06Copper Lab Behaviour Protocol V1

Background

Characteristics of the Mouse Model:

The ATP7B Knockout mouse, a genetically modified model, is designed to replicate aspects of Wilson Disease, a genetic disorder caused by mutations in the ATP7B gene responsible for regulating copper metabolism. In our model, a portion of exon 2 of the ATP7B gene is replaced with a neomycin selection cassette, resulting in abnormal splicing, frameshift mutation, and an unstable full-length ATP7B protein.

Behavioral Studies:

We assess behavioral parameters to explore the genetic modification's impact on motor coordination, cognitive function, and emotional responses. Conducting tests such as the open field, object recognition, rotarod, beam walk, and sucrose preference tests, we evaluate exploratory behavior, memory, motor coordination, and hedonic responses. By comparing ATP7B Knockout mice with control mice, we aim to identify behavioral phenotypes related to the ATP7B gene knockout.

Knockout and Heterozygous ATP7B Mice:

We are also investigating heterozygous ATP7B mice, as recent clinical observations suggest that a single dysfunctional ATP7B gene copy may impact phenotype, indicating potential gene dosage effects. We aim to understand how heterozygous mice exhibit subtle behavioral alterations compared to wild-type mice.

Connecting Behavioral and Biological Findings:

In preclinical mouse models, researchers often establish connections between behavioral testing and cellular and molecular biology by correlating observed behaviors with underlying biological changes. This approach allows for a better understanding of how genetic modifications or disease-related alterations affect both behavior and the biological processes within the organism.

For our study, we will analyze the behavioral data from the ATP7B Knockout and heterozygous mice and relate it to cellular and molecular changes, such as alterations in brain inflammation, structure, and the expression

of copper and zinc handling proteins. This will help us identify the potential pathways and mechanisms through which ATP7B gene dysfunction impacts cognitive, motor, and emotional functioning in the mouse.

Brief background on Bio portion of the project

Brain Inflammation and Structure:

We will study the effects of disabling the Atp7b gene on brain inflammation and structure, treating mice with various therapies and measuring metal levels, oxidative stress, and inflammation. We will examine brain weight, structure, and neuron numbers using ICP-MS for metal concentrations, immunofluorescence for inflammation markers, and additional methods for oxidative stress assessment.

Age and Brain Region Effects:

Our objective is to understand how age and brain region influence copper buildup and brain abnormalities, collecting tissue samples and analyzing differences in treatment and gene types. We anticipate increased inflammation and reduced neuron density in Atp7b-/- mice.

Copper and Zinc Handling Proteins:

Furthermore, we will investigate how copper and zinc handling proteins change in the brain when treated with different therapies. Utilizing a proteomics approach, we will examine brain regions affected by WD, focusing on the impact of copper buildup on zinc-related targets. Our aim is to identify changes in key zinc machinery in the brain.

Treatments

Mice will be treated with either Zinc (Zn) or DPEN, which have potential therapeutic effects on Wilson Disease. We will administer these treatments to Atp7b-/-, Atp7b+/-, and wild type mice and assess their impact on trace metal concentrations and oxidative stress markers in the brain.

Additionally, we will evaluate the effects of these treatments on the animals' performance in our behavioral tests. This will allow us to understand how Zn and DPEN treatments influence cognitive, motor, and emotional functioning in the mice.

By comparing treatment outcomes across different genotypes, we hope to gain insights into the effectiveness of Zn and DPEN in alleviating molecular phenotypes associated with Wilson Disease, as well as their impact on behavior.

Treatment Timing

Treatment for behavior starts at either 6 weeks and continues to 4 months or it begins at 4 months and continues to 6 months based on group assignment.

Genotyping

Jason figured out what Transnetyx uses for their qPCR

Zoey offered to run the old protocol

Here is the protocol for what I tried to isolate the ATP7b -/- amplicon: Amlpicon attempt The primers and MM should be labeled in the small freezer and a print out and labeled tube are on the shelf above the desk with the old XPS computer.

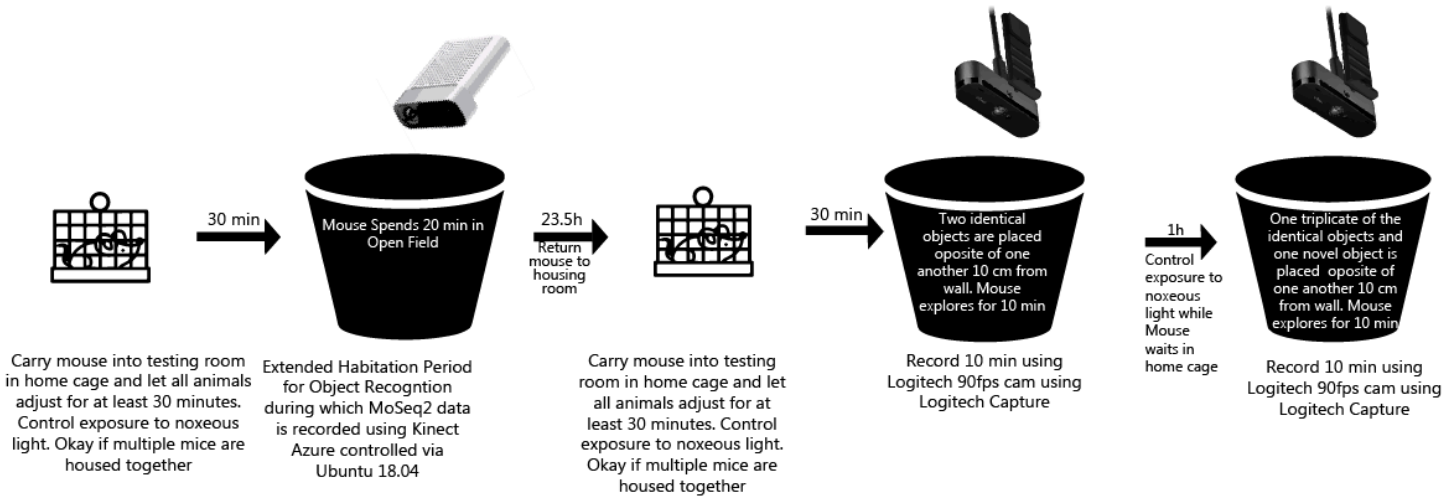


Figure 1: Combined MoSeq2 and Object Recognition Data Collection with Kinect Azure and Logitech Capture

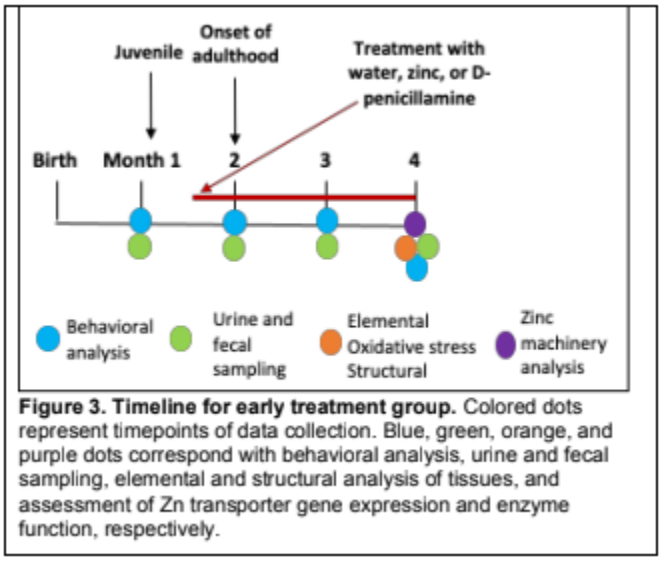


Figure 3. Timeline for early treatment group. Colored dots represent timepoints of data collection. Blue, green, orange, and purple dots correspond with behavioral analysis, urine and fecal sampling, elemental and structural analysis of tissues, and assessment of Zn transporter gene expression and enzyme function, respectively.

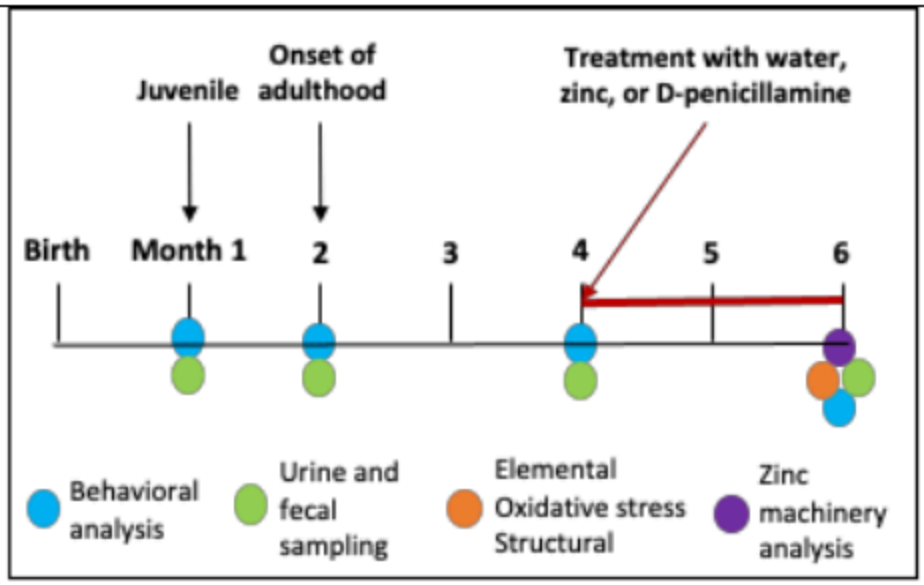


Figure 4: Timeline for late treatment group. Colored dots represent timepoints of data collection. Blue, green, orange, and purple dots correspond with behavioral analysis, urine and fecal sampling, elemental and structural analysis of tissues, and assessment of Zn transporter gene expression and enzyme function, respectively.

