An automatic tracking system for measuring activity of single neurons in freely-behaving animals

User Manual

Zaslaver Lab[1]

[1]Dept. of Genetics, The Alexander Silberman Inst. of Life Sciences
The Hebrew University of Jerusalem, Edmond J. Safra Campus, Jerusalem, Israel

The Required hardware and software

1) **Microscope system**: Olympus IX83. Required components:
    a) Lightpath filters: DIC, filter for a marker of interest (i.e. EGFP 525/50, mCherry 630/75)
    b) Objectives: x10, x4.
2) **Motorized stage**: Prior H117, driven by ProScan III controller. The system is operated via a joystick (Prior PS3J100).
3) **LEDS**: X-Cite XLED1 with a green module ($\lambda = 450 - 495_{nm}$)
4) **Camera**: PHOTOMETRICS Evolve 512 Delta
5) **PC software**:
    a) Matlab 2012b (The code doesn't support newer versions).
    b) Micro-Manager v1.4
    c) Drivers installed for all previously mentioned hardware.
    d) Windows 7 64bit.
6) **Optional** – Dual channel setup configuration:
    a) PHOTOMETRICS DV2 beam splitter. The dual view splitter creates two distinct frames, each in its wave length spectrum. DV unit is utilized only when acquiring the images in two channels, otherwise it doesn't affect the light path.
    b) X-Cite Red module ($\lambda = 540 - 600_{nm}$ with a narrowing filter of $\lambda = 555 - 590_{nm}$).

