

motr manual

Getting started

Which motr should I use?

motr comes in two flavors, a Windows version and a Linux version; both need MATLAB (version 2009b or later and the Image processing, Signal processing and Statistics toolboxes; the Linux version also needs the MATLAB compiler).

Both versions can be run on your local computer; the Linux version can also run on a cluster (this splits tracking up into multiple separate jobs and makes it faster).

Downloading and installing motr

Simple install (zipped files):

- a) Go to <https://github.com/motr/motr> and click on the button that says Download ZIP (on the righthand side of the page).
- b) Extract the zipped files to a directory.
- c) Start MATLAB
- d) Add the directory with the motr files to your MATLAB path

Slightly more complicated install (clone the git repository):

- a) If you don't already have a copy, download and install a copy of git (I like tortoisegit: <http://code.google.com/p/tortoisegit/>).
- b) Make a directory for the motr files, right click and select "Git clone". Enter the URL of the motr repository: <https://github.com/motr/motr> and make sure the Directory name is where you want the motr files to go. Click "OK".
- c) Add the motr directory to your MATLAB path

note: Running motr on a cluster

The details of this are going to vary depending on where you are. You'll need to find someone computer savvy (if you aren't yourself) at your local institution to help with the details. However, if you are running Windows, a good place to start would be to download an NX client for Windows (see Appendices for detail).

Recording Video

General recording notes

When it comes to tracking mice, a lot depends on the quality of the input video. It is worth your time to try to make your videos easy to track.

Easy to track means:

- high contrast between the mice and the background
- no features in the environment that occlude the mice
- no large features in the environment the same color as the mice
- uniform lighting over the arena and over time
- patterns on the mice that are different from each other (see figures A and B below).

Picking a camera and recording method

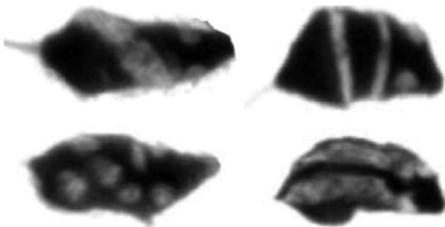
Currently motr can track .avi, .mj2 and .seq files. We use a Basler A622f camera, but any camera will probably work as long as the video format and resolution are right.

Resolution: We have mostly worked with 768x768 pixels over 0.6x0.6 meters. Higher resolution would likely be fine, lower resolution might well work too. Let us know! motr assumes a mono image.

Frame rate: 30 Hz works well for mice (no major teleportation events), but you can get reasonable tracking at 10 Hz.

Preparing the mice

1. Picking patterns



We've used variations on the patterns on the left, but there is nothing magic about them. If you are using motr to track only a single mouse, you don't need a pattern at all. If you are tracking multiple mice, though, pick patterns that are easy to apply. The spots are pretty easy to put on with a Q-tip, freehand, but the stripes are much crisper if you mask a pattern onto the fur with tape (lab tape works well). Caveat: Be careful that the marked patterns don't break the mice up too much on the video (don't use thick lines) or

they will be hard to track.

2. Marking black or agouti mice

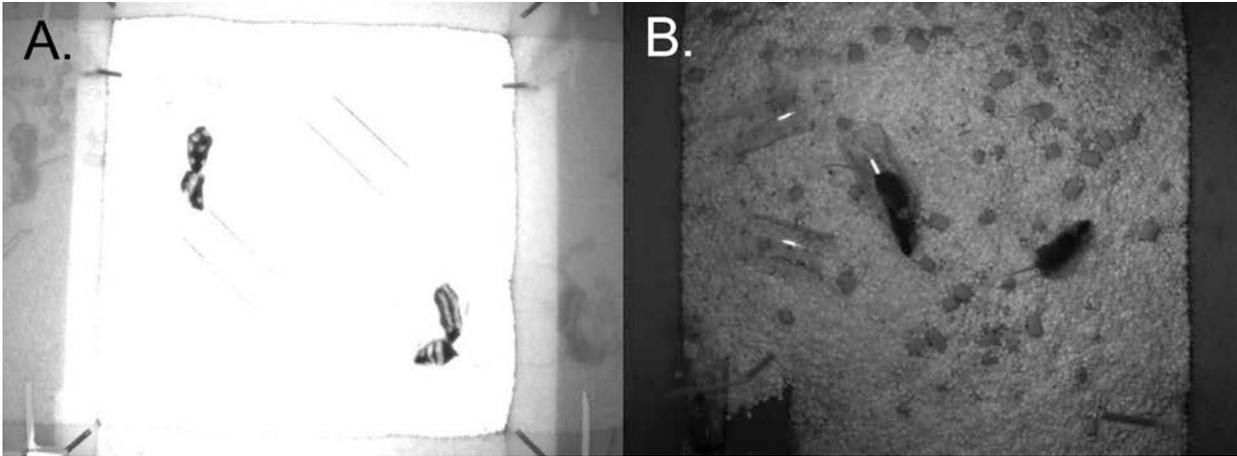
Human hair bleach is safe to use on mice. Anesthetize the mice, mix up the bleach, apply the patterns, follow manufacturers instructions for time. Make sure the mice stay anesthetized for the whole time, or the patterns will smear. Be sure to rinse the bleach completely to avoid any skin irritation.