Experiment Timing

Tag and ear tissue sample at 3-4 weeks, update metadata with genotype when known

Behavior testing begins at 4 weeks of life for the animal, repeat based on group assignment and occur in the following sequence:

- Day 1= 30 min acclimatization to test room in home cage -> Open field, 20 min test-> 24 hr rest
- Day 2= 30 min acclimatization to test room in home cage -> Object Familiarization, 10 min test (two identical toys)->1 hr rest -> Novel Object Recognition, 10 min test (one different toy and a triplicate of the habituated toys) -> 24 hr rest
- Day 3= 30 min acclimatization to test room in home cage -> Elevated Zero Maze 5 min test -> 24 Hour rest
- Day 4= 30 min acclimatization to test room in home cage -> Rotarod testing, 2 attempts back to back -> 10 min break -> 2 attempts back to back-> Can go into metabolic testing/urine & fecal collection as needed

It seems that it's more practical to run an entire group of siblings housed together at the same time since the cage mates come along anyway. When litters are too big some animals can be sent into biomarker protocol. This seems like an issue that will be quickly iterated based on time and experience

Jason does not want to do Sucrose Preference testing as he believes it will be low yield based on preliminary data (5/5/23)

Timing, what is important and why?

Keep it during wakefull period and keep the timing between tests/stages as close to described in collection section as possible. Simpler then dealing with another variable plus more reliable data

1. Timing between Object Familiarization and Recognition
    a. An hour is supposed to be plenty but I'm not sure why? What happens when it's +/- n minutes?
2. Timing between habituation to Open Field and beginning object recognition (OR) protocol
    a. ?
3. Having at least 24 hours to between OR and Elevated zero maze and again for Rota Rod (allowing mice to reset/destress)
    a. Why?
4. What happens when timing isn't perfect due to experimenter logistics?
5. Robust behavior vs unwanted confounders
    a. Do we keep the data that isn't perfect then compare it to well executed recordings?

Beam Walk will be done at 4 or 6 months of age dependent on group and will only be done once for each animal (animals still get multiple exposures as Beam Walk requires training)

Group Assignment

Animals can be assigned to either early or late treatment (fig 3 vs fig 4)

As they turn 4 weeks they get ear punched and we tag them, decide based on info at the time what group you want to assign the animals to and how many animals we can handle putting through behavior at the time.

We don't know their genotype at 4 weeks but we can use this to guess when deciding what animals to prioritize entering into breeding or into experiment:

WT x WT cross (AA x AA):
a. WT: 100% (1)
b. HET: 0%
c. KO: 0%

WT x HET cross (AA x Aa):
a. WT: 50% (1/2)
b. HET: 50% (1/2)
c. KO: 0%

WT x KO cross (AA x aa):
a. WT: 0%
b. HET: 100% (1)
c. KO: 0%

HET x HET cross (Aa x Aa):
a. WT: 25% (1/4)
b. HET: 50% (1/2)
c. KO: 25% (1/4)

HET x KO cross (Aa x aa):
a. WT: 0%
b. HET: 50% (1/2)
c. KO: 50% (1/2)

KO x KO cross (aa x aa):
a. WT: 0%
b. HET: 0%
c. KO: 100% (1)

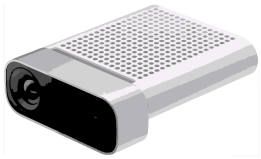
Estimated Number of Mice Needed Per Group

Blindedness

Mouse Body Temps

Initially the mice seemed to get cold as they explored the open arena. We have since increased the temperature of the testing room but it would still be good to note the before and after body temps of the mice via stomach temperature scans to make sure they are not getting cold during experiments. Standardize how the animal is handled.

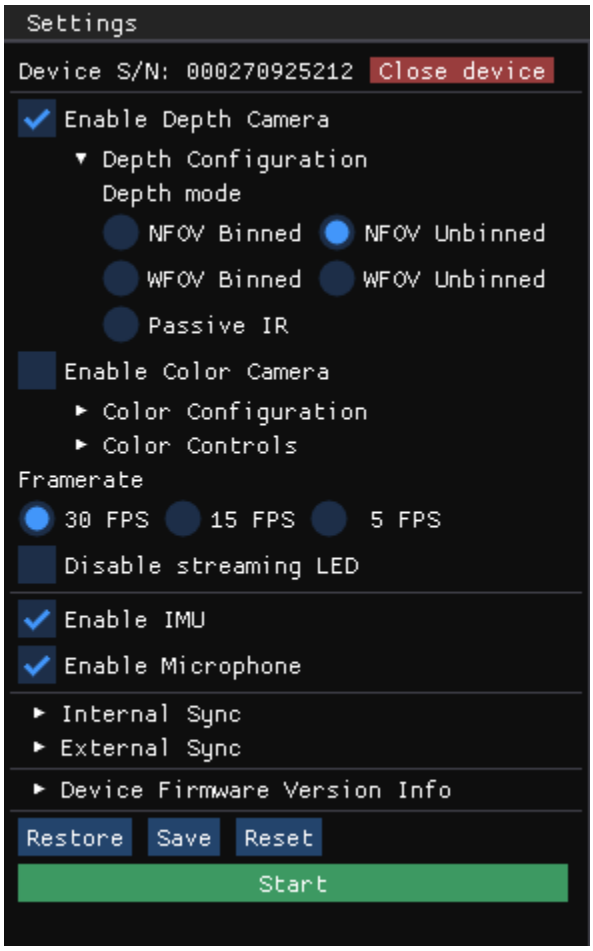
Discontinue temp checks if findings are unremarkable. Note that the temperature of the testing room should be similar to the home room. UAF uses the same dimensions of room for testing if a consult about the environment is helpful at the time.



It is a bit difficult remembering to take both the before and after temp.

MoSeq2 Camera Settings

1. Open Azure Kinect Viewer (Win key + S and search Azure Kinect Viewer)
2. Select the recording device device from the drop-down list (Azure must be plugged into USB 3.0 port on PC)
3. Choose the device and note the serial number. (Can click start to ensure the camera is seeing what it is supposed to)



4. Select the device to configure and click “Open Device” button to open the device. Enable color per Luk?
5. Check Enable Depth Camera stream and select NFOV (Near Field of View) Unbinned for Depth mode.
6. [Recommended] Uncheck Enable color camera.

7. Select 30 FPS for framerate.
8. Click Save to save the settings.
9. Click Close device. You must close the device before you exit the program.

Lighting

The mice seem to hang out in the bottom left corner of the arena. I wonder if it is because there is less light there due to the room light falling left to right. I added an additional light to the rack next to the camera stand to see if mice freeze in a different spot in the arena because of more uniform lighting.

Adjusted second light angle by minimizing shadow cast by mouse recognizable objects.

Is too much light during open field testing a bad thing? How is this standardized in other experiments

Using the Kinect Azure Recorder via Windows Command prompt

1. Make sure there is enough room on the drive you are trying to record the animal behavior
2. Open a Windows Command Prompt as administrator by typing cmd into the Windows Search bar and open the terminal as administrator
3. At the beginning of a session create a folder to save all of the videos that will be processed together (maybe all animals at 1 month old or as a certain number of videos accumulates) by running the command:

```
mkdir "D:\base_dir\Name of your test session"
#You only need to run this command once for all of the recordings that will go into the folder
```

4. Highly advocate running a five second test recording during animal acclimation to ensure all is set up and ready to go! This can be done with the following command (will just need to edit in your file name): "C:\Program Files\Azure Kinect SDK v1.4.1\tools\k4arecorder.exe" -l 005 -d NFOV_UNBINNED -c OFF -r 30 --imu OFF D:\base_dir\filename\test.mkv CHECK VERSION: NEW SYSTM SDK TOOLS IS 1.4.2

New system :"C:\Program Files\Azure Kinect SDK v1.4.2\tools\k4arecorder.exe" -l 005 -d NFOV_UNBINNED -c OFF -r 30 --imu OFF D:\base_dir\filename\test.mkv