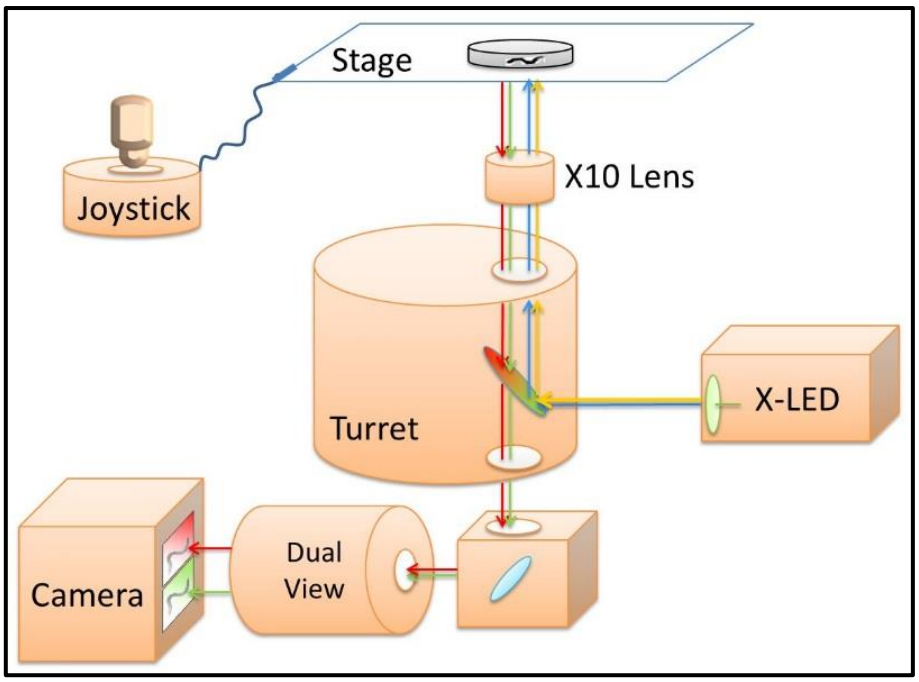


Figure 1: Scheme of the microscope system including the optional dual

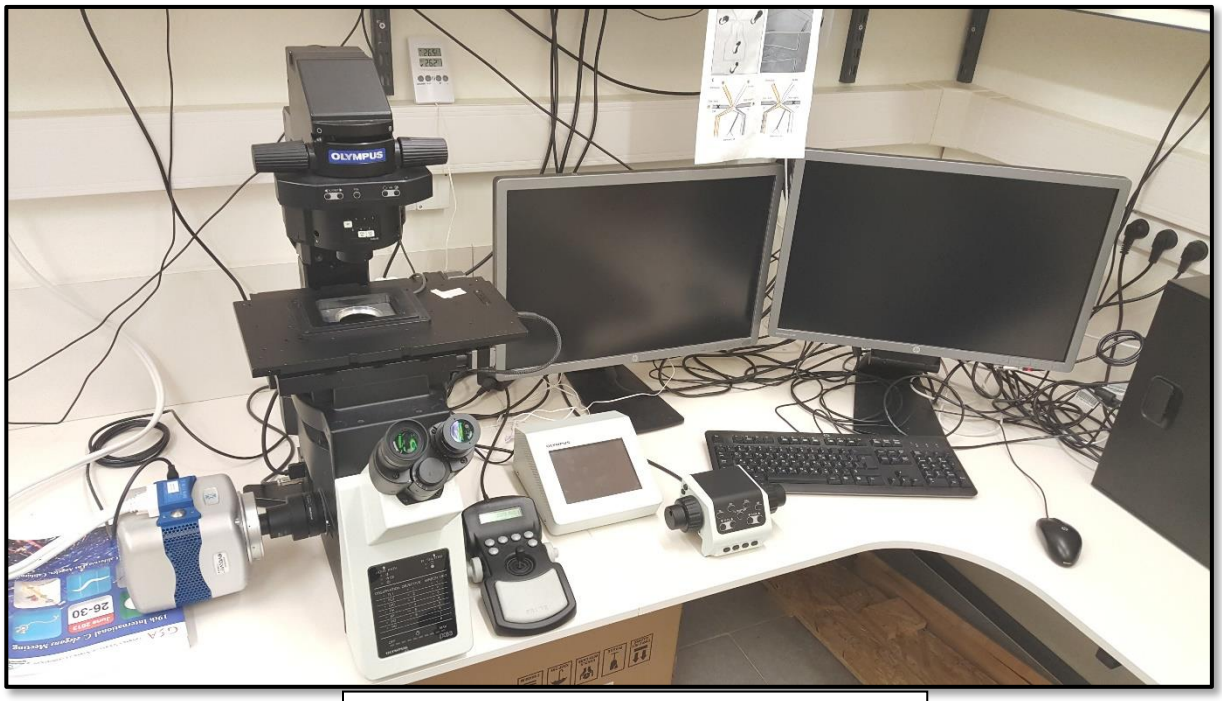


Figure 2: Microscope system without the beam

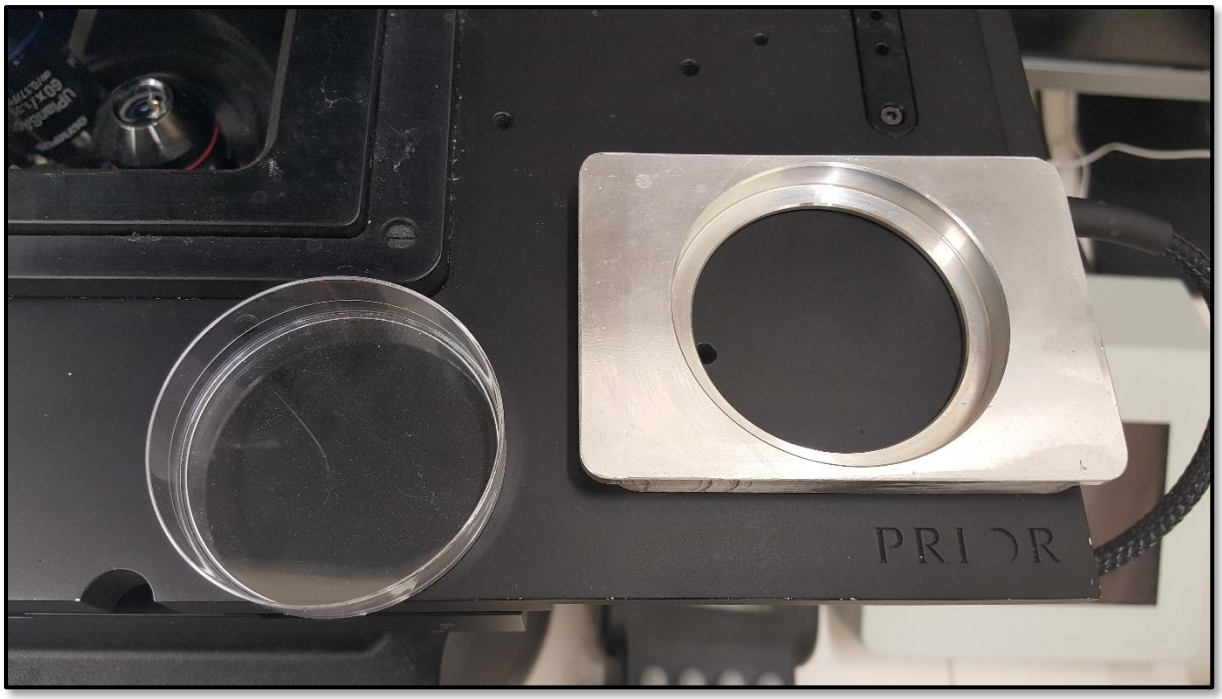


Figure 3: The petri plate mount. The square piece is fixated in the center of the

Installation of the software components

This sections' assumption is that you have only installed Matlab 2012b on your PC, and all the drivers for the hardware. The steps 1-6 are adopted from here.

1) Download and install MicroManager 1.4 (Use default settings)
2) Unzip the contents of the "Tracker_module.zip" into a preferred location (i.e. "C:\Tracker_package").

   Open Matlab 2012b, change current directory to the location from previous step:

   ```
   >>  cd C:\Tracker_package
   ```
3) Add the new folder to the working path definition of matlab by running:

   ```
   >>  addpath(genpath('C:\Tracker_package'))
   ```
4) Run:

   ```
   >>  MMsetup_javaclasspath('C:\Program Files\Micro-Manager-1.4')
   ```

   Change the directory according to the installation location of Micro manager.

   A file 'MMjavaclasspath.txt' is created in the working directory.
5) Run:

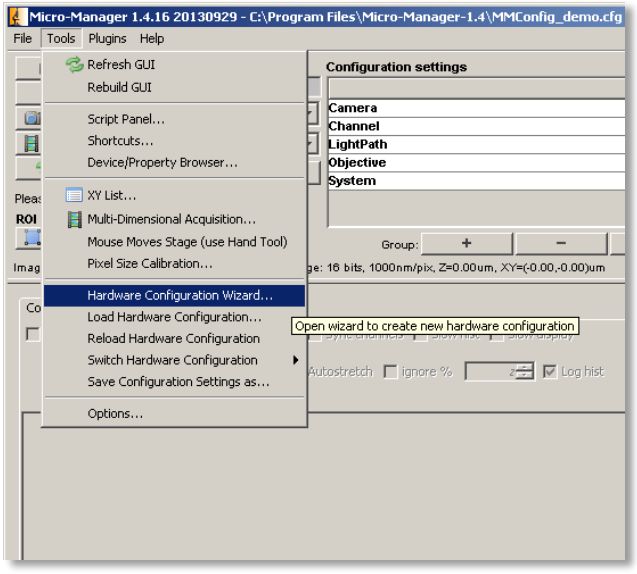
   ```
   >>  edit classpath.txt
   ```

   If the file doesn't exist, create one.

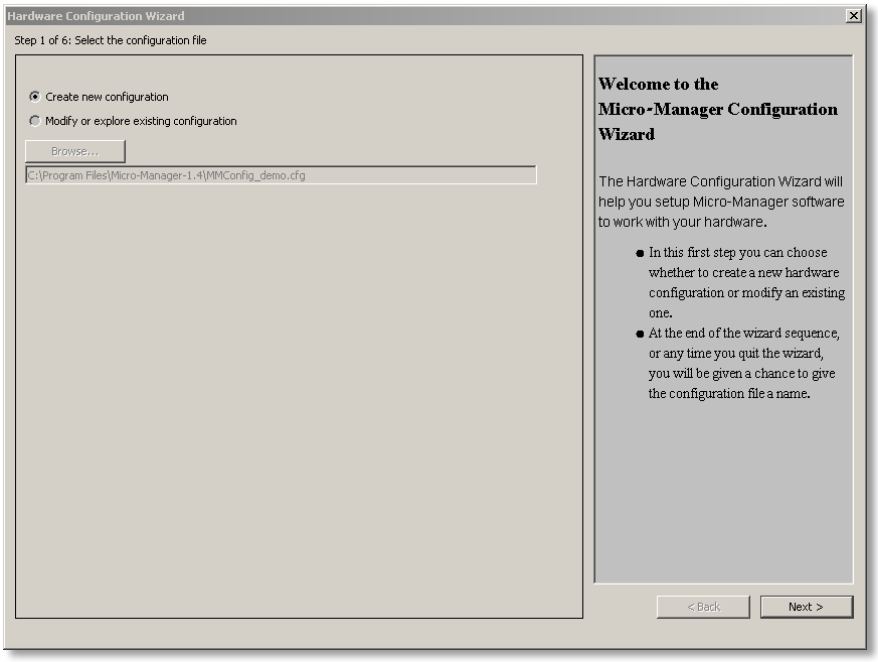
   Append contents of 'MMjavaclasspath.txt' to 'classpath.txt'
6) Add path of MicroManager to java variables by running:

   ```
   >>  edit 'librarypath.txt'
   ```

   Add the line `'C:\Program Files\Micro-Manager-1.4'` to the opened file, perform 'save'.
7) Restart matlab
8) Turn on the camera, Xled, stage controllers, microscope controller.
9) Open MicroManager, create a new configuration using press 'tools' -> 'Hardware configuration wizard'

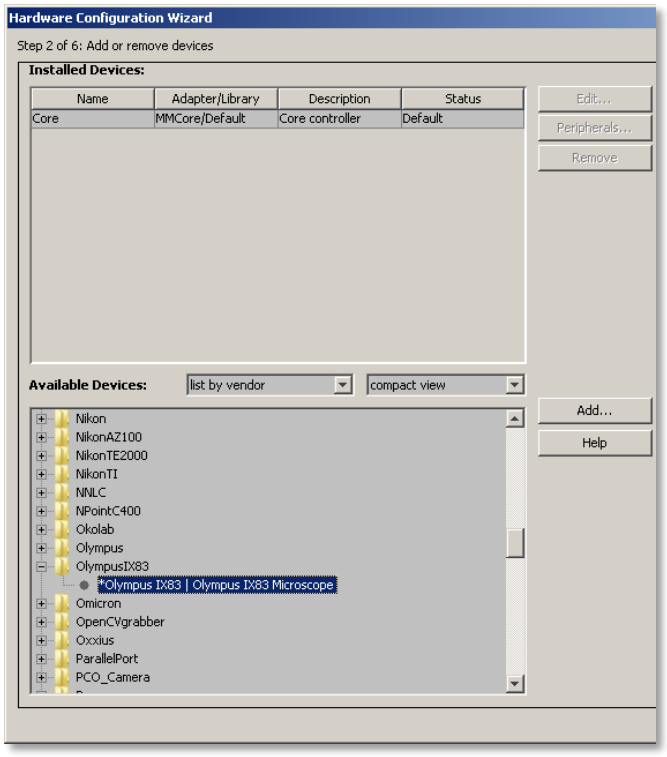


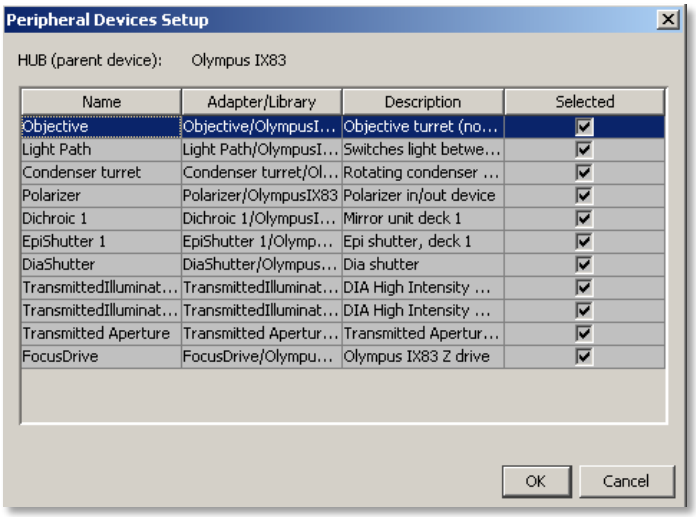
10) Pick “Create new configuration”, and hit next.



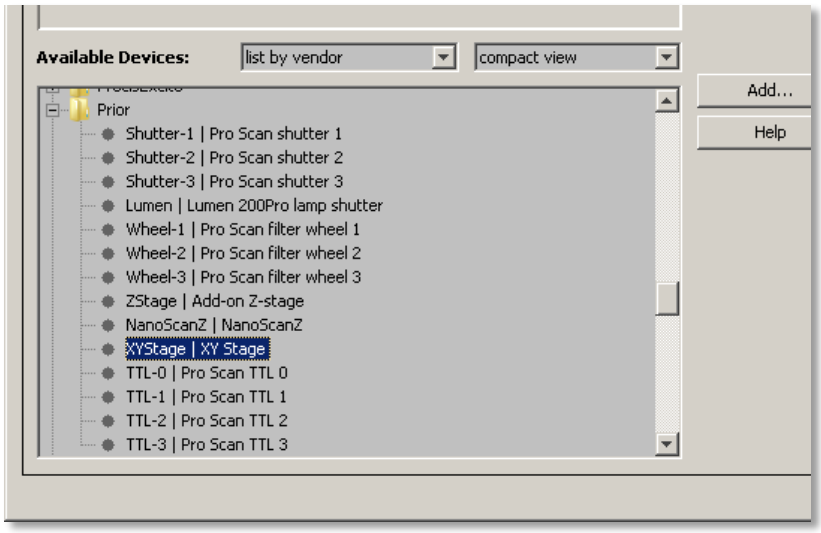
In the “Available devices list” find “Olympus IX83” folder, expand it, pick the “Olympus IX83” device and click “Add…”.

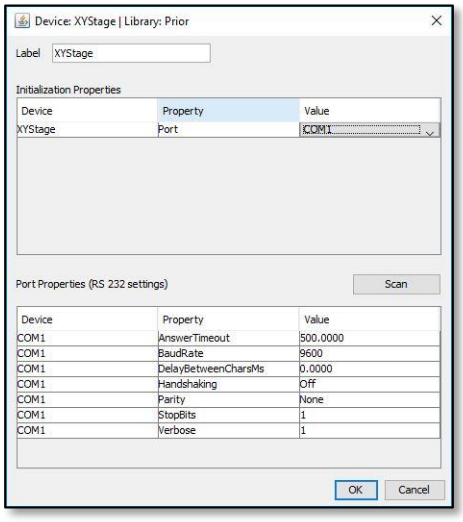
In the next window mark all available modules of the microscope ad click “OK”.





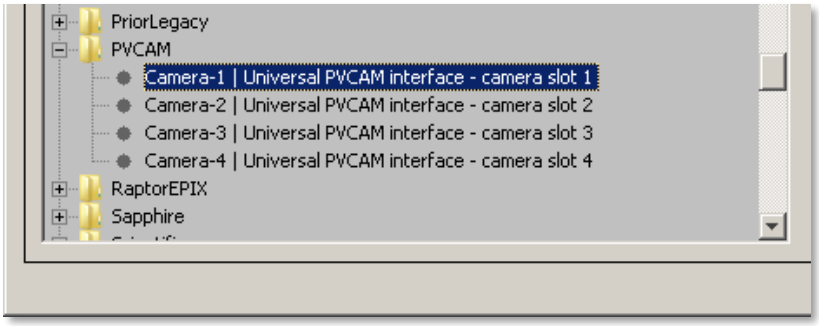
11) Find “Prior” folder and add XYStage device.