3. Marking white mice

All the commercially available black hair dyes we've tested are not black in the infrared, so if you are using IR light, check to make sure your marks will show in the video. If you find a black hair dye that is dark in the IR, let us know! A black sharpie is a convenient way of marking white mice that actually does show up in IR. It doesn't last as long as dye, though. Black tattoo ink is also IR dark, as is squid ink.

Preparing the recording arena

Adjust your lighting so that there are no large changes in light intensity across the arena. Adjust the gain on your camera until you get the highest contrast you can between the mice and the background.



You want your recording to look A, not like B.

A: high contrast between mice and background, high contrast marks on the mice, low change in light intensity across the arena, good contrast between the mice and the walls near the floor, no mouse-colored objects in the arena that aren't mice.

B: low contrast between the mice and the background, low contrast marks on the mice, change in light intensity across the arena, low contrast between the mice and the walls, and dark corner that is close to the same color as the mice.

Note: IR lights are useful if you want to film in the dark (recommended, since mice are nocturnal). An IR pass filter is useful if you want to film across dark/light cycles (need to avoid strong lighting variation across the recording).

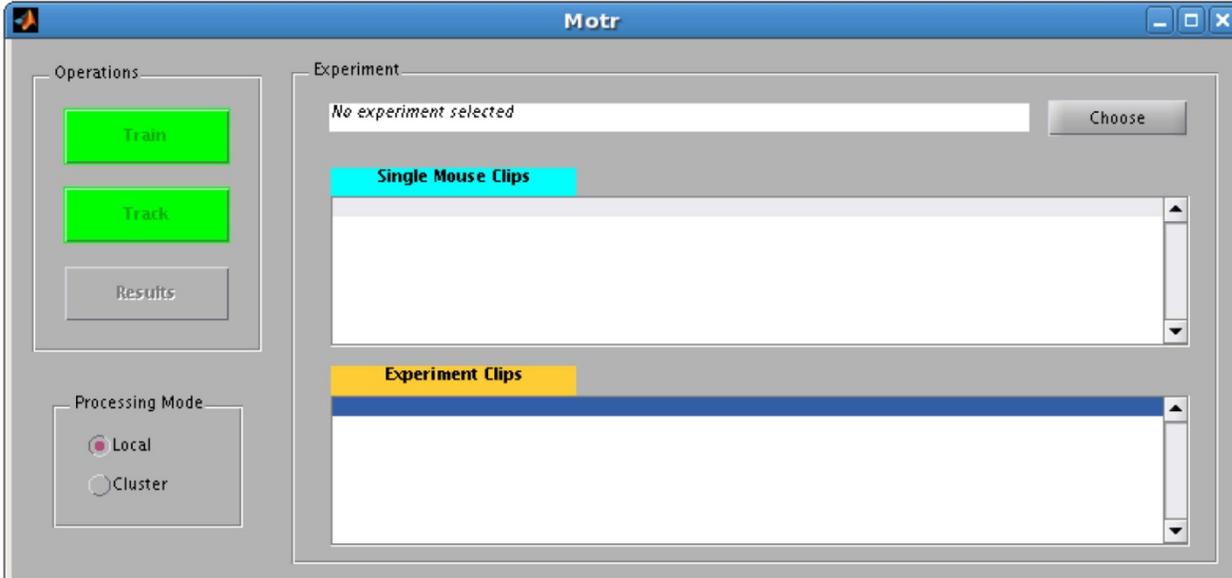
Recording multi-mouse video

Record your video. Make sure your frame rate and contrast stay constant.