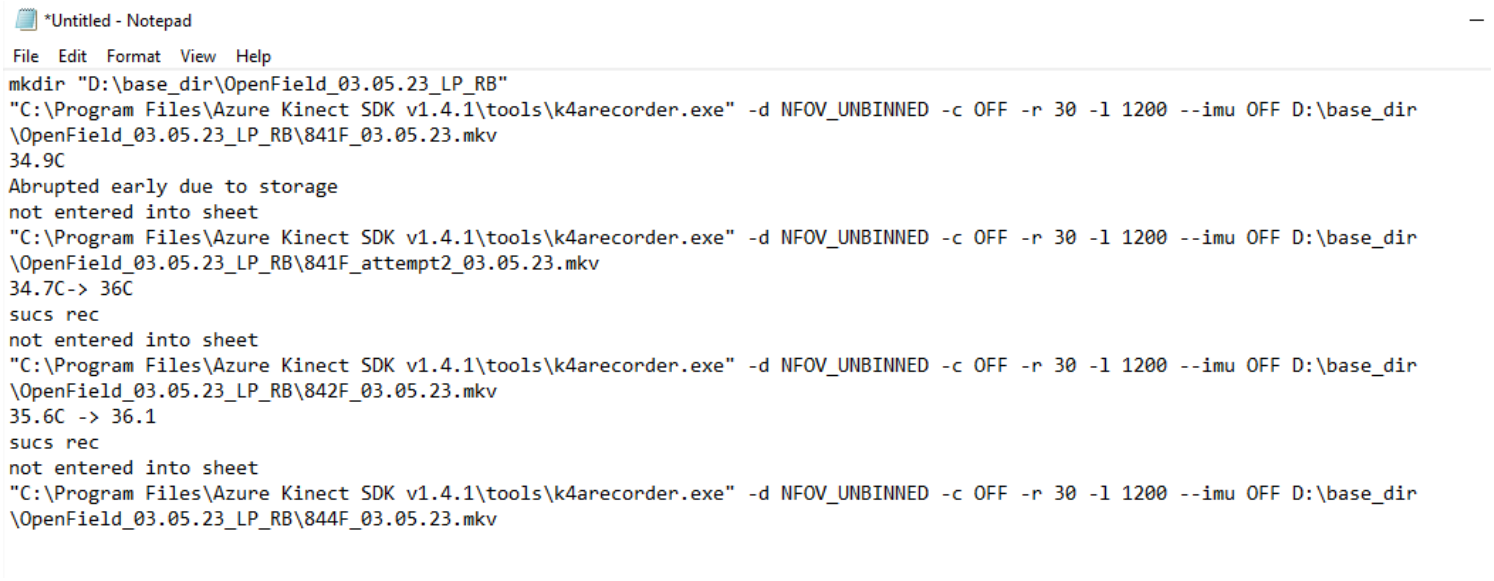
5. To initiate the recording edit the following command and paste it into the terminal to initiate the recording of a 20 minute Open Field video (Make sure that you give the video file a unique name and that file name in the command matches the folder you created in the previous step):

```
"C:\Program Files\Azure Kinect SDK v1.4.1\tools\k4arecorder.exe" -l 1200 -d NFOV_UNBINNED -c OFF -r 30 --imu OFF D:\base_dir\Name of Your Test Session\Tag Number of animal and date.mkv
```

6. Instead of copy and pasting this argument every time for each animal, after your first animal, you can click the up arrow key, which will autopopulate with your last command, allowing you to simply edit in the next animal's tag number!
7. At the end of recording session you can either process the videos or move them to the H drive for storage
8. Make sure to input the metadata into sheet and label the recordings

NOTE: turns out you cannot transfer files between drives while recording with kinect azure. If you find yourself in this situation (or any requiring termination of recording) you can use CTRL - C

Example of notes from a recording session:



MoSeq2 and Open Field (OF) Habituation Testing Protocol

Timing: 25 min per mouse

1. Let Each mouse adjust to the testing room for at least 30 minutes after transfer from home room in home cage (Place home cage on first shelf from the bottom on the rack against the O wall, cardboard box to keep light out)
2. ~~Check the animal tag, take body temp reading on stomach using laser thermometer (the rack on the shelf was meant as an option to access belly temp better, not sure if its the best option)~~
3. Ensure Kinect Azure Sensor is ~26.5 Inches from Bottom of arena and arena is pressed snug against both legs of 80/20 camera stand
4. Enter Mouse and test identifying information into Google Sheet
5. Edit the command to Include mouse specific data:

```
"C:\Program Files\Azure Kinect SDK v1.4.1\tools\k4arecorder.exe" -d NFOV_UNBINNED -c OFF -r
30 -l 1200 --imu OFF D:\base_dir\Name of Your Test Session\Tag Number of animal and date.mkv
```

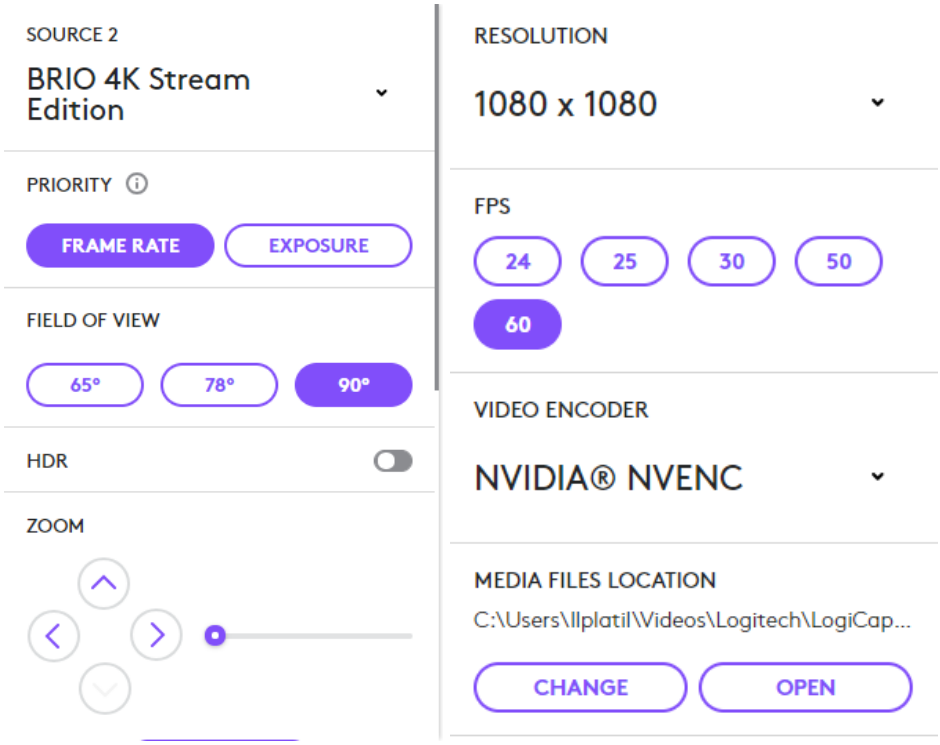
6. Begin Recording the MoSeq2/Habituation footage, via path specific command in terminal by pressing enter
7. Place the mouse in the empty arena, facing the wall nearest to the experimenter, and allow it to explore the arena for 20 minutes while collecting MoSeq2 footage.
8. ~~Take a second body temperature reading and return the mouse to its home cage.~~
9. Clean the arena with 70% (vol/vol) ethanol and 10% bleach to minimize olfactory cues (Rotating buckets really helps).
10. The recorded MoSeq2 video will be saved as you designated in the command and it can be helpful to verify successful recordings in the File Explorer and that the file has a unique name

Still need to address the file sizes (too big and don't know a good compression strat atm that is compatible downstream)

Object Recognition Camera Settings and Position

1. Plug in the BRIO 4K Stream Edition camera into a USB 3.0 Port (designated by blue teeth on computer end of port)

2. Open Logi Capture via Windows Search bar and adjust the camera settings as necessary for adequate view of the Open Field. With the camera taped to the Kinect Azure on the 80/20 stand these were our settings:



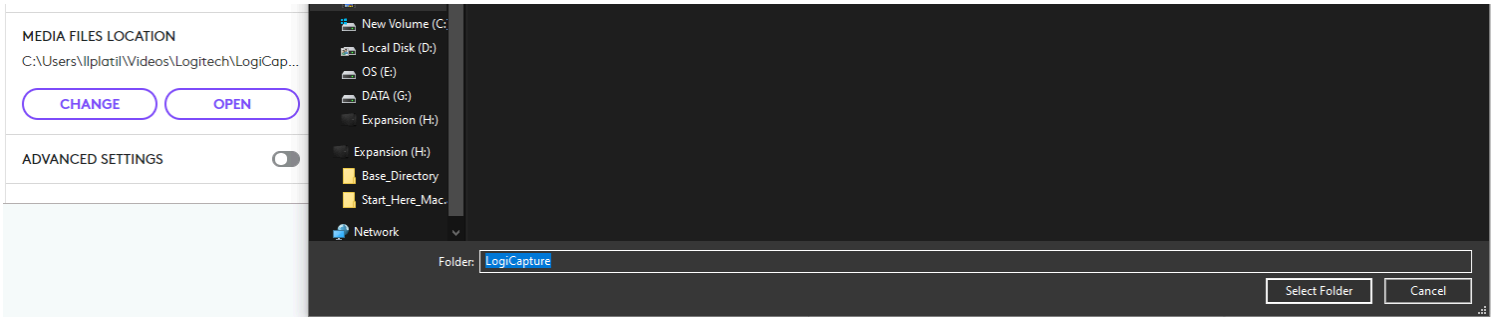
3.