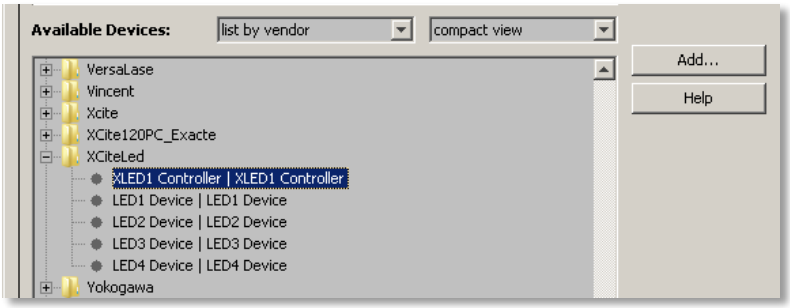


In the next window you will have to manually define the connection port (i.e. COM1 under the ‘value’ field, as shown above) through which the Prior Stage is connected to the PC. If you don’t know the exact port, jump to the section of this manual “Appendix: Identify COM ports”.
Pick “9600” for BoudRate.
Click “OK” to add the device.

12) Find “PVCAM” folder, expand it and add “Camera-1|Universal … “.



13) Find “XCiteLed” folder, expand it and add “XLED1 Controller|XLED1 Controller”.
In the next configuration pane pick the right port through which the XLED is connected.
Pick “19200” for BoudRate. Click “OK” to add the device.
In same device folder add the XLED modules you are interested to control (i.e. green LED module). They are marked as “LED1 device”. Device number is varying according to your hardware installation, i.e. in our setup the “LED1 Device” is the green module.



14) After all devices were added finish the configuration by pressing “Next”. Call your file “Tracker_config”. It is stored in the Micro-Manager install folder under the name “Tracker_config.cfg”
15) Close Micro-manager.
16) Open Matlab 2012b in administration mode (right click on matlab icon, ‘run as administrator’).
17) Check the connectivity with the configured hardware by running (First check that the installation directory of Micro manager and the name of the configuration file fits your setup):

```
>>  addpath(genpath('C:\Tracker_package'))
>>  checkCommunication('C:\Program Files\Micro-Manager-1.4\Tracker_config.cfg')
```

The script checks if matlab can successfully connect to the hardware using the MicroManager libraries, and perform basic functions. Check matlabs’ command line for the test results.

Adjusting moving directions and agar plate location on the stage

Three independent components can define the directions of movement when a joystick is activated: The camera orientation, the stage orientation or PC-driver configurations. The specific adjustment may be unique to each system, so here we present our setup, for which the provided software has been adjusted.

When joystick moved to the right: The stage moves to the left (Figure 4, yellow arrow), and the X values on the stage controller decreases. The image as seen in the tracking module is moving left (as if he observer is moved to the right relative to it).
When joystick moved up: The stage moves forward (from the operator, see green arrow), and the Y values on the stage controller decreases. The image as seen in the tracking module is moving down (as if he observer is moved down).
We flipped the function of the joystick via its controller, as shown to the here:

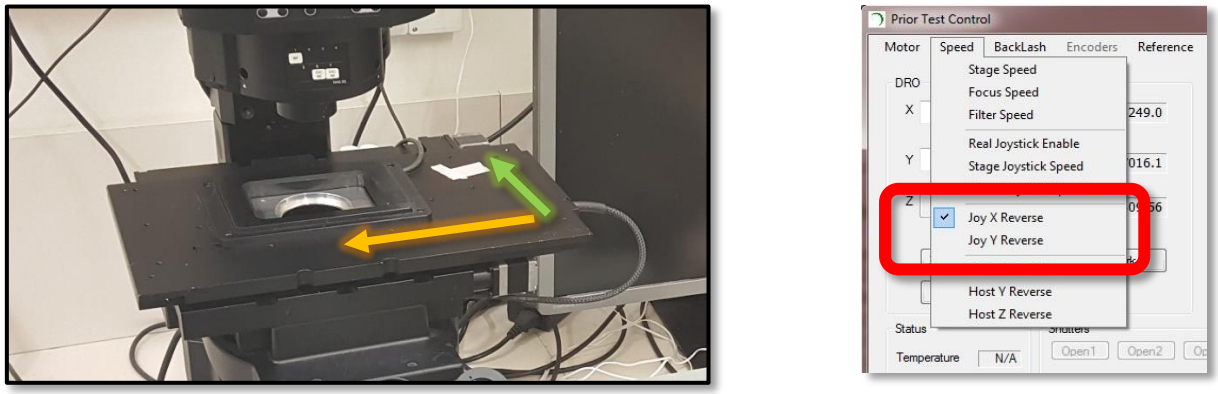


Figure 4: Directions of stage movement as a result of the joystick movement forward (green) or to the right (yellow)

In our setup the plate located at the center of the stage, using a custom made small petri holder.
The exact stage location of the [upper, lower, right, left] points of the petri dish should be specified in 'WormTracker.m', line 95.

Modules within the software package

The tracking software package consists of 3 main modules:

1. **Tracking module** – `Tracker_package\Modules\Track_module` - Module that automatically tracks a freely moving single worms. The output is a movie file (.tif format) in which each frame includes all necessary metadata of the recorded field of view.
2. **Analysis module** – `Tracker_package\Modules\Analyze_module` - This module performs analysis of the acquired experiments movie.
   The input is the previously created movie file, and the output is a compact .mat file that contains the results of the analysis i.e. fluorescence of the worms' neuron at each time point, its head location, direction etc.
3. **Visualization module** – `Tracker_package\Modules\Visualize_module` - This module provides tools for the experiment result visualization. The input to the module is the .mat file from previous step and (optionally) additional separate movie of the attractants' circumference. The output is a graphical interactive tool that simulates the crawling progress of the worm along the recorded period.
   Another visualization tool is a 'plot_results.m' script which draws the various data VS time, superimposed one against the other. Its other output is a final .mat file that contains all previous data with additional analysis results as distance to attractant at each time point, head direction relative to attractant etc.