Recording single mouse video for identity templates

After you have recorded multi-mouse data, you need to record some video of each mouse in the recording arena alone. Ten minutes is usually enough, but make sure that the mice move around during that time and explore the entire arena, to make sure you get representative samples of poses. This data will be used to train the identity classifier. If you are only recording a single mouse, you can skip this step (as your main video is also a single mouse video).

Running motr



There are three buttons in the motr GUI: Train, Track, and Results

Train: trains the segmentation and identification classifiers

Track: tracks the mice using the classifiers

Results: lets you look at what you just tracked

If the buttons are red, this means that the operation has already been performed for the current data set. If the buttons are yellow, it means they are in the middle of doing it. If the buttons are green it means you can start analysis.

In the Linux version you can toggle between two processing modes: local and cluster. Local will just run on your local computer, cluster can take advantage of the increased computing power of a computer cluster by splitting the tracking into jobs. The Windows version will currently only run locally.

Starting motr

a) Start Matlab

b) cd to the directory with the motr files in it

c) Enter motr at the Matlab command line and hit return

(If you want to run on the cluster, click the “cluster” option in the Processing Mode box (lower lefthand corner) of the motr GUI).

Start an analysis session.

Right click on the arrow to the right of the experiment directory box and select “Select a new experiment”. This will pop up a box to navigate to the directory you want to keep the analysis files in.

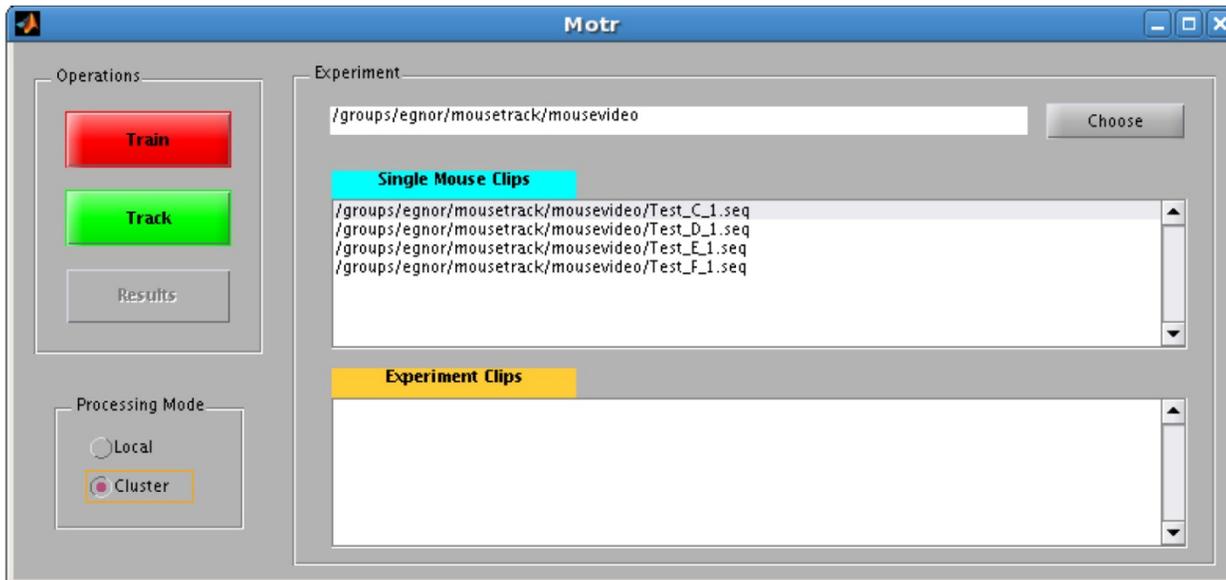
Training the identity template

Now click the Train button

This will pop up a box to request the single mouse videos. The title bar of the window that pops up says: select all single-mouse clips - one clip for each mouse. You select with the mouse while holding

the Ctrl button down, to select all the single mouse clips. Once you have selected all the single mouse clips, click OK. A message box will pop up that asks “would you like to train now?” (Y/N)

