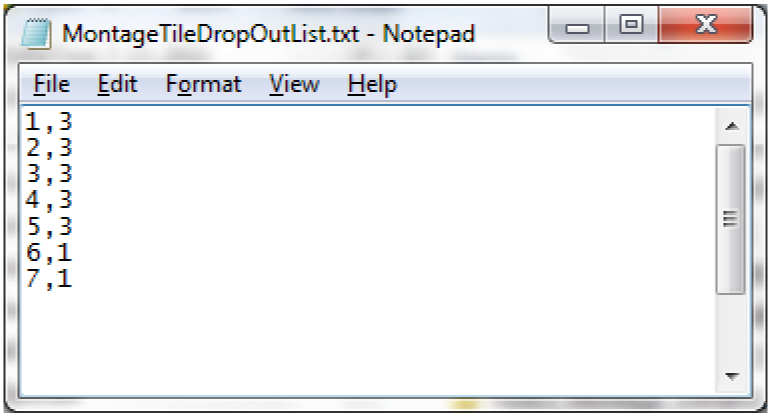


The directory structure for storing the tile images consists of a having a subdirectory (named for example ‘w01_Sec35_Montage’) containing each section’s tile images.

To stop an imaging run simply do a ctrl+c in the Matlab command window. To restart simply choose **Montage Setup -> Acquire Montage Stack Main** and choose the same directory that you did before. The program will pick up right where it left off because before it moves to a stage position and takes an image it makes sure that the file does not exist already. This is also a simple way to do retakes – just rename the problem tile images or montage directories and rerun the montage – only those missing images will be retaken.

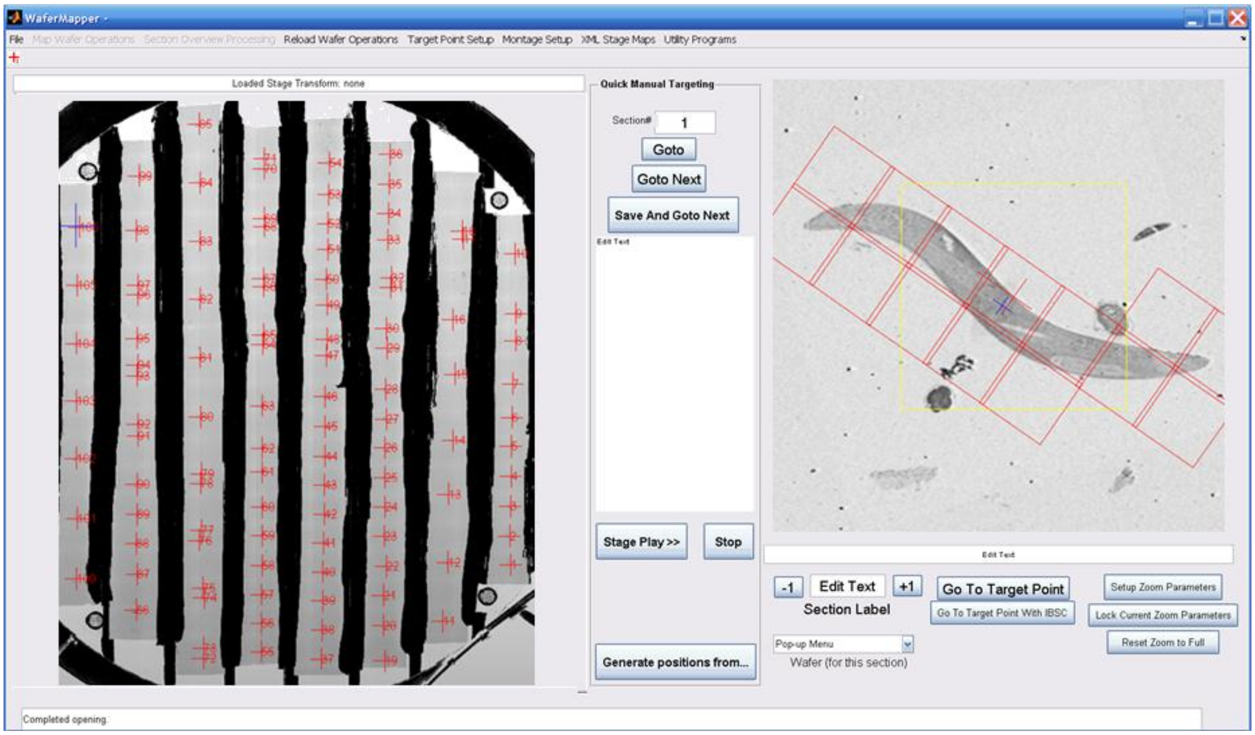
1.14 Specifying drop-out tiles

When imaging a montage with many tiles it is often useful to be able to skip some of the tiles that do not contain tissue or that contain tissue which is not currently of interest. In WaferMapper this is accomplished by creating a text file called ‘MontageTileDropOutList.txt’ and putting it directly in the wafer directory. The format of the file is to list tiles in row,col format each on a separate line. This is shown in the following screen capture of the text file:


MontageTileDropOutList.txt - Notepad
File Edit Format View Help

```
1,3
2,3
3,3
4,3
5,3
6,1
7,1
```

Here is the resulting dropout pattern corresponding to the above text file:



Loaded Stage Transform: none

Completed opening

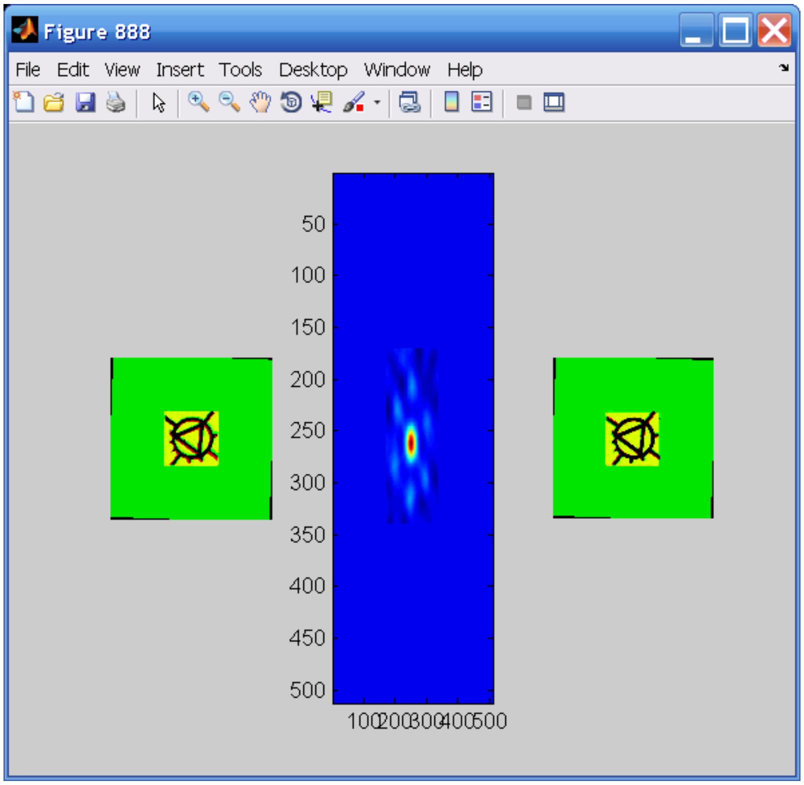
1.15 Reloading a wafer with automatic stage correction

When a wafer is taken out of the SEM and put back in at a later date the original stage section mapping coordinates will no longer be valid. To correct for this the WaferMapper software has automatic routines that will reimage the low resolution and high resolution fiducials that were imaged during the original mapping process and it will generate a stage correction file that will compensate for any misalignment when the wafer is reloaded.

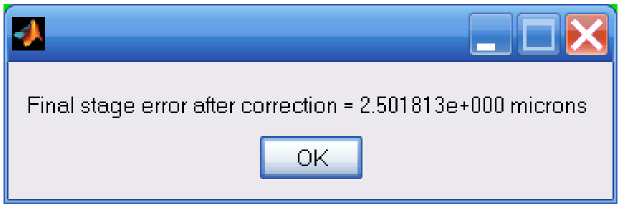
Reloading process:

Reload the wafer in the SEM attempting to get as close as possible to the original positioning on the SEM wafer holder. Use WaferMapper to open up the UTSL and wafer directories associated with this wafer. Choose **Map Wafer Operations -> Free View** and use the crosshairs to click on one of the fisucials in the wafer overview display. The microscope will be driven to the original stage position of this fiducial but due to the reload it may be offset. Rotate (without X-Y translation, just stage rotation) the stage manually (with the joystick) to correct as much of the offset as possible. Use free view to go to another of the fiducials – it should also be close. Press 'esc' to remove the free view crosshairs.