1. The Tracking module

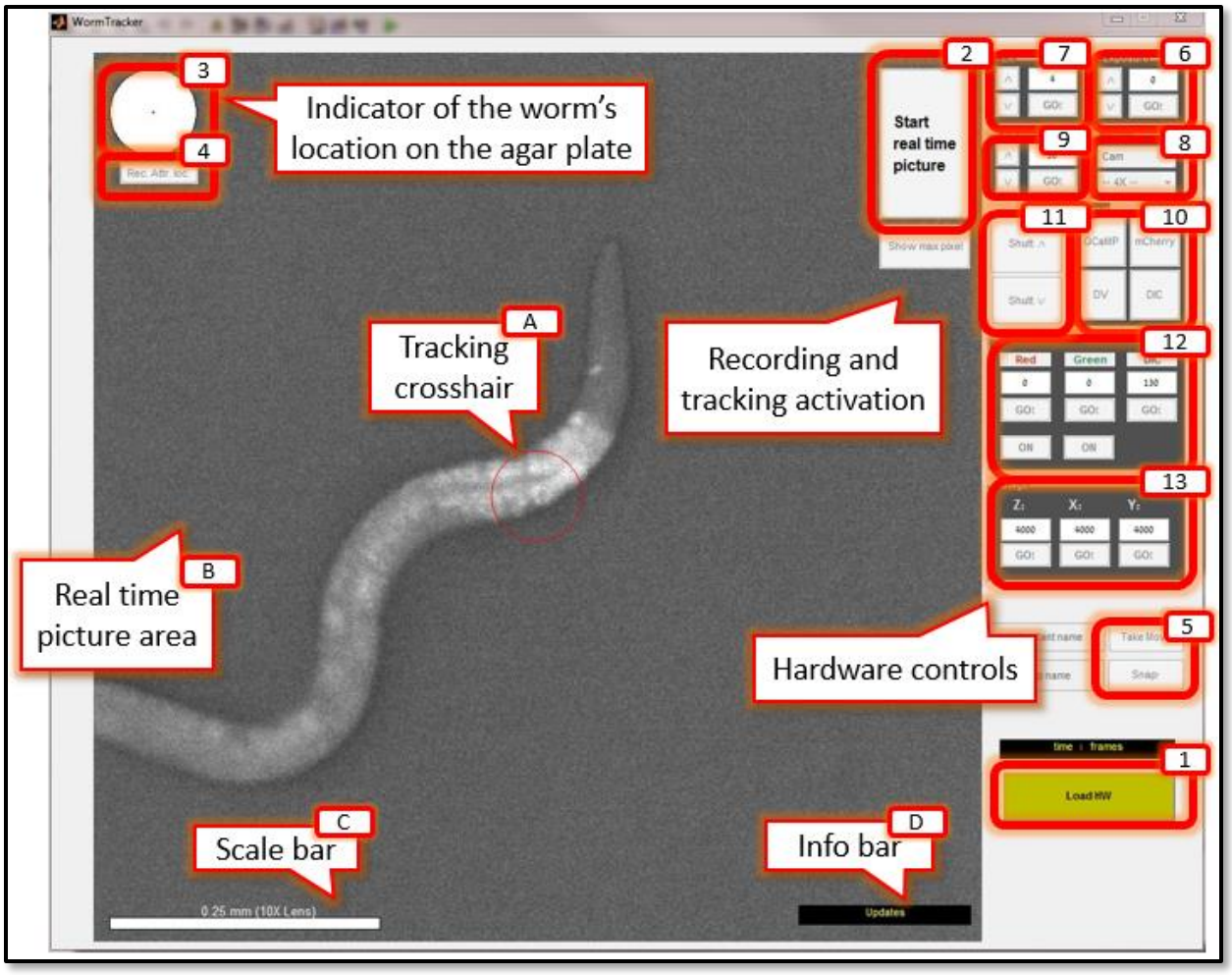


Figure 5: Tracking user graphical interface is the main part of the tracking module

This graphical user interface provides full control of the tracking and filming features of the module. Area (B) draws the last taken image, such that the image changes with the imaging rate (default of 11 frames/second), providing a “real-time” picture of the agar petri plate. Size of objects in the view field can be assessed using the scale bar (C). The tracking of a worm starts when the neuron is manually caught in the crosshair (A) using the joystick and the recording function if evoked (2.4 button, as shown below).
The current state of the system (Recording, tracking etc.) can be viewed in the information bar (D). A location viewer (3) provides a quick assessment of the worms’ location on the plate (as well as manually inserted locations of chemoattractants), depicted as red dot (and respectively green shapes, see figure 6).

1.1 The functions

More specifically, here presented all buttons and the functions they evoke:

No.	Function
1	Binds/releases all the needed hardware. Includes connectivity check. Upon success the info bar (D) will blink with green letters “HW loaded” (or “HW released”).
2	When loaded, the camera is initiated in resting state (a real-time picture is not shown). In order to see a real-time picture of the imaging area this function is to be activated. Upon activation the button divides into 4 different buttons as shown in the next figure, see figure 7.
3	Location window – shows the location of the worm in respect to the agar plate. It also shows added chemoattractants (or any other object), see figure 6.
4	Manual addition of objects to the location window, without creating a movie .tif file as done when activating button (5). As this function activated, the green shape on the window will copy the joystick movements until de-activation by pressing again on (4).
5	Creation of custom movies and snapshots. The attractant movie is recorded using the “Take movie” button: see step sub-section 1.2.6 in next page. As a result of these actions: (a) a shape of the drop will be drawn on the location window in green, see figure 6 (b) a separate .tif movie file will be created with the drops circumference, storing the exact X,Y locations. The name of the attractant movie can be changed by altering the text in the box to the left of the (6) buttons (rewriting the “Attractant name” string).
6	Camera Exposure controls. The current value is shown in the white window. The buttons “ ^ / v “ shift the value by a single step. Other option is changing the string to the requested value and hitting “GO!”.
7	Camera’s electron multiplication (EM) value. Same user interface as in Exposure (7).
8	Light path and magnification. Light path can be changed to “eyepieces” or “50/50”, in order to use the ocular lenses. Magnification can be reduced to X4 for easier navigation and localization of the worm on the plate. The filming is to be done in X10.
9	Brightening multiplication factor. The real-time image may be too dark to be efficiently visualized, so this function brightens the picture as it being presented to the user. It doesn’t affect the stored image in the created .tif movie.
10	Turret filter selector.

11	Open\close the epi and transmitter illumination shutters.
12	Controls of the LED modules and the bright-field light source. The “on” buttons activate\deactivate the leds. The intensities can be changed by typing-in the desired value and hitting “GO!”.
13	Indication of the X,Y,Z location of the stage.

