**pySolo-Video:**

1. Launch pySolo-Video from the shortcut on the desktop.

* In Option > Configure, define the output data folder location. You can change different settings like FPS, how many monitors/videos you want to be displayed, size of these thumbnails, etc. Click Save when finished. I have set this up already on the Chrimson PC, but you will have to change the FPS according to the video you are using.
* Since I have done all the following steps on the Chrimson PC, whenever you open pySolo-video or pySolo-Acquire (see below what pySolo-Acquire is), the file that was used in the previous session will open automatically. You will have to the save your configurations as a new file. It sometimes doesn’t work in one go since the previous files and settings don’t get removed. You will have to play around with it until it works.
* In thumbnail view,
  + Select a monitor
  + Use Select Video Input > File > Browse to find the movie that will play on that monitor
  + Apply
  + Play
    - Don’t worry if it plays in the wrong window.
* Select Monitor number from the dropdown menu. Then Select Video Input > File > choose the video file you want to analyze. Click Apply > click Play. This will associate the chosen video file to the chosen monitor number. The video should play in the monitor next to the chosen monitor number (there is a bug in the code). Sometimes the video will play in the same monitor that you chose instead of in the next monitor. I can’t predict which one out of the two it will be. Wherever it plays, you will see it.
* Once all movies have been applied to monitors, close PySolo-Video.
* Restart PySolo-Video
* Go immediately to the live view tab.
  + Any activity on the thumbnail view at this point will confuse the program as far as which file goes with which monitor.
* Use the dropdown menu to select the monitor to be viewed.
* The movie name will appear below the dropdown menu
* The movie will play.
* Other movies that were assigned when the program was shut down can be viewed as well, just don’t do anything in thumbnail view until you are done using live view.
* On top right of pySolo-Video window there are two buttons: Thumbnails and Live View. We are in the Thumbnails panel by default when pySolo-Video is launched. Go to Live View panel by clicking on the Live View button. This is where the mask with ROIs for fly tracking is created.
* Select monitor from the dropdown menu. Your video will play.
* Define ROIs for each well with a fly in it. To define an ROI and start tracking a fly, left click and drag around the well > a red rectangle will be drawn for your selection > inside this rectangle, single click the middle mouse button > fly tracking in this ROI will start when the fly moves. Each fly will have a small red rectangle drawn on it and white + sign and you will be able to see it moving as the fly moves.
* You may de-select the most recently drawn ROI by right clicking anywhere. You may de-select all the ROI by clicking the ‘Clear All’ button in Mask Editing section displayed below the monitor number.
* The order in which you draw the ROIs is the order in which the data will be written in the final file. The ROIs are analogous to channel numbers in DAM.
* Once all the ROIs for all the wells with flies are drawn, click Save Mask or Save and Apply button under Mask File section. You will be prompted to save the file where you can change the name of the mask file (.msk file). I think both the buttons are doing the same thing but to be sure, if you clicked the Save Mask button, click the Load Mask button afterwards and chose the same mask file that you just created and saved. You can apply this mask to any other monitor, even for a different video file, by using the Load Mask button, as long as the 96-well plates are more or less in the same place in all the videos.
* Go back to the Thumbnails panels (button on top right corner) > under Set Tracking Parameters choose the mask file that you just created in the Mask File section.
* Under Calculate Fly Activity as…, choose Activity as Distance Traveled.
* Click and check Activate Tracking > make sure the initial video file and the associated monitor number are correct > if everything looks right, click Apply in the Select Monitor section.
* Click File > Save as and save this entire configuration under a recognizable name (.cfg file). Click File > Exit program to close pySolo-Video.

**pySolo-Acquire:**

* Now each input video is associated with a mask file that is used for tracking the flies. To acquire tracking data from the file that you just created, launch pySolo-Acquire. Yes, it is a separate GUI. I think I made a shortcut on the Desktop but if not, search for it in the Start menu.
* Choose the file that you just created, Pick File > Browse. Once the file is selected, you will see the monitor number, source video, associated mask file, output file name, and tracking type for each monitor (Track type 0 = Activity as Distance Traveled).
* Check the monitor you want to acquire data from > click Start button at bottom right of the window. A .txt monitor file will be created and pySolo-Acquire will start writing movement data into it. It doesn’t stop acquiring data, and writing zeros when the source video is over, until you hit the Stop button. I don’t know how long it takes for how much data to be acquired. I just let it run for couple hours to an evening if the video is long.

The .txt monitor files created always start with the same date and time stamp (Dec 31, 1969, 19:00:01). I don’t know why!

**Excel manipulation:**

This is what Rosbash’s postdoc sent me:

* Chunk .txt files into 32 channels
  1. Open Excel, in “Data” tab, go to Get External Data 🡪 “From text” button. Select .txt file
  2. Steps 1 and 2, click “next.” In step 3, data format: text and highlight the two columns with the date and time. “Finish”
  3. Column J contains “?”. Replace all “?” to “0” by copy/paste from column I.
  4. Columns A- J is basic info. Columns K-AP are the first 32 fly activity data. AQ-AU are the remaining five flies.
  5. Each text file needs to have 32 data points. If more than 32, copy columns A-J (basic info) & transfer excess data columns into a new Excel sheet tab. If less than 32, complete sheet with “zero” data points.
* Save each Excel sheet as individual .txt files.
* Convert the modified .txt file by using normal **DAMFileScanner** software.

DAMFileScanner doesn’t detect any data in the .txt file created by pySolo from our videos. Even pySolo main, a 3rd GUI which is used to analyze DAM data, can’t detect any data in the .txt file. This is where we are stuck right now.

If this would work, Rosbash’s lab uses **MATLAB** to analyze sleep from the binned data from DAMFileScanner.