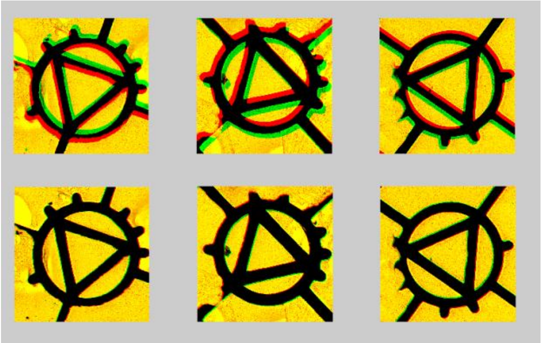
Now adjust the focus and contrast settings to make the fiducial images clear and choose **Reload Wafer Operations -> Do All Steps For Stage Correction**. This will start an ~7 minute process in which the low res fiducials are retaken and then compared to the original images and a rigid transformation of stage coordinates is fit. Then this coarse correction is used to find the high res fiducials positions and these are retaken and used to determine the final rigid stage transformation. Finally the high res images are retaken with the new stage transformation and used to determine the overall accuracy of the correction. The screen capture below shows the alignment process in action. As before a clear hotspot in the middle of the images is an indicator that the process is working.



At the end of the process the following message box is displayed and a figure that shows the before and after positioning of the high res fiducials:

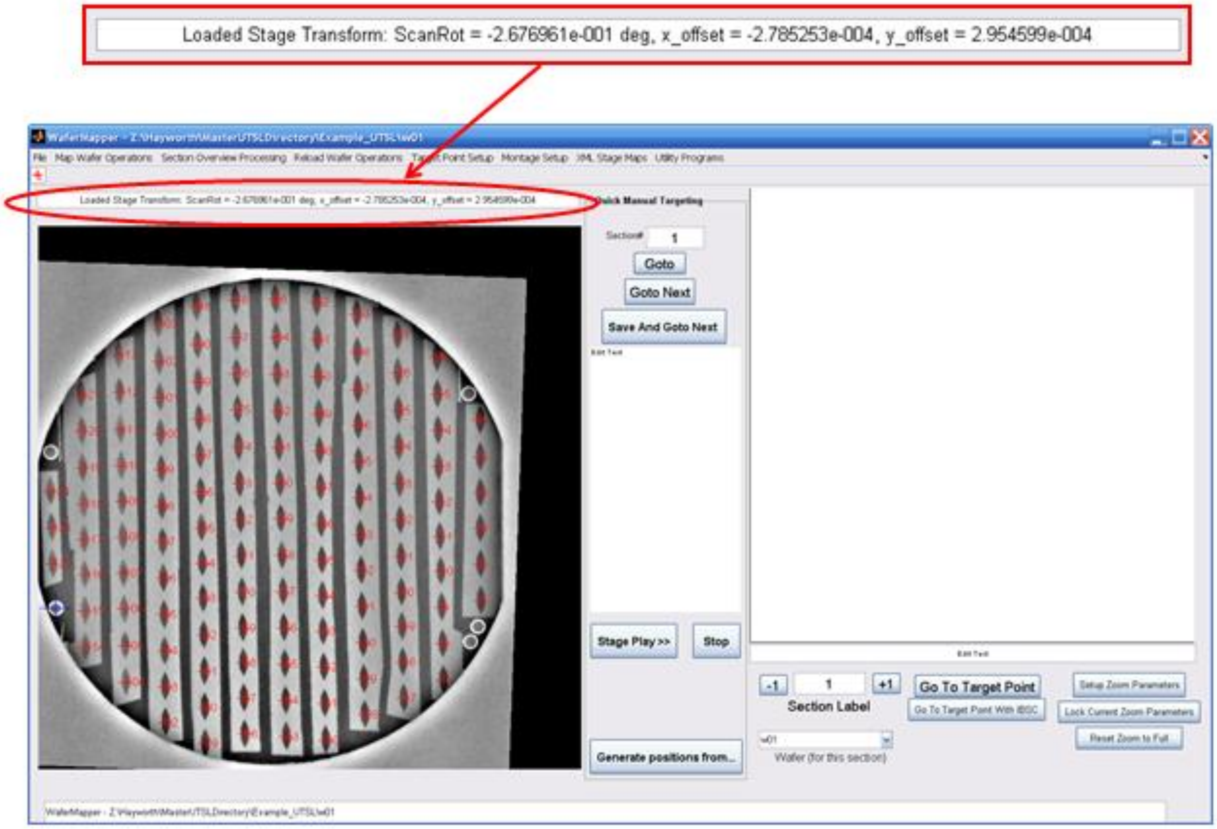


Final stage error after correction = 2.501813e+000 microns



The end result is a file in the ‘ReImageHighResFiducialsDirectory’ subdirectory called ‘HighResStageTransformation.mat’. This file is only good for this imaging session, but if you shut down the program and restart without removing the wafer from the SEM then you can use the **Reload Wafer Operations -> Reload Stage Correction** menu item to load this file.