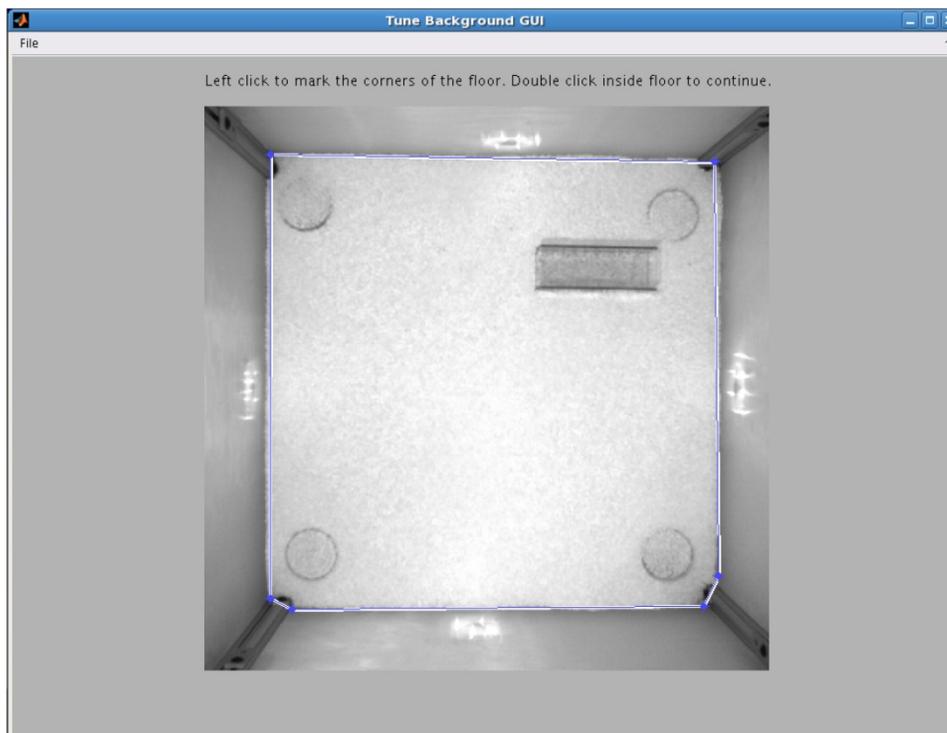
If you say Y, the Train button will turn yellow, and the cursor will have a little rotating icon, indicating its working on training. When it’s done, the Train button will turn red.



Track the mice

Click the Track button.

This will prompt for the experimental video file(s) to track.



Once you have selected the files to track, the Tune Background GUI will pop up with the instruction “Left click to mark the corners of the floor. Double click inside the floor to continue”. Left clicking will

start the creation of a blue line to outline the floor of the cage. This is selecting what motr will consider the background, for segmenting mice from the background. When you are done, double click inside the floor.



Once tuning is done, there will be colored ellipses in the TuneBackgroundGUI, one for each mouse. Using the arrow-slider at the bottom, click through each frame to make sure each mouse has one and only one ellipse on it. If you find errors you can correct them. Move the whole ellipse by click and dragging the center dot, stretch by click and dragging the peripheral dots. If the frame has no ellipses and says “invalid frame”, don’t freak out, just skip it. Don’t worry about matching ellipse color to mouse, this is random across frames. When all the frames are correct, click the button at the bottom that says “Finished Marking — Perform Optimization”.

note: what is happening here is that motr is deciding which are the mouse pixels (“foreground segmentation”). Inside the ellipse is , outside ellipse is . Head– tail direction and identity don’t matter for this step.

The button on the TuneBackgroundGUI will now say “Finished Segmentation Optimization — click to close window”. So, click to close the window. If you don’t you’ll just be sitting there forever.

A message box will pop up that says “Do you want to run tracking now? (Y/N)”. Click Y. Tracking will now commence. When it is complete the Track button will turn red, and you can look at the results. This will take a while, so now is a good time to get up from your desk and stretch.

Track more

To add experiment clips, you can click the Track button again, even though it is red. It will pop up a message box that says “Do you want to select new experiment clips?” Options are: [Yes],[Yes, but keep the old ones],[No, keep these]

If you select Yes, you can select new clips. The old clips will be removed from the list.

If you select [Yes, but keep the old ones], you can add new clips.

If you select [No, keep these], you can retrack the current experiment clips.

Note: If you have already tracked this video (the Results files exist) and nothing has changed in the classifiers motr warn you that the track files have already been generated, and ask you if you are sure you want to re-track. You should not re-track if the classifiers haven't changed, because the resulting tracks will be identical--i.e. it's a waste of time and compute power.

Once you have selected more clips a new message box will pop up that says: "Detection is already tuned. Do you want to retune it?" Options are: [Yes, but start from the previous tuning],[Yes, ignore the previous tuning],[No, keep existing tuning].

If you select [Yes, but start from previous tuning] it will prompt you to outline the arena, and verify the mouse segmentation again, but it will start the segmentation classifier training from the previous end point.