Here shown the process of marking an attractant drop. Both buttons (4) and (5) will create the green shape on the preview window, while only (5) will create a new ‘.tif’ movie. present at the location window:

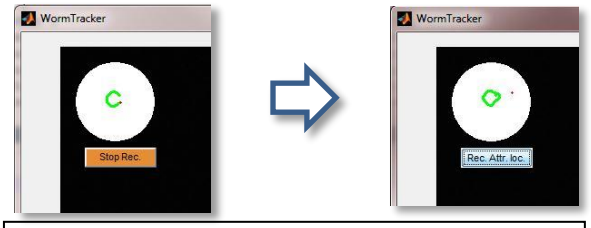


Figure 6: Recording and marking an attractant that was introduced onto the plate

After the real-time picture is started, different 4 buttons appear:

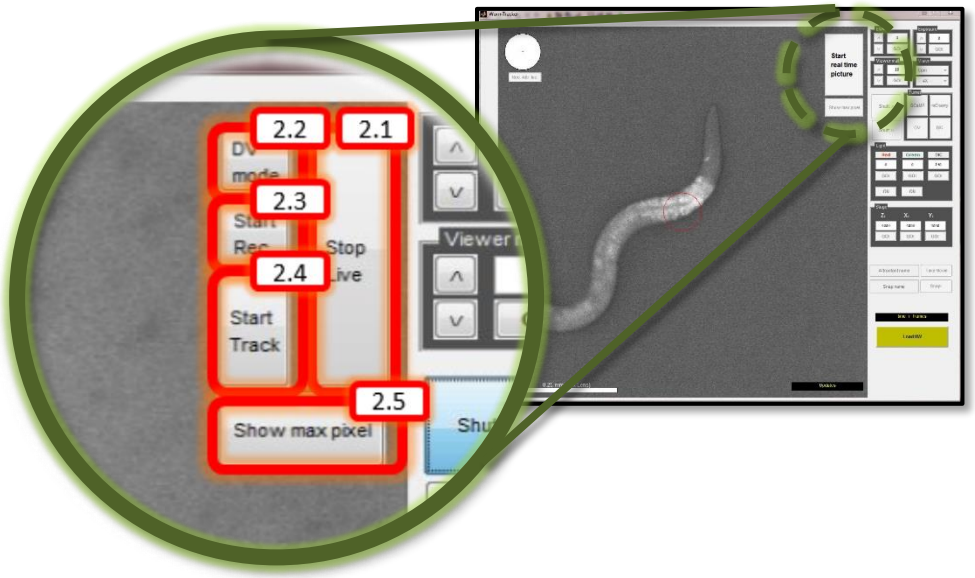


Figure 7: New features appear once the real-time picture is activated

No.	Function
2.1	Stops the real-time picture (puts camera to an idle state), and reactivates button (2)
2.2	Activates the dual view mode – should be used with the PHOTOMETRICS DV2 beam splitter. In this mode the tracking is performed using only the left half of the image (the red channel), such that the crosshair (A) moves to the left side.

2.3	Start/stop recording – starts to accumulate the acquired frames into a .tif file. The name of the file can be manually defined in the text field near buttons (6).
2.4	Start/stop tracking – starts the auto-tracking feature. First the crosshairs red circle has to be manually brought over the worms' neuron. When the auto-tracking is active, the system will keep the neuron in the crosshair despite the free moving of the worm.
2.5	Mark-up the brightest pixel(s) in the current frame. This function is useful when there is another bright spot near the neuron and one wishes to verify that the neuron is still the brightest object (crucial for efficient auto-tracking).

1.2 A typical workflow

1. After all hardware modules are powered on and matlab is initialized in "Admin" mode, the tracking module can be activated through:

   >> WormTracker

   Remember to set the path of matlab to the folder where the "WormTracker.m" is stored.
2. Press button "Load HW" (1). Info bar (D) should show green text "---- HW loaded----"
3. Place the plate with a worm on the stage and bring it under the objective (using joystick).
4. Press "Start real time capture" (2)
5. Adjust hardware parameters in order to get sharp picture of the agar surface.
6. Optional (but recommended) – if the experiment involves attractants that are dropped on the agar plate, do next steps in order to record the location of the drop on the plate:
   a. Take the agar plate of the microscope, and add a small drop onto the surface.
   b. Rapidly paint a small dot with a marker on the plates' other side under the drop location and fix the plate on the stage.
   c. Move the drop into the view field of the objective.
   d. Move the crosshairs to an arbitrary point on the circumference of the drop
   e. Change the text "Attractant movie" to the current chemical name, i.e. "DA_2_-3" for a $[DA] = 2 \cdot 10^{-3} M$
   f. Press "Take movie" (5)
   g. Using the joystick, move the stage along the circumference of the drop such that the drop border always stays in the center of the crosshairs (A)
   h. When the starting point was reached again, stop recording by "Take movie" button.

   # As a result a new .tif movie will appear in the working directory storing the [X, Y] location of the drops circumference. It will be later used for post-analysis. The circumference of the drop will be drawn in green on the location window (3), see figure 6. For convenience you can run

   >> getDataFromTif('Save')

   Which will extract the data into a separate .mat which is sufficient to all the following analysis steps. After that you may erase the original .tif file of the attractant.
7. Locate the worm and bring it under the objective (use X4). Change to X10 magnification (8).
8. Activate the blue LED, change turret cube, open\close the relevant shutters using the controls on the GUI.
9. Place the neuron of the worm in the crosshair (A) and press "Start track" (2.4).
10. If the tracking is successful press "Start rec." (2.3)
11. To stop the movie recording press again "stop rec." (2.3)

12. Stop tracking by “Stop track” (2.4)
13. Stop frame acquisition by “Stop live” (2)
14. Release all hardware by “Unload all HW” (1)
15. The newly created movie (.tif file) will appear in the working directory. The module automatically divides the experiment into different movies, each of 1000 frames (~1.5 minutes).

1.3 Example

In the folder 'Tracker_package\mini_example' there is a fragment of an experiment from out lab, "mini_movie.tif". The fragment is an output file of the acquisition module. A preview file is provided: "mini_movie.mp4".

2. The Analysis module

The analysis module requires minimal interaction from the user, and operates autonomously once the neuron is manually localized in the first frame of the movie.

2.1 A typical workflow

1. After a .tif movie is acquired using the tracking module, it can be analyzed by typing:
   >> NeuroTracker
2. A file-choosing window will open. Pick the .tif file that contains the movie.
3. A dialog box will pop up, with two integer input fields: First is the starting frame (automatically initialized to 1) from which the analysis should start.
   The second is the end frame of the analysis, automatically initialized to the last frame.
   If the entire movie is to be processed, just click “OK”.
4. The first frame will be opened in a viewer. By UP and DOWN arrows adjust the desired brightness of the image, and press SPACE to continue.
   The brightness only affects the visualization of the image in the viewer window, not the analysis itself.
5. Here the mouse arrow will be transformed into a crosshair. Double-click on the neuron center.
6. The analysis process will start, and a visualization pane will show the progress.