The currently active stage correction is displayed at the very top of the whole wafer overview display shown below:



```
Loaded Stage Transform: ScanRot = -2.676961e-001 deg, x_offset = -2.785253e-004, y_offset = 2.954599e-004
```

To check how well this stage correction did you can choose **Reload Wafer Operations -> Free View With Stage Correction**. This operates just like the free view in the Map Wafer Operations menu but this one always uses the currently loaded stage transformation whereas the other does not.

The program is written to intelligently use this stage corrections in all the functions that need it. For example, you can generate a new XML stage map and its positions will be generated to be correct for the current wafer reload. Also, if you use the Quick Manual Targeting menu to perform ctrl+tab adjustment to new or old ‘CorrectedStagePostion_x.mat’ files the stage correction will be correctly handled.

Appendix A. Documentation for the Visual Basic COM Wrapper of the Zeiss and Fibics API Functions

Zeiss provides an Application Programming Interface (API) with its SEM so that third party developers can automate the SEM's functions. This Zeiss API (see pdf document SmartSEM_Remote_API_Manual.pdf) is in the form of an ActiveX control (CZEMAPI.OCX) that can be directly used in programs like VB, C++, etc. Unfortunately the functions in this Zeiss API use a variant data type passed by reference and because of this Matlab cannot directly access the API. I have therefor written a 'wrapper' COM library around this Zeiss API using Visual Basic. This wrapper provides Matlab users with functions taking specific data types and passes these on to the Zeiss API functions.

The Fibics frame grabber hardware/software also provides an API as a Type Library, registered as "NPVE3Z Library". I encountered some problems trying to access these directly through Matlab as well so I included wrapper functions to them in the same COM wrapper as the Zeiss API.

The following is a complete listing of the functions made available to Matlab via my VBComObjectWrapperForZeissAPI Class Library:

```
Function Initialise(ByVal Machine As String) As Integer

Function InitialiseRemoting() As Integer

Function GetLimits_MaxValue(ByVal Param As String) As Single

Function GetLimits_MinValue(ByVal Param As String) As Single

Function GetMag() As Double

Function Get_ReturnTypeSingle(ByVal CommandStr As String) As Single

Function Get_ReturnTypeString(ByVal CommandStr As String) As String

Function MoveStage(ByVal x As Single, ByVal y As Single, ByVal z As Single,
ByVal t As Single, ByVal r As Single, ByVal m As Single) As Integer

Function Grab(ByVal xoff As Short, ByVal yoff As Short, ByVal width As Short,
ByVal height As Short, ByVal reduction As Short, ByVal Filename As String) As
Integer

Function GetLastError() As String

Function ClosingControl() As Integer

Function GetLastRemotingConnectionError() As String

Function SetSuppressRemotingConnectionErrors() As Integer
Function Set_PassedTypeSingle(ByVal CommandStr As String, ByVal PassedValue
As Single) As Integer

Function Set_PassedTypeString(ByVal CommandStr As String, ByVal PassedValue
As String) As Integer

Function Execute(ByVal CommandStr As String) As Integer

Function Fibics_Initialise() As Integer

Function Fibics_AcquireImage(ByVal W As Long, ByVal H As Long, ByVal Dwell As
Single, ByVal Filename As String) As Integer

Function Fibics_IsBusy() As Boolean

Function Fibics_ReadFOV() As Single

Function Fibics_WriteFOV(ByVal FOV As Single)

Function Fibics_Cancel()
```

Using the COM Wrapper in Matlab:

To use the COM wrapper in Matlab first call the following function to create the object MyCZEMAPIClass in the Matlab workspace:

```
MyCZEMAPIClass =
actxserver('VBComObjectWrapperForZeissAPI.KHZeissSEMWrapperComClass');
```

Then initialize as follows:

```
MyCZEMAPIClass.InitialiseRemoting();
```

This initializes the communications between the ActiveX control and the SmartSEM server on the SEM PC. This must be called before any other Zeiss API functions.

