

Stage Tracker – DM plugin to monitor x/y stage positions

About

Stage Tracker is a DigitalMicrograph tool scripted by Bernhard Schaffer / SuperSTEM UK Daresbury to create visual feedback while navigating the stage. It was initially written for the NION UltraSTEM system but could be used with any microscope/stage system capable of reading & setting stage coordinates by the scripting language of DigitalMicrograph. The script can be freely distributed and used, but the author would like to be notified if it is installed at a new location. Email: SuperSTEM@gmx.at

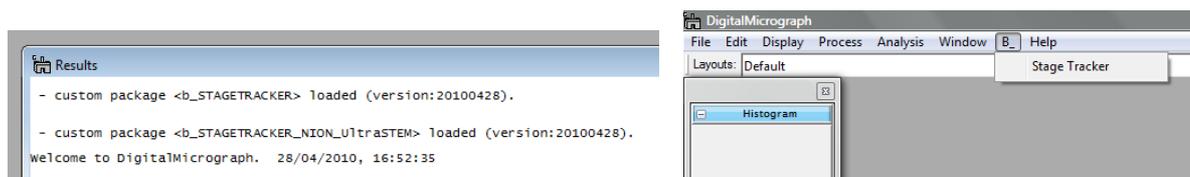
Installation

The package comes with three files, this PDF documentation and some plugin files called:

- **b_STAGETRACKER.gtk**
- **b_STAGETRACKER_NION_UltraSTEM.gtk**

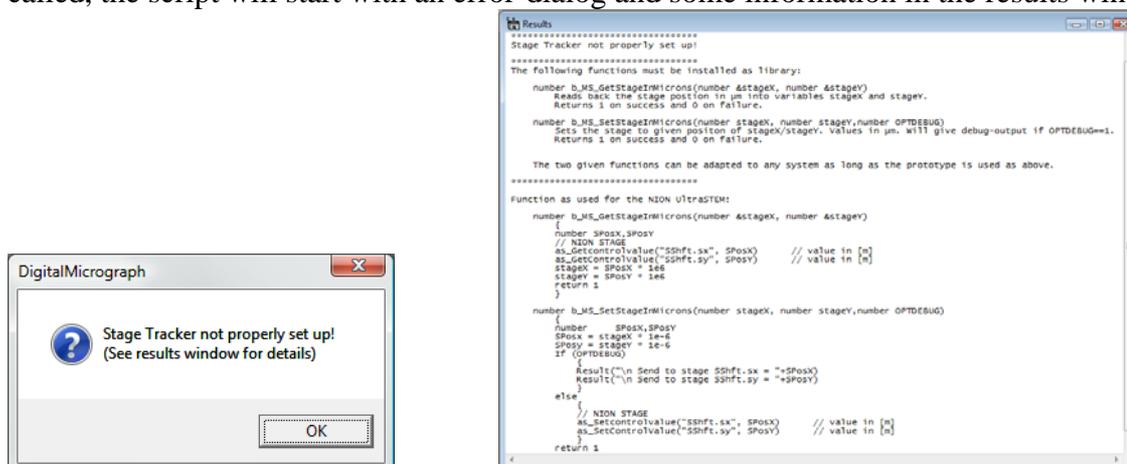
The tool is installed by moving all needed *.gtk files into the plugin folder of Digital Micrograph. The first file (b_STAGETRACKER.gtk) is the actual tool, the other files specify interface functions for different microscopes and one should choose the appropriate one. If the correct microscope is not supported, see section “*Installation for unsupported stages*”. Do not install more than one interface-function plugin!

At restarting DigitalMicrograph, the results window should list successfully loaded packages, and a menu-command could be found:



Installation for unsupported stages

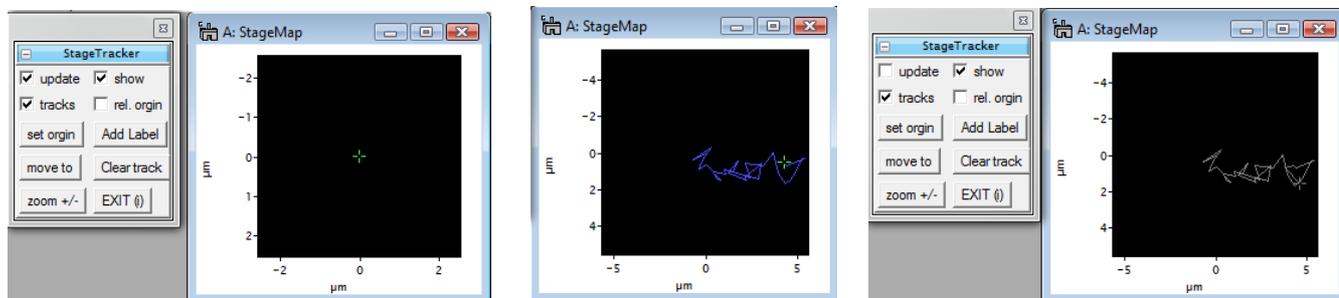
If your microscope/stage is not supported, DM-functions which communicate with the stage need to be customized/installed first. Copy the b_STAGETRACKER.gtk only. The first time, the menu command is called, the script will start with an error-dialog and some information in the results window:



You will have to create two wrapper-functions of the given prototype which read and set the stage X/Y coordinates in μm , respectively. Examples for the NION UltraSTEM are given. Install the two customized wrapper-functions as a DigitalMicrograph library and restart DigitalMicrograph.