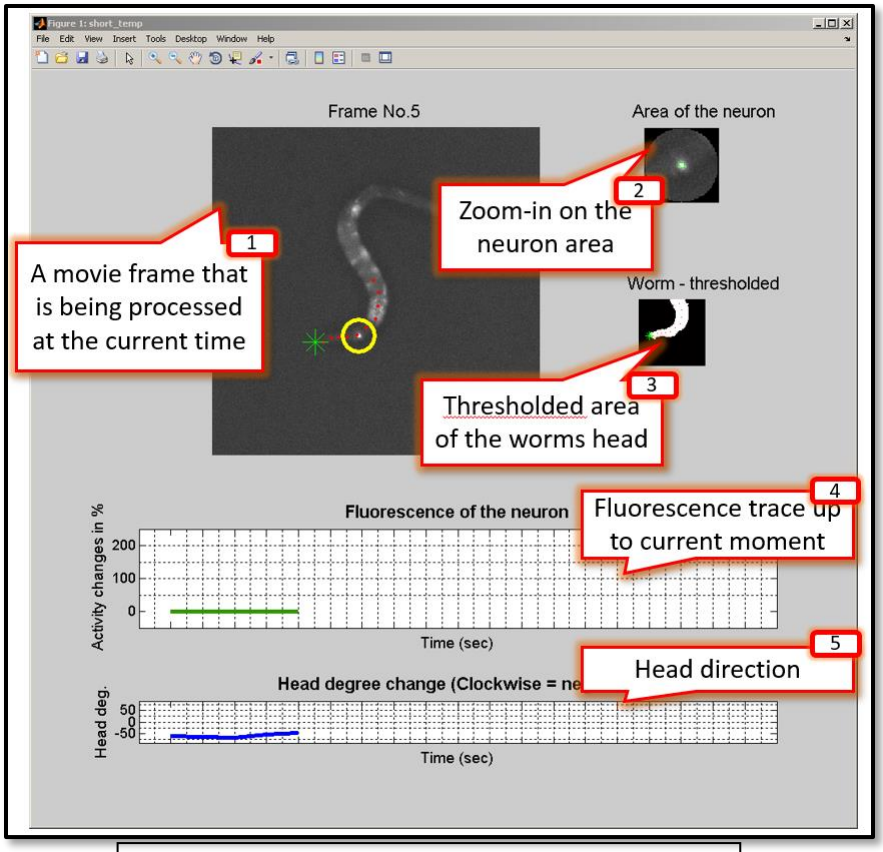


Figure 8: Preview window of the analysis module.

No.	Function
1	The current frame that is being processed. The neuron is shown by the yellow circle. The worms' head is marked by a green asterisk. The worms' skeleton is marked by red dots.
2	Zoom-in on the neuron area. The neuron pixels are surrounded by a green line.
3	The threshold image of the worms' head. The skeleton and the head point are painted on top.
4	Fluorescence trace of the neuron, percentage VS time. The plot is being filled as the analysis progresses.
5	Head angle relative to body direction, VS time.

7. When the analysis completes a new data-file (.mat) will be created in the working directory.

2.2 Example

1. Run the analysis module, and pick `'Tracker_package\mini_example\mini_movie.tif'`.
2. The process should look like the movie `'Tracker_package\mini_example\mini_analysis_process.avi'`.
3. The output should be identical to the provided `'Tracker_package\mini_example\mini_analysis_data.mat'`.

3. The Visualization module

This module is operated through a graphical interface as shown below. The crawling path of the worm is drawn in the main window (A), as the progress of the experiment is adjusted by the scroll bar at the bottom (E). The crawling track is colored according to the activation of the neuron at that location and moment of the experiment, where red color is a fully activated state of the neuron and blue is a background low activity.

The track plane can be explored or enlarged using the controls (F).

The info of the drawn results and progress in the experiments are shown in the info bars (B, D, C).

If the additional .tif movie of the attractants circumference is loaded, the module creates a new analysis file .mat with the calculations of relative distance to attractant at each time point and direction of the head relative to attractant. The attractant is drawn on the simulation plane such that the progress of the worm towards it can be easily visualized and assessed.

3.1 The functions

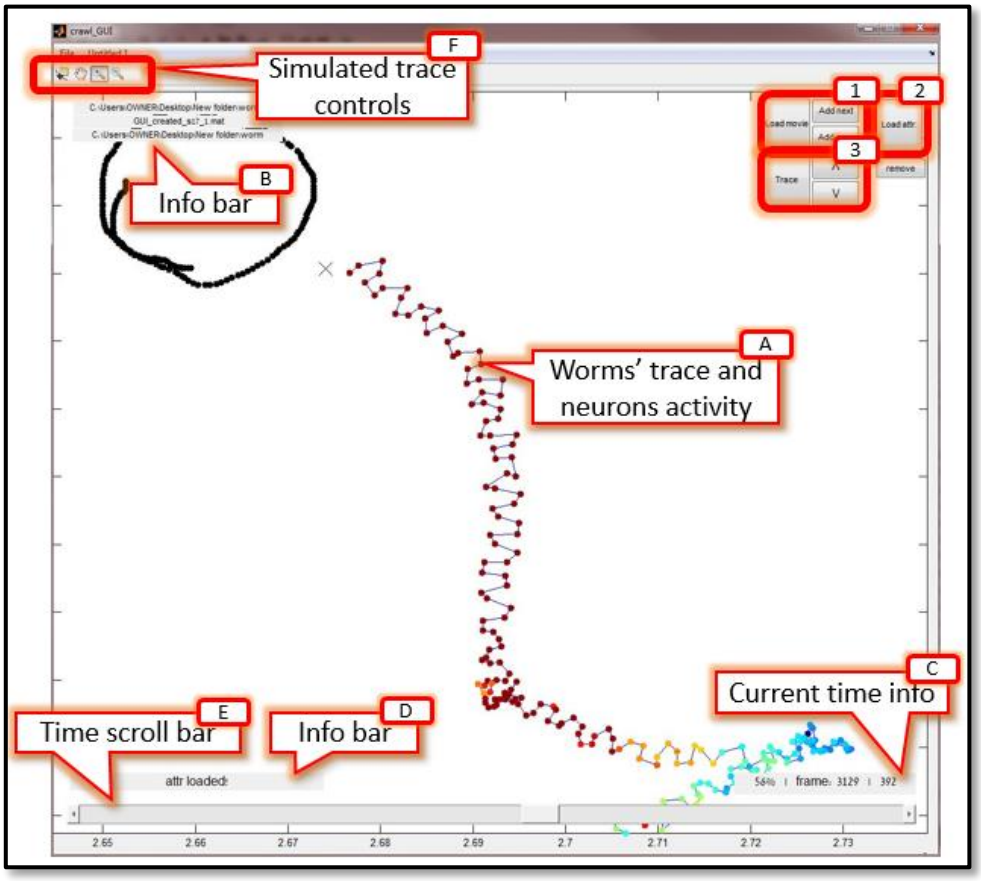


Figure 9: Preview window of the visualization module.