4. Create a labeled folder, make sure to designate Familiarization and Recognition ie:

```
└── Object Recognition
    ├── session_n.familarization
    ├   └── animal_n.mp4
        ...
    ├── session_n.recognition
    └   └── animal_n.mp4
        ...
```

5. Label each video by editing file name to include pertinent metadata

6. Ensure the path to your desired folder is designated in Logi Capture designate where you want to save your recordings within Logi Capture:



7. Ensure you have sufficient space on designated hard drive for all recordings (these recordings are compressed and have far lower storage demands

Object Recognition Familiarization (ORF) Session

Timing: 15 min per mouse

7. Twenty-four hours after the habituation/OF session, assign a mouse to two identical objects in the arena, 10 cm away from the walls. Randomize the pair of objects between each mouse by grabbing the first mouse out of the home cage at random and cycling through the Objects: (3D printed, 3 unique objects, each in triplicate)
8. Let Each mouse adjust to the testing room for at least 30 minutes after transfer from home room in home cage (Place home cage on first shelf from the bottom on the rack against the O wall, the cardboard box is supposed to keep the mice from getting sleepy)
9. Set objects equidistant and ~10 cm from walls in the arena. Alternate between sets of matching objects with each subsequent mouse. (I have tried using magnets + tape measure, markings on back of bucket as well as a jig to estimate object placement. The acid test to determine importance of precision in placing objects will be training DeepLabCut on the first 10 recordings I have on flash drive 5/19/23 -LP)
10. Align the bucket so that the toys are aligned along the horizontal axis of what the camera sees and the camera view is centered and sees the entire field and at least 8 inches of wall (the angle of view button and zoom buttons in Logi Capture are useful)

11. Check the animal tag while mouse is allowed to grab the bars on the third shelf, take body temp reading on stomach using laser thermometer
12. Ensure Logi Capture is ~26.5 Inches from Bottom of arena and arena is pressed snug against both legs of 80/20 camera stand and that one can see the entirety of the arena without shadows via Logi Capture app on desktop
13. Enter Mouse and test identifying information into Google Sheet
14. Initiate recording with animal in hand and place animal in center of arena, facing one away from either object, there is a 3 second countdown on Logi Capture before recording begins. (Initial mouse decision > avoiding experimenter's hand in frame)
15. Record animal for **10 minutes** and try to be accurate about time spent in arena
16. Take a second body temperature reading and return the mouse to its home cage.
17. The recorded file will be where you designated the file to go via path selection in Logi Capture
18. Make sure to label and sort each recording into appropriate Folder

[1 hour rest here per animal]

Post-Familiarization Session Cleanup

Timing: 2 min

1. After the familiarization session, clean the objects and the arena with 70% (vol/vol) ethanol and 10% bleach to minimize olfactory cues.
2. Dry the objects with a paper towel and leave them to air dry. Dry the arena before the next use.
3. Swapping between the two buckets helps
4. I set up a small wash tub for the objects and setting them in the hood helps dry them really quickly

Object Recognition (ORT) Test Session

Timing: 15 min per mouse

14. Prep an arena with one familiar object and one novel object 10 cm away from the walls of the arena (the triplicate of the familiar object can be used to eliminate olfactory cues)
15. Align the bucket so that the toys are aligned along the horizontal axis of what the camera sees and the camera view is centered and sees the entire field and at least 8 inches of wall (the angle of view button and zoom buttons in Logi Capture are useful)
16. Annotate in the appropriate column within the spreadsheet which side the novel object lies on (left / right)

17. Check the animal tag while mouse is aloud to grab the bars on the third shelf, take body temp reading on stomach using laser thermometer
18. One Hour after the Familiarization session, place the familiarized animal in the center of the arena facing away from either object
19. Initiate recording before placing the animal in the arena and let the animal explore for 10 min
20. Take a second body temperature reading and return the mouse to its home cage.
21. The recorded file will be where you designated the file to go via path selection in Logi Capture
22. Make sure to label and sort each recording into appropriate Folder
23. Update metadata in Google Sheet

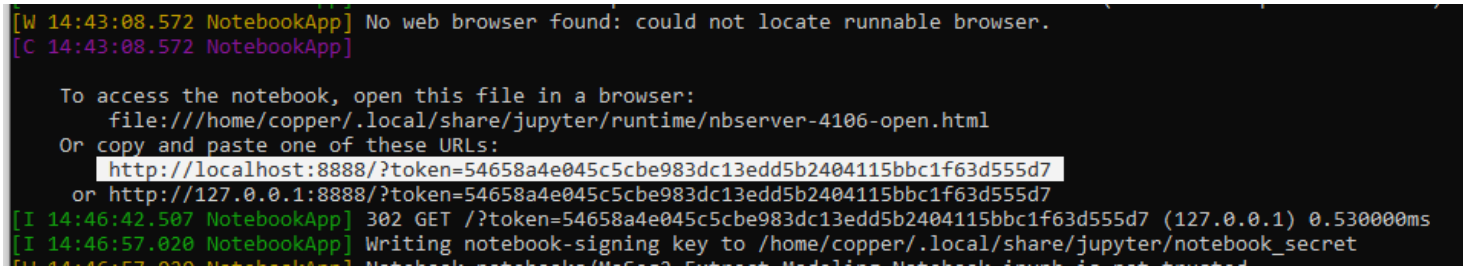
Post-Test Session Cleanup

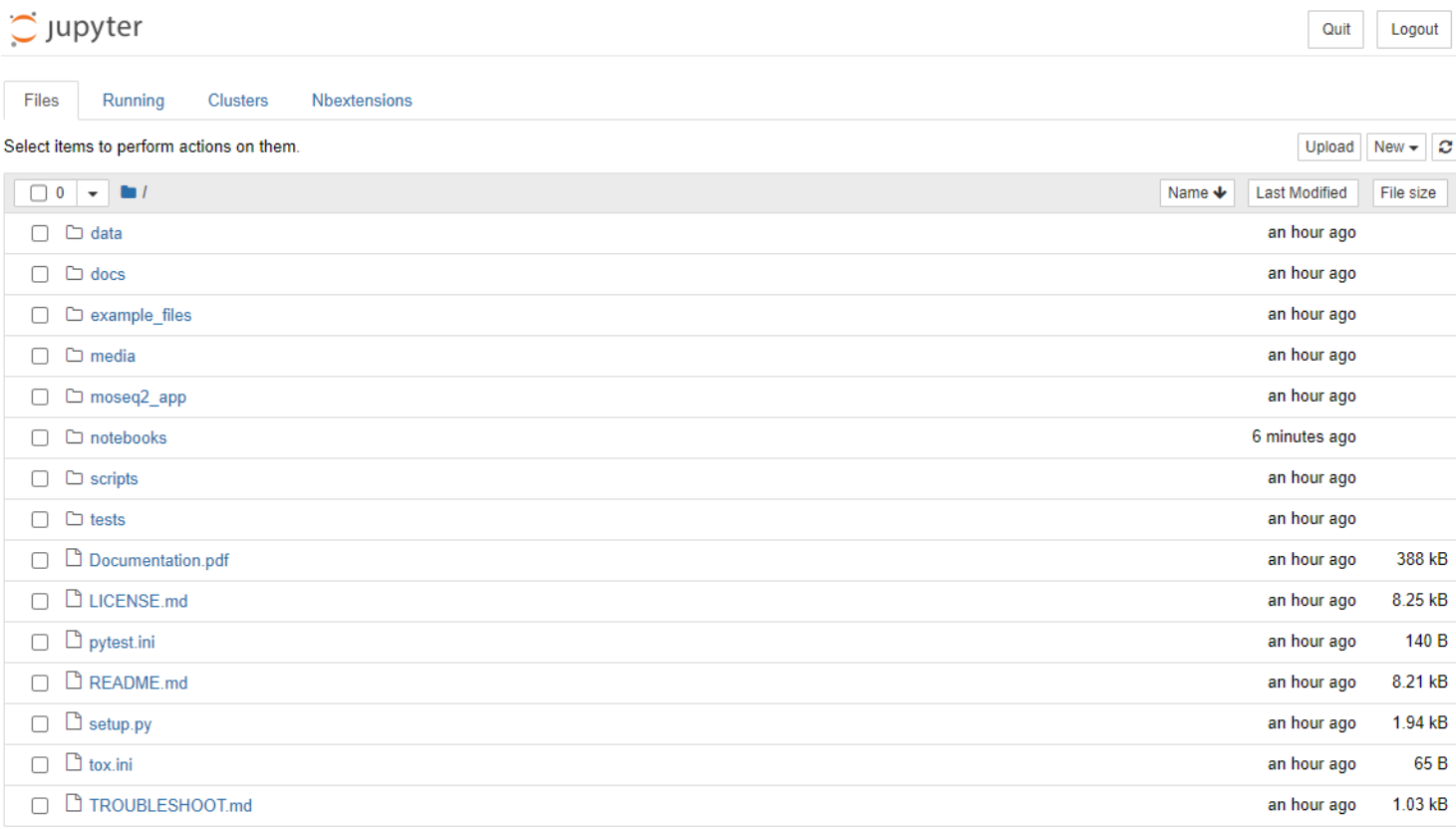
Timing: 2 min

19. After the test session, clean the objects and the arena with 70% (vol/vol) ethanol and 10% bleach to minimize any olfactory cues.
20. Dry the objects with a paper towel and leave them to air dry. Dry the arena before the next use.
21. Helps to set up one arena while the other is being used

Using the MoSeq2 App with Jupyter Notebook as GUI

1. Open Ubuntu 22.04 via Windows Search (Note that Ubuntu 18.04 and 22.04 have very similar icons)
   ***Update* The app works under my profile 9/8/23**
2. Run the Following Commands (highlighted in black)
   a. (base) copper@anc-5rhs1v3:~$ cd moseq2-app
   b. (base) copper@anc-5rhs1v3:~/moseq2-app$ conda activate moseq2-app
   c. (moseq2-app) copper@anc-5rhs1v3:~/moseq2-app$ jupyter notebook
3. Copy the URL from Ubuntu Terminal into Chrome browser

4. 