If you select [Yes, ignore the previous tuning], it will prompt you to outline the arena, and verify the mouse segmentation again, but it will start the segmentation classifier training from the default values. If you select [No, keep existing tuning], it will use the classifier trained on the previous clip to segment the new files.

After you make your selection, a message box will pop up that asks "Do you want to run tracking now? Y/N". Select "Y" to track more.

Look at results

Click the results button

This will pop up a window with the first frame of the tracked video, and colored ellipses over the mice. You can scroll through the video, using right and left arrows, or by clicking the play/stop button. You can change the playback speed using the slider on the right.

Looking at track data

Data files

Running motr produces several three directories: Jobs, Tuning and Results

The track files are in Results/Tracks and are named with "_track.mat" suffix after the name of the video file with which they are associated.

Track file format

The track files are MATLAB .mat files and contain two variables: astrctTrackers and strMovieFileName (which, as the name implies, is a string with the name of the movie file in it).

motr tracks the mice by fitting an ellipse around each mouse. astrctTrackers is a 1 x [number of tracked mice] struct with fields describing the position, size and orientation of the ellipse:

m_afX: x position of the ellipse centroid in pixels

m_afY: y position of the ellipse centroid in pixels

m_afA: major axis of ellipse in pixels

m_afB: minor axis of ellipse in pixels

m_afTheta: angle of the ellipse in radians

JAABA

JAABA (the Janelia Automatic Animal Behavior Annotator) can read in motr tracks. This is a very powerful and flexible tool for analyzing mouse movement trajectories. You can download JAABA and get detailed instructions for using it here: jaaba.sourceforge.net.

Miscellaneous notes

If you would like to express your mouse tracks in meters (or feet or furlongs), you'll need to calculate your pixel-to-meter factor. The easiest way to do this is to just put a ruler onto the floor of the arena, take a picture with the camera, load the picture into MATLAB and use `imagesc` and `ginput` to measure the number of pixels in your chosen distance unit.

Appendices

List of useful materials

IR lights:

Outdoor IR Illuminator, 850nm, 30 degree beam, 12VDC or 24VAC (Reytec imaging; part # IR-LT30)

IR light power supply:

24 VAC power supply (Reytec imaging; part # C24AC-40)

camera:

Basler A622FM 2/3" CMOS FireWire.A Monochrome Camera (Edmund Optics; part # NT58-297)

Lens:

16 mm Fixed Focal Length Lens with Manual Focus and Iris, C-mount, 2/3" format, F-stop: 1.4,

IR-pass filter:

Hoya RM72 Infrared filter (B&H Photo; HOIR7252), passes above 720nm

Hair bleach:

Nice 'n Easy Born Blonde Maxi

Git:

tortoisegit.net

NX client:

<http://www.nomachine.com/download.php>

Hysterically brief git primer

First download:

Once you've installed git on Windows, when you right click you have a drop down menu that includes some git commands. To download the data from github, right click to display the menu and select "Git Clone...". This will pop up a window with that says "Clone existing repository" at the top. Enter

<https://github.com/motr/motr> for the URL and directory on your computer where you'd like to store the motr files for Directory (e.g. `c:\USERS\egnor\motr`) and then click OK.

Subsequent downloads (updates):

To stay current as improvements are made to motr, you can update using git. If you'd like to update the files you already have, just right click on the motr directory and select TortoiseGit, and then select Pull from the drop down menu.

What is in all those directories?

In general, with the exception of the `*_track` files, all the files and directories created by motr while running can be deleted once you are happy with the `*_track` files. Below is a more detailed description so you can decide what to keep for yourself.

Jobs directory

The Jobs directory contains a sub-directory per tracked video file. Inside, there are many files (`JobInXXX.mat`, each for a submitted job). Those files contain the initial conditions (where the mice are), and which frames should be tracked. Once final results are available, those files can be discarded.

Results/[individual video file] directories

These directories contain files called "JobOut". Those have the tracked position of the mice in the short interval of that job (along with classifier's data about head/tail and ID). These can be discarded once you have the final results.

Results/Tracks directory

This is where the `[videofilename]_track.mat` files are. These track files contain the trajectories of the individual mice.

Tuning directory

The Tuning directory contains a `Tuning.mat` and `Identities.mat` file, as well as a directory for each single-mouse video, with an single `Identities.mat` file in it (which was used to created the `Tuning/Identities.mat` file). The `Tuning.mat` file contains information about how to do the segmentation. It can be discarded once the track file is finished.