No.	Function
1	Add '.mat' data file (file that is created as an output of the analysis module) A typical experiment that spans over dozens of minutes may is stored in many consequent files, each of 1.5 minute. The analysis is performed on each part alone, so many .mat files are created for a single experiment. The consequent analyzed data files can be added using the "Add next" and "Add previous" buttons, or by picking multiple files after pressing the "Add movie".

2	Loads the attractant movie, that was created by the recording module (See 1.2.6 section)
3	"Trace" button shifts between the full crawl path representation and the "short memory" representation of only last n frames. n can be adjusted using the "^/v" arrows. The full crawl path prints location evolution of the worm from the start of the experiment up to the current moment. "Short memory" only shows the trace of the crawl in the last few seconds.

3.2 A typical workflow

1. After the .tif movie is analyzed and the .mat file is created in the analysis step, the visualization module is started by:

   >> crawl_GUI
2. Press 'Load Movie' button (1). In the browser window pick the .mat file (file that was created by the analysis module).
   If there are few consequent movies of the same experiment, pick them all in the chronological order.
3. Press 'Load attr' if there is an attractant .tif movie (result of stage 1.2.6). Pick the .tif or .mat file that contains the attractant drop circumference.
4. Scroll the time scroll bar (E) to view the worms crawl progress.
5. Use the controls (F) to see worms' location relative to attractant or zoom in on a specific region.
6. Press 'Trace' (3) to reduce the showed trace size. Adjust the trace length using the arrows.

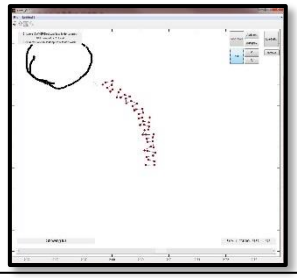


Figure 10: Adjusting the length of the trace length

3.3 A qualitative visualization of the results

An alternative way to superimpose the behavior and the neuronal activity is by picturing all the data in a single multi-plot figure. It is done by running the next script:

>> plot_results

The script will prompt for 2 files: An output of an analysis module 'NeuroTracker.m' and the respective data of an attractant movie (.mat or .tif file)

The opened plots will look similarly figure 11. Every plot is explained in the next table:

No.	Function
1	Fluorescence of the neuron in %. Red asterisk sequences at the bottom of the plot indicate times when the worm moved backwards. Black asterisk sequences at the upper part indicate times when the focus was changed.
2	Speed of the worm (scalar). Backward crawling is indicated as a negative speed.
3	Binary direction indicating reverse crawling events: $1 = back$
4	Distance to the closest point on the attractant circumference.
5	Direction of the head relative to the attractant drop center
6	Angle of the head relative to the previous body segment.
7	Z axis of the stage. Changes represent refocusing events

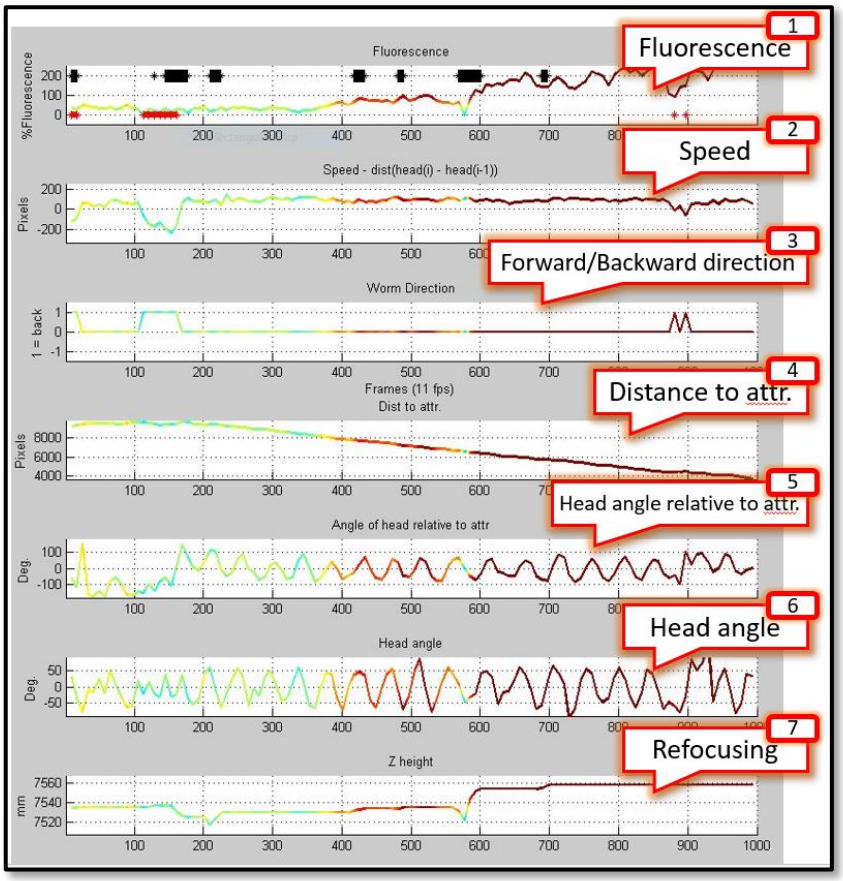


Figure 11: Plots drawn by the visualization module

3.4 Usage examples

1. Run 'crawl_GUI'
2. Press 'Add movie', pick 'raw_data.mat' from 'Tracker_package\long_example' folder.
3. Press 'Add attr.', pick 'Attr_movie_XY_data.mat' from same folder.
4. Run ' plot_results' and pick same files in same order

Appendix A: Find out COM ports for the connected hardware

1. Open “Control panel” -> “Hardware and sounds” -> “Device manager”
2. In device manager open Ports (COM & LPT) list
3. The connected device names are listed together with the correlating port:
    a. XCite lumen device is connected as COM7
    b. Prior Stage as connected to COM1 and is named as 'Communications Port'.