5. A Jupyter Notebook should open and look something like this:
6. 

Open Field Data File structure for MoSeq2 using Kinect Azure

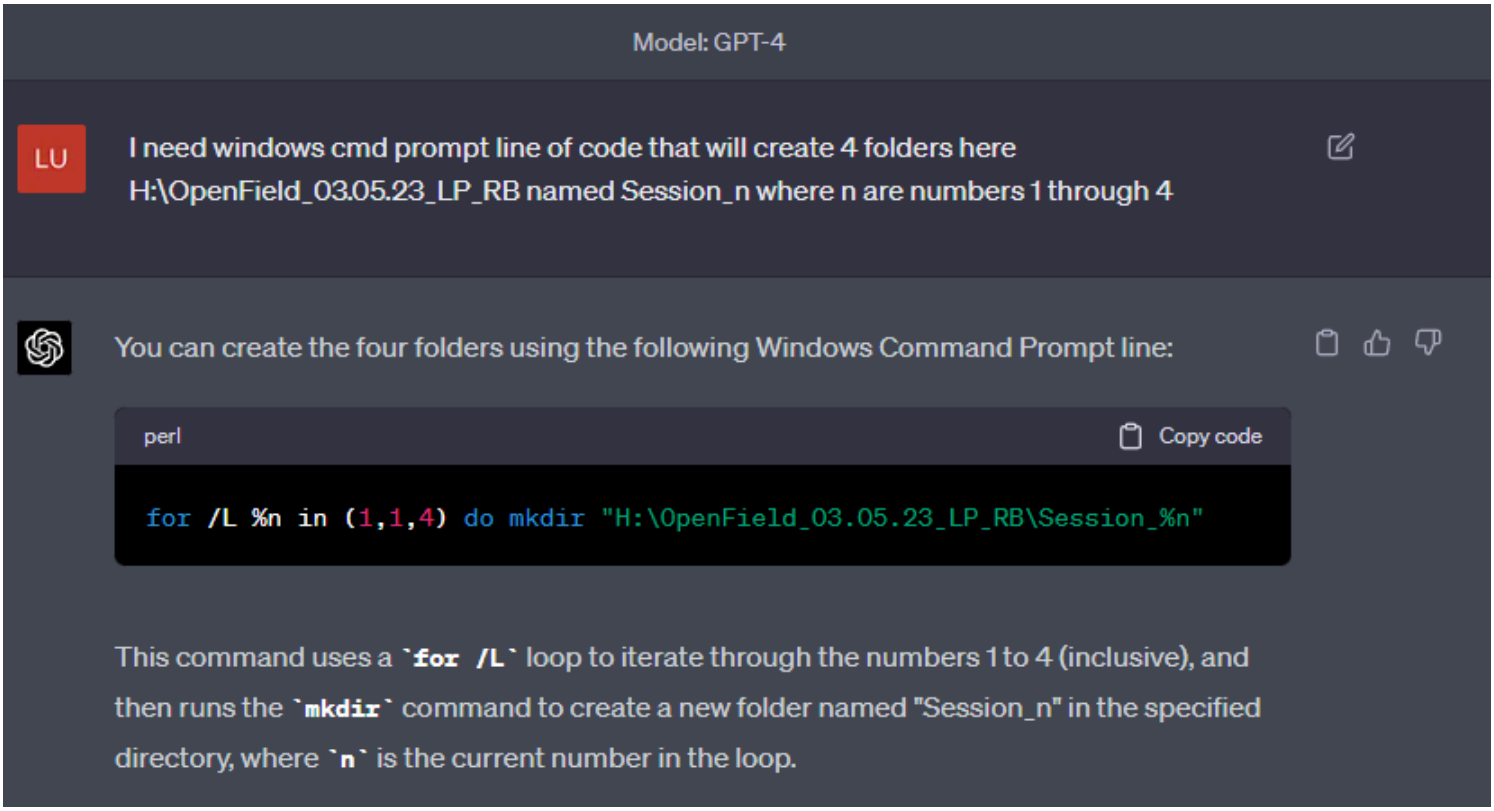
1. Open the Location where you stored the unprocessed .mvk files
2. Create a file structure where each recording is stored in its own folder and all the recordings from the batch to be processed are stored in a folder ie:

Unprocessed Open Field Data

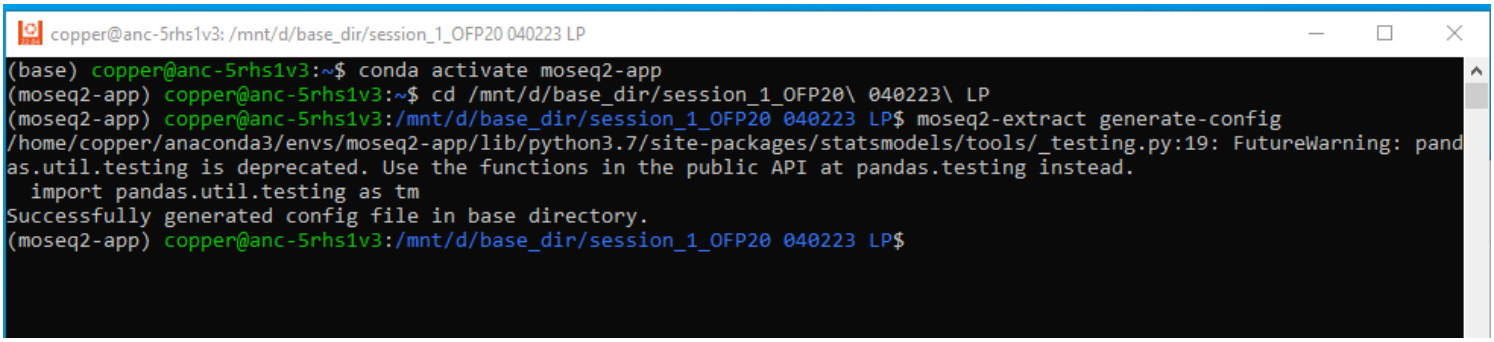
.
└── <base_dir>/
    ├── session_1/

├ └── session_1.mkv
...
├── session_n/
└── └── session_n.mkv

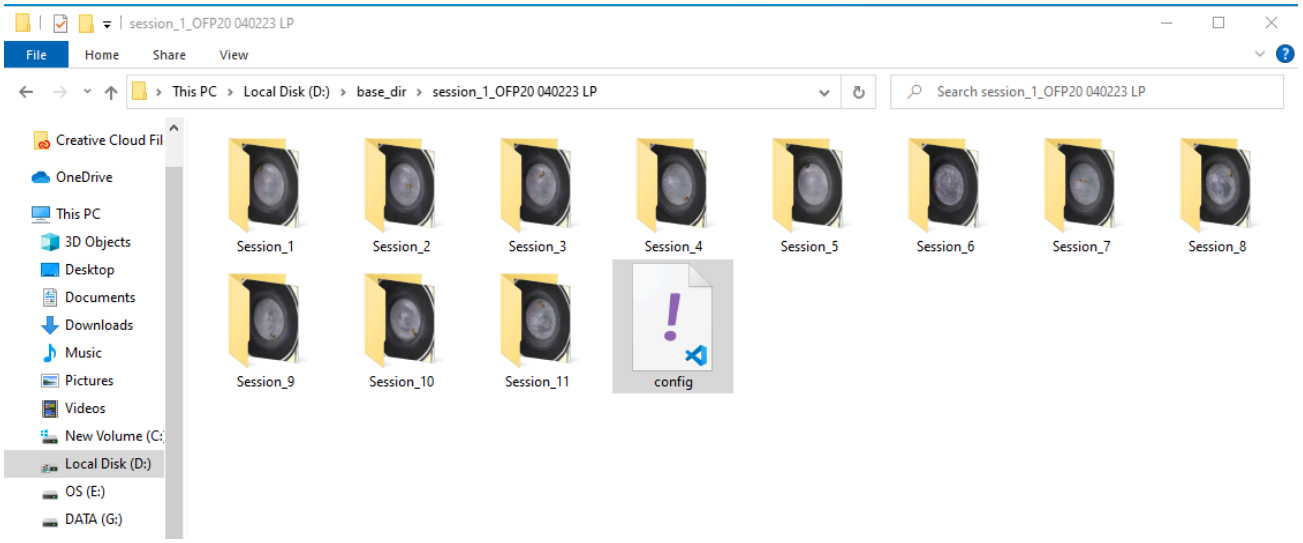
In cmd this can be executed by: for /L %i in (1,1,10) do mkdir Session_%i
Which will create 10 folders into which you can place individual recordings. GPT can edit it for you ie:



3. Create a config.yaml file for the batch of recordings
    a. conda activate moseq2-app
    b. cd /mnt/*location of your file*
    c. moseq2-extract generate-config
    d. Example:



e. Example of output in file directory (note the config file):



4. Jupyter Notebook needs to know where this specific file is located (your recorded and unprocessed data)

Symlinks as a way to manage file location and path structures

A symlink, short for symbolic link, is a file system object that serves as a reference or pointer to another file or directory. Symlinks are useful in cases where you need to maintain a consistent file structure or make files and directories accessible from different locations without duplicating the data. They are particularly helpful when dealing with large files or when storage capacity is limited.