Using StageTracker

After launching the menu command, a black image window and a floating palette is created. The image window is calibrated and shows a green cross at the current nominal stage position. After a defined period (0.5sec by default) the tool reads the actual stage-position and replaces the green cross accordingly. A blue track connects all stored positions with a continuous straight line. FOV and calibration update automatically as needed.



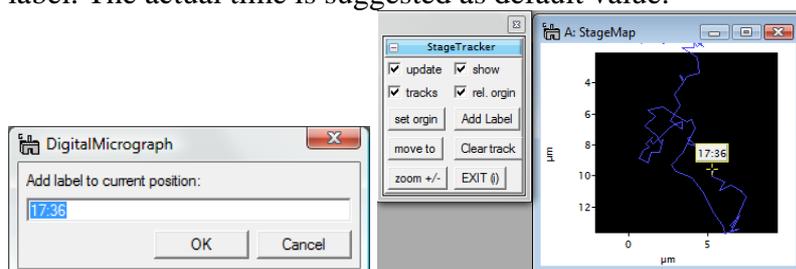
The image window should never be closed/deleted. All functionality of the tool is given in the floating palette and described below.

checkboxes:

- **update:** if disabled, both tracks and cross marker become gray and StageTracker no longer performs periodic updates.
- **show:** if disabled, the StageMap is no longer displayed, but it is still updated.
- **tracks:** if disabled, the blue tracks are not displayed, but they are still updated.
- **rel.origin:** if enabled, the StageMap is centred on a custom origin defined with the *set origin* button.

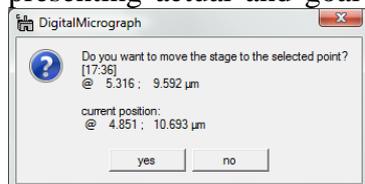
buttons:

- **set origin:** The current stage position is stored as custom origin for the StageMap if the *rel.origin* checkbox is enabled. This is useful for small-scale navigation once a region of interest has been found, because the map will always zoom in/out of the origin.
- **zoom +/-:** Changes the scale of the StageMap. Best used in combination with a custom origin. Keep the ALT key pressed when clicking the button to zoom out.
- **Add Label:** A labelled point-marker is inserted at the current stage position. A dialog will ask for a label. The actual time is suggested as default value.



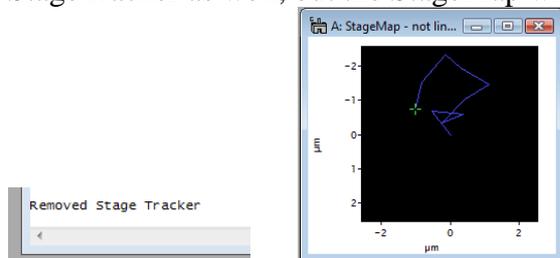
Note, that all point markers are standard point-ROIs which can be selected and deleted in the usual manner. However, they can not be moved.

- **move to:** Select a previously created label-marker before pressing this button. A dialog window presenting actual and goal coordinates will ask if the stage should be moved to these coordinates.

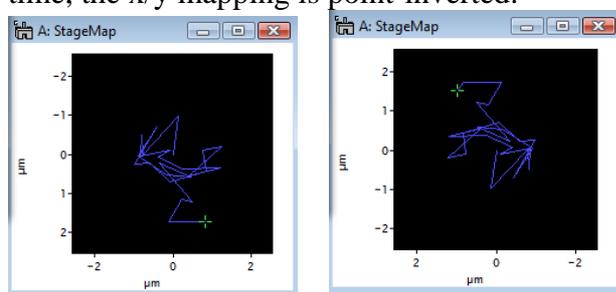


- **Clear track:** Deletes the history of stored positions (and thus the blue track). Note, that the list of stored positions is ever-growing and that redrawing the StageMap may become slow if too many points are stored.
- **Exit(i):** This button closes the StageTracker program. An according message is given in the Results window, the floating palette and the map are deleted. The memory is cleared.

Note, that closing the floating palette alone (by the little x sign in the top-right corner) will stop StageTracker as well, but the StageMap will then stay visible and in memory.



- **Exit(i):** There is a second function of this button. If clicked with the ALT key held down at the same time, the x/y mapping is point-inverted.



Some notes

- It is not possible to navigate the stage by clicking in the StageMap.
- The StageMap is calibrated. Thus, hovering the mouse over the map allows one to read the stage-coordinates underneath in the Image Status floating palette. These value could be used to navigate the stage to specific positions.
- Stage Tracker does not track tilt changes. “Stored positions” might thus not show the same sample position if the tilt has been changed in between.
- Starting StageTracker with the ALT key held down allows running the tool with non-default values for update-period. A shorter period will give more detailed tracks but many stored points slow down the redrawing process.
- The StageMap is a 512x512pixel image, but the stage positions are stored as ROIs which may lie outside the image borders. As a result “resizing” the image window might have unwanted display effects.
- The regular update of the StageMap does need a tiny bit of computational power. Thus, I recommend exiting the tool for final data acquisition.
- I do not take any responsibility for program instabilities, memory-leaks or other difficulties experienced during or due to the use of StageTracke. However, I have used the tool for 2 years without problems on the UltraSTEM Daresbury system.