The `Identities.mat` file is the output of the analysis on the single-mouse videos. It contains both the position of the mouse in that video, as well as small image patches extracted at those locations, and the HOG (Histogram of Oriented Gradients) features needed to build a multi-mouse classifier. The `Identities.mat` file contains a struct called `strctIdentityClassifier`, with fields: `m_ astrctClassifiers`, `m_ astrctClassifiersNegClass`, and `m_ a3fReplImages`

The `m_ a3fReplImages` struct has representative images of the tracked mice, which can be useful in determining which track is which mouse, without having to compare the tracks to the original video (e.g. track 1 is the mouse with the spots, track 2 is the mouse with the horizontal stripes, etc).

Running motr on a cluster

A. If you are going to access the cluster from Windows, download and install an NX client
<http://www.nomachine.com/download.php>

B. Login to the cluster

C. `qlogin` to avoid running MATLAB on the head node

At the command line enter: `qlogin -l interactive=true`

This will prompt you for your password.

D. make sure your `.bashrc` file has the correct path to MATLAB

If you get a complaint about finding MATLAB, then you don't have MATLAB in your path (i.e. you need to go to your `.bashrc` file and add the following to export `PATH=`

`:/usr/local/matlab-2011b/bin:`

e.g. `export PATH="$PATH:/usr/local/matlab-2011b/bin"`

once you modify the `.bashrc` file you need to update it by typing [`source ~/.bashrc`] at the command line and hitting return.

E. if `./build` doesn't work

open matlab

`>> fnSetUpFolders`

`>> fnCompile`

F. To check on the jobs:

At the cluster command line, type: `qstat`

e.g. `[egnorrr@e02u19 egnorlab]$ qstat`

job-ID	prior	name	user	state	submit/start at	queue	slots	ja-task-ID
1940700	5.97500	MouseSingl	egnorrr	r	09/13/2012 10:28:47	f10.q@f10u09.ext.janelia.org	1	
1940701	0.00000	MouseID_Jo	egnorrr	hqw	09/13/2012 10:28:46		1	

`r=running`, `hqw=waiting in the queue`

there will be N `MouseSingl` jobs, and one `MouseID` job

to monitor the status of the `MouseSingl` jobs, can `cd` to the `Jobs` directory in the experiment directory and type `more` at the command line (e.g.):

`[egnorrr@e02u19 Jobs]$ more MouseSingleJob.o1940700`

or

`[egnorrr@e02u19 Jobs]$ tail -f MouseSingleJob.o1940700`

(`tail` looks at the end of a file, and `-f` = "follow", so it doesn't return, but updates)

(note, this never terminates, need to `^C` out of it)

other ways to know when done: cursor stops spinning

or

ls `-ll` at the command line, and look at the size of the error files (e.g. `MouseSingleJob.e1940700`. You want the error files to have size 0, i.e. no errors).

G. When you run tracking, you should see the following at the MATLAB command line:

Setting initial ellipses for all jobs

Setting ellipses at frame 1

Setting ellipses at frame 5001

Setting ellipses at frame 10001

.
. .
.

Setting ellipses at frame 520001

0 key frames will be inserted due to frame drops!

Done!

Generating submit script named /groups/egnor/mousetrack/mousevideo/Jobs/Test_B_1/submitscript

Submitting jobs

```
qsub -t 1-157 -N MouseJob -e /groups/egnor/mousetrack/mousevideo/Jobs/Test_B_1/ -o
```

```
/groups/egnor/mousetrack/mousevideo/Jobs/Test_B_1/ -b y -cwd -V
```

```
'/groups/egnor/mousetrack/mousevideo/Jobs/Test_B_1/submitscript'
```

Your job-array 586095.1-157:1 ("MouseJob") has been submitted

Generating submit file at

```
/groups/egnor/mousetrack/mousevideo/Jobs/Test_B_1/submitPostProcessScript
```

Submitting jobs

```
qsub -N MousePostProcessJob -hold_jid MouseJob -e
```

```
/groups/egnor/mousetrack/mousevideo/Jobs/Test_B_1 -o
```

```
/groups/egnor/mousetrack/mousevideo/Jobs/Test_B_1 -b y -cwd -V
```

```
'/groups/egnor/mousetrack/mousevideo/Jobs/Test_B_1/submitPostProcessScript'
```

Your job 586096 ("MousePostProcessJob") has been submitted