In this specific case, a symlink was needed because the large data folder (D:\base_dir\session_1_OFP20 040223 LP) could not be moved to the MoSeq2 app directory on the 'C:' drive due to insufficient space. The MoSeq2 app required access to this data for processing and analysis. By creating a symlink, we established a reference to the original data folder on the 'D:' drive within the MoSeq2 app directory. This allowed the Jupyter Notebook server running in the moseq2-app environment to access and work with the data without physically moving or duplicating it, thus conserving storage space on the 'C:' drive.

Here are the step-by-step instructions for creating a symlink to a folder in a Jupyter Notebook server running in the moseq2-app environment. These instructions are specific to Windows 10 with WSL2 (Ubuntu 22.04) and assume that you have already set up the MoSeq2 app and Jupyter Notebook server.

1. Locate the folder you want to create a symlink to in your Windows file system. In this example, the folder is D:\base_dir\session_1_OFP20 040223 LP.

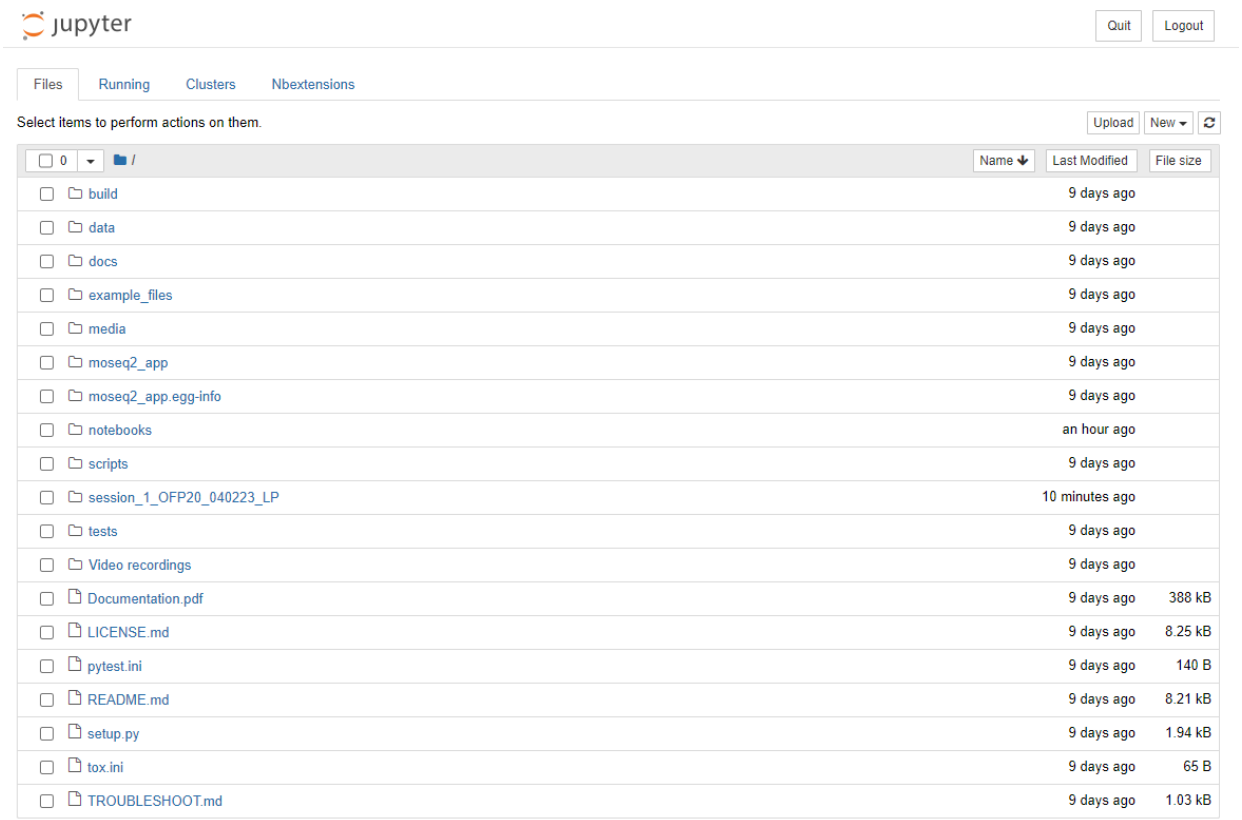
2. Open the WSL2 terminal and navigate to the MoSeq2 app directory. In this example, the MoSeq2 app directory is /home/copper/moseq2-app. Run the following command:

3. cd /home/copper/moseq2-app
4. Convert the Windows folder path to a WSL2 folder path. In this example, the Windows folder path is D:\base_dir\session_1_OFP20 040223 LP, and the corresponding WSL2 folder path is /mnt/d/base_dir/session_1_OFP20 040223 LP.

5. Create a symlink to the folder in the MoSeq2 app directory by running the following command in the WSL2 terminal. Replace the folder paths with the appropriate paths for your setup:

6. ln -s /mnt/d/base_dir/session_1_OFP20\ 040223\ LP /home/copper/moseq2-app/session_1_OFP20_040223_LP
7. Note that you need to escape spaces in the folder name with backslashes (\) in the WSL2 command.

8. After creating the symlink, refresh the Jupyter Notebook files page in your web browser. You should see the symlinked folder session_1_OFP20_040223_LP in the list. Any changes you make to the files within the symlinked folder through the Jupyter Notebook server will be reflected in the original folder on the 'D:' drive. ie:



Moseq2 Potential FIle Compression Strategy

GZIP compress decompress then mosque works on mac, need to test on lab pc

Lossles compression via GZIP then test Moseq2 compatibility:

1. **Install gzip**: If you're using a Unix-like operating system (like Linux or macOS), gzip is likely already installed. If you're using Windows, you can install gzip through a software package like Cygwin.

2. **Compress the .mvk file**: Use the gzip command in the terminal to compress your .mvk file. The syntax is as follows:

```
gzip yourfile.mvk
```

```

This will compress yourfile.mvk and rename it to yourfile.mvk.gz. The original .mvk file will be removed.

3. **Decompress the .mvk.gz file**: When you're ready to use the .mvk file again, you can decompress it using the gunzip command:

```
gunzip yourfile.mvk.gz
```

This will decompress yourfile.mvk.gz back into yourfile.mvk. The .mvk.gz file will be removed.

4.Test the compatibility by adding the decomressed files to your Jupyter Notbook and running the app.

Moseq2 Data Extraction

1. #Open Ubuntu 22.04 for WSL and run the following commands

```
cd moseq2-app
conda activate moseq2-app
jupyter notebook
```

2. Copy and paste one of the URLs to Access Jupyter Notebook
3. Click on Notebooks
4.

Data Extraction:

The Read Me on the MoSeq2 web page does a great job walking one through using Jupyter Notebook as a GUI to use their tool for video processing. Re-stating fragments as I started to here is probably not effective. *Add file structure after compression note* Jupyter Notebook needs to know where the files are, where to put the
```

processing/training/progress files ie make it aware of folder structure and most other hiccups seem to come from dependencies and env issues.

1> Open MoSeq2

```
cd moseq2-app
```

```
conda config --set channel_priority strict
```

```
conda activate moseq2-app
```

2. Run Jupyter Notebook

```
jupyter notebook --ip=0.0.0.0 --no-browser
```

3. Navigate to

```
mv /mnt/c/Users/<your_username>/Downloads/small_test_ds/* session_1/
```

Options:

`--version` #Show the version and exit. [default: False]

`--help` #Show this message and exit. [default: False]

Here are the various cell status indicators in Jupyter Notebook:

1. [ ]: The cell has not been executed yet or has been cleared/reset.
2. [*]: The cell is currently running/being executed.
3. [n]: The cell has finished executing, where n is the execution order number (e.g., [1], [2], [3], etc.).

Commands:

aggregate-results Copies all extracted results (h5, yaml, avi) files...

batch-extract Batch processes all the raw depth recordings located...

convert-raw-to-avi Converts/Compresses a raw depth file into an avi file...

copy-slice Copies a segment of an input depth recording into a...

download-flip-file Downloads Flip-correction model that helps with...

extract Processes raw input depth recordings to output a...

find-roi Finds the ROI and background distance to subtract from...