The key functions that a typical Matlab SEM automation program will be using are:

```
Function Get_ReturnTypeSingle(ByVal CommandStr As String) As Single

Function Get_ReturnTypeString(ByVal CommandStr As String) As String

Function Set_PassedTypeSingle(ByVal CommandStr As String, ByVal PassedValue
As Single) As Integer

Function Set_PassedTypeString(ByVal CommandStr As String, ByVal PassedValue
As String) As Integer

Function Execute(ByVal CommandStr As String) As Integer
Function MoveStage(ByVal x As Single, ByVal y As Single, ByVal z As Single,
ByVal t As Single, ByVal r As Single, ByVal m As Single) As Integer
```

These functions are wrappers around the Zeiss API functions Get(), Set(), Execute(), and MoveStage(). The only difference between calling the wrapper functions vs. the original API functions is that if one is calling a Get() command that returns a string one needs to use the Get_ReturnTypeString() function etc. See the pdf document SmartSEM_Remote_API_Manual.pdf for a listing of all command arguments that can be used with these functions.

Here is some example Matlab code using these functions:

```
MyCZEMAPIClass =
actxserver('VBComObjectWrapperForZeissAPI.KHZeissSEMWrapperComClass');

MyCZEMAPIClass.InitialiseRemoting();

MyMag = MyCZEMAPIClass.Get_ReturnTypeSingle('AP_MAG');

MyCZEMAPIClass.Set_PassedTypeSingle('AP_MAG',100);

MyCZEMAPIClass.Execute('CMD_AUTO_FOCUS_FINE');
```

To use the Fibics API from Matlab you must have already initialized the Zeiss API as shown above. You must then make sure that the ATLAS software GUI is shut down and then call:

```
MyCZEMAPIClass.Fibics_Initialise();
```

The Visual basic code behind this call is the following:

```
Public Function Fibics_Initialise() As Integer

  veobj = New NPVE3Z.FibicsSEMVE()

  Return 0
End Function
```

As you can see this call creates a new `NPVE3Z.FibicsSEMVE` object within the COM wrapper called veobj. This object is used for all subsequent calls to the Fibics API.

The key Fibics functions that a typical Matlab SEM automation program will be using are:

```
Function Fibics_Initialise() As Integer

Function Fibics_AcquireImage(ByVal W As Long, ByVal H As Long, ByVal Dwell As
Single, ByVal Filename As String) As Integer

Function Fibics_IsBusy() As Boolean

Function Fibics_ReadFOV() As Single
Function Fibics_WriteFOV(ByVal FOV As Single)
```

Here is some example Matlab code using these functions to grab an image:

```
FOV_microns = 4096;

MyCZEMAPIClass.Fibics_WriteFOV(FOV_microns);

ImageWidthInPixels = 4096;

ImageHeightInPixels = 4096;

DwellTimeInMicroseconds = 1;

FileNameStr = 'C:\TestImage.tif';

MyCZEMAPIClass.Fibics_AcquireImage(ImageWidthInPixels, ...

   ImageHeightInPixels, DwellTimeInMicroseconds, FileNameStr);

while(MyCZEMAPIClass.Fibics_IsBusy)

        pause(1);

end
```

See the Fibics API documentation (ATLAS_API_Draft_User_Guide_April_2010.pdf) for more information on these functions.

NOTE: I have noticed a bug in the Fibics FOV-to-Magnification calibration such that the mag for a particular FOV will be slightly different based on what mag the SEM was in when Fibics initialized. The solution to this that I have been using is to always call the Zeiss API to set the microscope to 25x just before calling MyCZEMAPIClass.Fibics_Initialise();

****

Creating and installing the VBComObjectWrapperForZeissAPI Class Library:

Note: I have already created this COM library and installed it on the Fibics PC. The following steps are listed here simply as documentation on how this was done.

1. Create in VB.net a new project of type Class Library
2. Delete the existing Class1.vb from the solution explorer
3. Right click on project and Add.. New... COM Object. Call it "KHZeissSEMWrapperComClass.vb"
4. Project -> Add Reference... under COM add CZ EM API OLE Control

5. Add a form to the project (only way to get to activex control). Call it Form1
6. In the Form1.vb [Design] use the toolbox to drag a CZ EM API control onto the form
7. In the top of the KHZeissSEMWrapperComClass.vb put the line: "Dim MyForm As New Form1"
8. Any function call to the api will use the following syntax: MyForm.AxApi1.InitialiseRemoting()
9. Build this DLL and then register using the RegAsm.exe program with /tlb option: C:\>C:\WINDOWS\Microsoft.NET\Framework\v2.0.50727\RegAsm C:\Users\Hayworth\VBComObjectWrapperForZeissAPI\VBComObjectWrapperForZeissAPI\bin\Debug\VBComObjectWrapperForZeissAPI.dll /tlb
10. In Matlab create an object with this: MyCZEMAPIClass = actxserver('VBComObjectWrapperForZeissAPI.KHZeissSEMWrapperComClass')