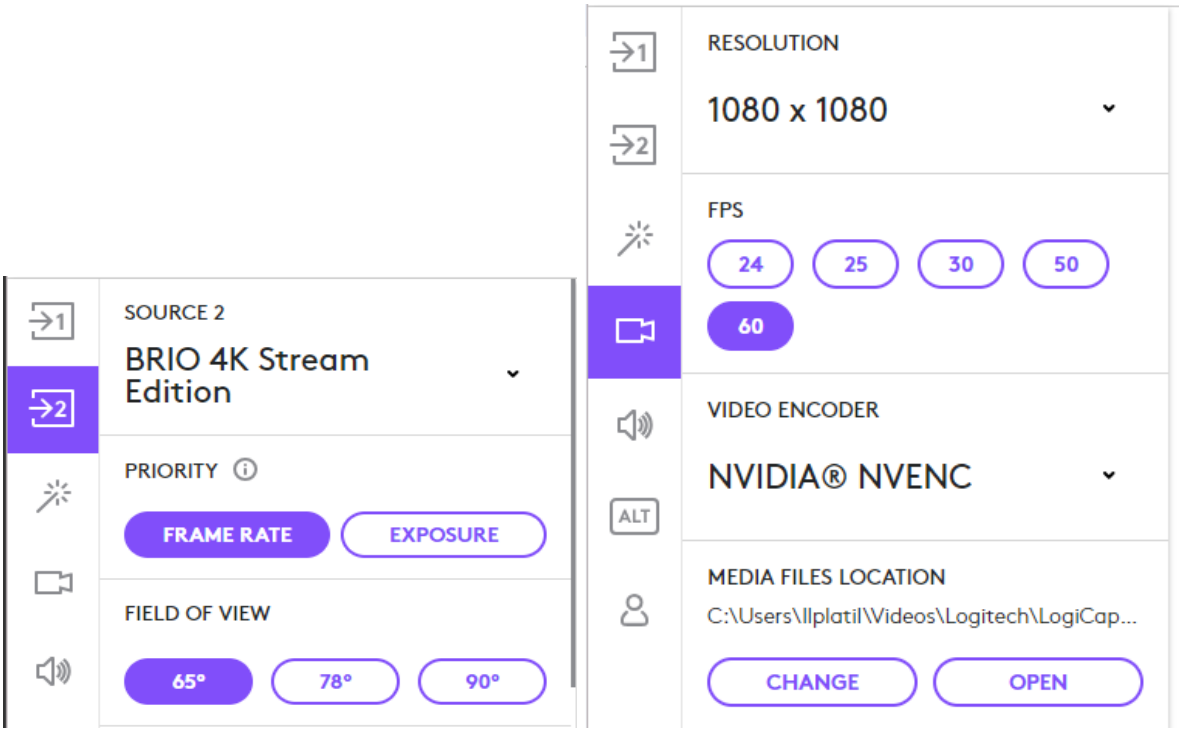
generate-config Generates a configuration file that holds editable...

generate-index Generates an index YAML file containing all extracted...

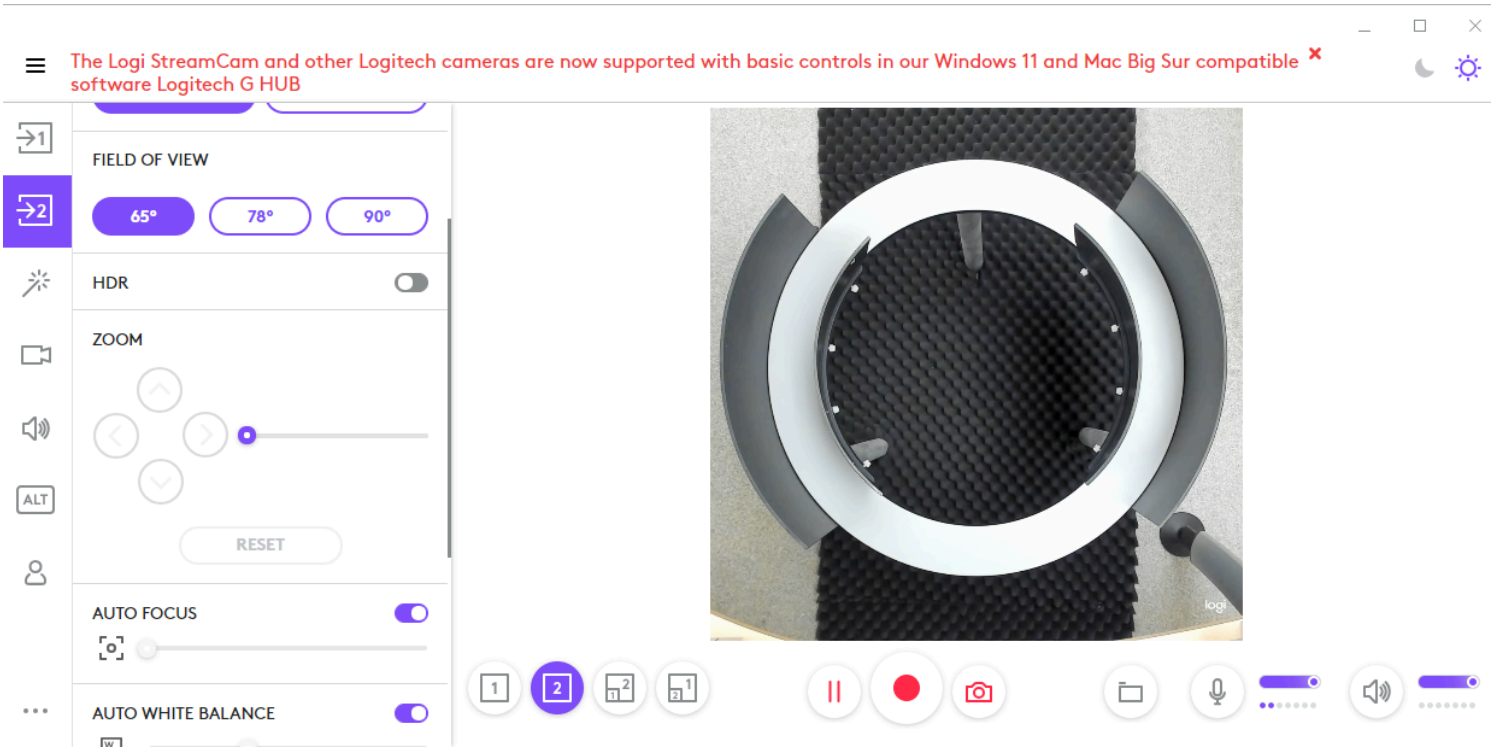
Elevated Zero Maze Set Up

1. Position the BRIO 4k Stream Edition camera above the Elevated Zero Maze so that the entirety of the zero maze is in view and ensure it is connected to the PC via USB. Reference photo below:

2. Open Logi Capture and adjust the camera settings as necessary for adequate view of the Elevated Zero Maze. With the camera stand placed on the table and the maze placed on the floor these were our settings:



3. Ensure there is adequate padding below the Elevated Zero Maze, in the case of a mouse fall. We did this by fitting foam onto the base of the maze.
4. Adjust the maze as necessary to give optimal view of all components in the camera field.
5. Given the layout of the desk, your foot may be visible in the background of the recording. Try to avoid this as it will make it harder for the model to abstract the background



6. The Video file automatically stores where you designate it within the app, for example: C:\Users\YourNameHere\Videos\Logitech\LogiCapture

Elevated Zero Maze Protocol

1. Allow the mice to rest for 24 hours after completing the novel object recognition test. This will help minimize carryover effects and ensure that the mice have adequate time to recover from any stress or fatigue caused by the previous test.

2. On the day of the elevated zero maze test, bring the mice into the testing room in their home cage. This will help reduce stress associated with transportation and handling.

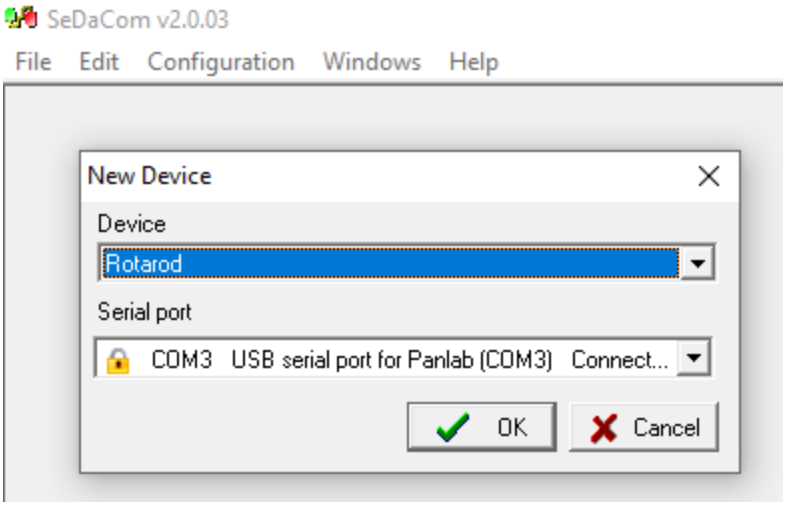
3. Allow the mice to habituate to the testing room environment for at least 30 minutes prior to the start of the elevated zero maze test. Keep the testing room quiet and dimly lit during the habituation period. Ie close the door to the testing room and use the habituation box

4. Enure your camera is ready to record and have a timer set for 5 minutes (Tucker et al 2017)

5. Place the mouse in the center of one of the closed arms of the maze, facing toward the interior wall
6. Start the recording before the animal is placed in the maze
7. Label your video file place it in a session folder
8. Enter Mouse and test identifying information and relevant metadata into Google Sheet
9. Clean the maze with 70% (vol/vol) ethanol and 10% bleach to minimize any olfactory cues
10. Dry the maze with a paper towel and leave them to air dry.
11. While maze is drying label the recording an place it in a folder that designates that the recording has not been processed yet ie: 842F_06.05.23

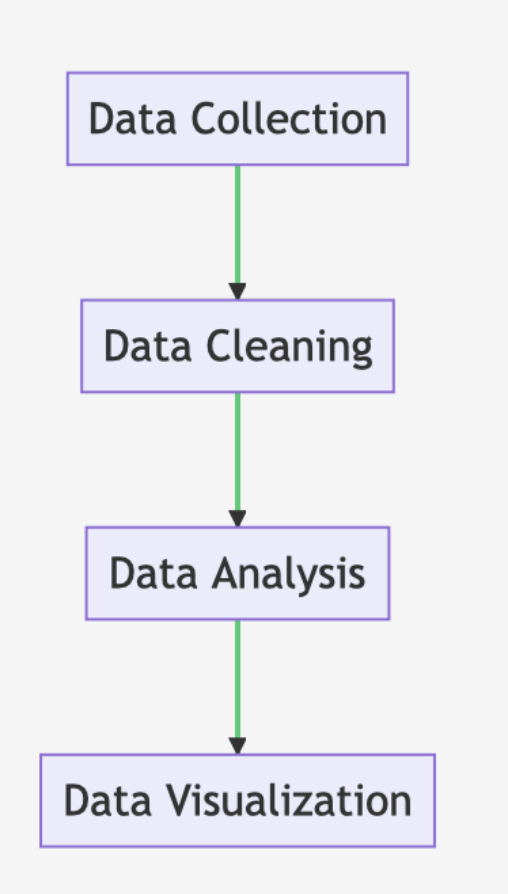
Rotarod Set Up and Data Collection

1. Let the animals acclimate to the testing room for 30 minutes
2. Connect the Rotarod to the computer via USB cable and plug into power
3. Turn on the Rotarod
4. Open the SeDaCOM software -> Windows -> New



5. Set the apparatus to accelerate from 0–40 rpm over 60 sec via touchscreen interface. (This is a good standard for young adult mice, although it should be noted that juvenile and older animals perform poorly at this task.) **11/12/2024 NOTE:** machine is set to this standard! All you, as technician, need to do is hit "RUN" on the touchscreen to initiate the baseline speed.
6. Choose the lanes you will use and flip those levers up while the others are down (If mice are small sensitivity of levers can be adjusted by moving screw on back, but leftmost two levers are pretty well dialed in for our adults)
7. With animal held by the mid-tail, gently swing onto the rod, such that they initially grab with front paws, and then stabilize with rear paws. Once the animal is safely on the rod, push "START" on the touch screen, and allow them to run until they fall off and depress lever.
8. Let each animal take three attempts at the rotarod, resetting levers between each run. If an attempt does not trigger the timer try until you record three attempts.
9. You should be able to see the data as it records in the SeDaCOM GUI
10. Save the data to google docs and export it as a an excel document

Rotarod Data Processing Protocol



1. **Data Collection**: Copy SeDa COM output into Google Sheets.
2. **Data Cleaning**: …
3. **Data Analysis**: …
4. **Data Visualization**: …
5.

Note: Tried to take the data we have, copy it into an excel sheet, add some basic summation and averages then feed it into GPT Data Interpreter, provide context by sharing excerpts from publications that use rotarod data, and then produce excel figures, stat calcs and r stat code. Found that it didn't work well or perform reliably and that as of July 2023 its probably still faster to follow along what others did without gpt + plugins.

Beam Walk

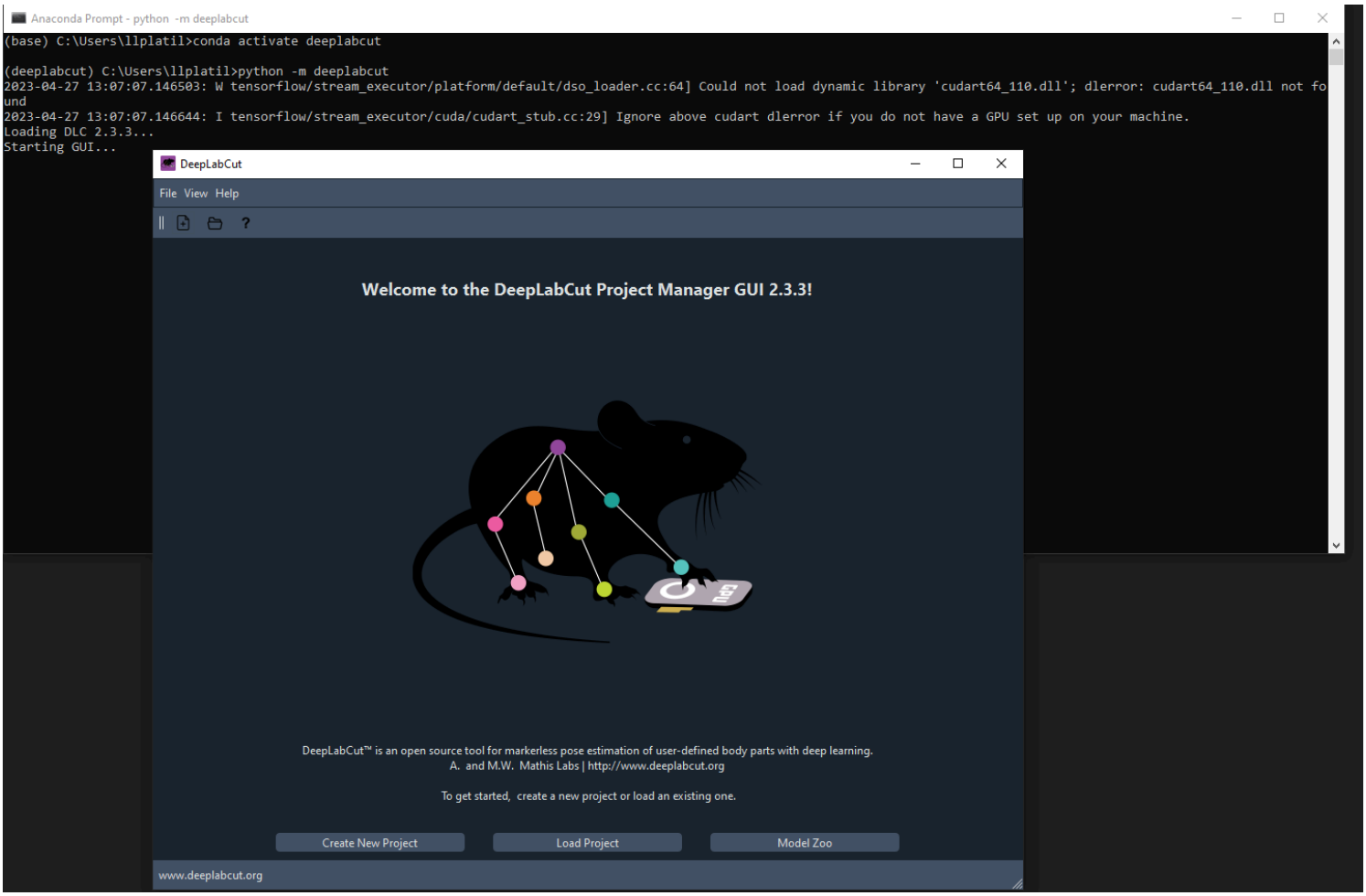
1. For mice, set up a beam approximately 0.6 cm wide and 120 cm in length, suspended about 60 cm above some foam pads. (A larger beam, approximately 1.8 cm wide and 240 cm in length, in addition to a flat platform at one end to rest between trials is required for rats.)
2. Place the animal on one end of the beam (for the rat this would be farthest from the platform). Animals from active strains such as the C57BL/6 mouse or the Long-Evans rat will instinctively walk along the beam to reach the opposite end. Once at this point they will generally turn 180° and continue to walk on to the opposite end. Establish a basal level of performance before surgery or treatment, and allow sufficient time for recovery (at least 24 hr) before retesting.
3. Count the number of foot faults, defined as the number of times the fore-paws and/or hindpaws slip from the horizontal surface of the beam over a predetermined number of steps (50 is usually adequate). Allow the performing animal sufficient time (approximately 5 min) to complete this task. (It is useful to use a mirror on the side of the beam opposite the observer and to videotape the performance for scoring.)
4. Remove to home cage and retest as appropriate. It should be noted that rodents, especially rats, tire and are reluctant to move if exposed to this test repeatedly over a short period on the same day.

8.4.3.2. Variation

This task works well for active rodent strains and may not be suitable for less active animals. Another variation partly designed to address this issue in the rat involves training animals to walk across the beam to a “safe” dark box; the cognitive requirements for this version, however, may influence motor outcome to some degree so care

should be taken here. A simpler approach measures the time taken to fall down onto the foam pads. In this instance, the investigator should vary the beam width until an acceptable latency is found for the particular strain to be used. Attention should also be paid to the body weight of the animal, as the suitable width of the beam may change according to the mouse's ability to grip the edge of the beam, for example, mice heavier than 35 g generally require a beam approximately 0.9-cm thick.

Running DeepLabCut



1. Open Anaconda Prompt from Windows Search as Administrator
2. Activate the environment

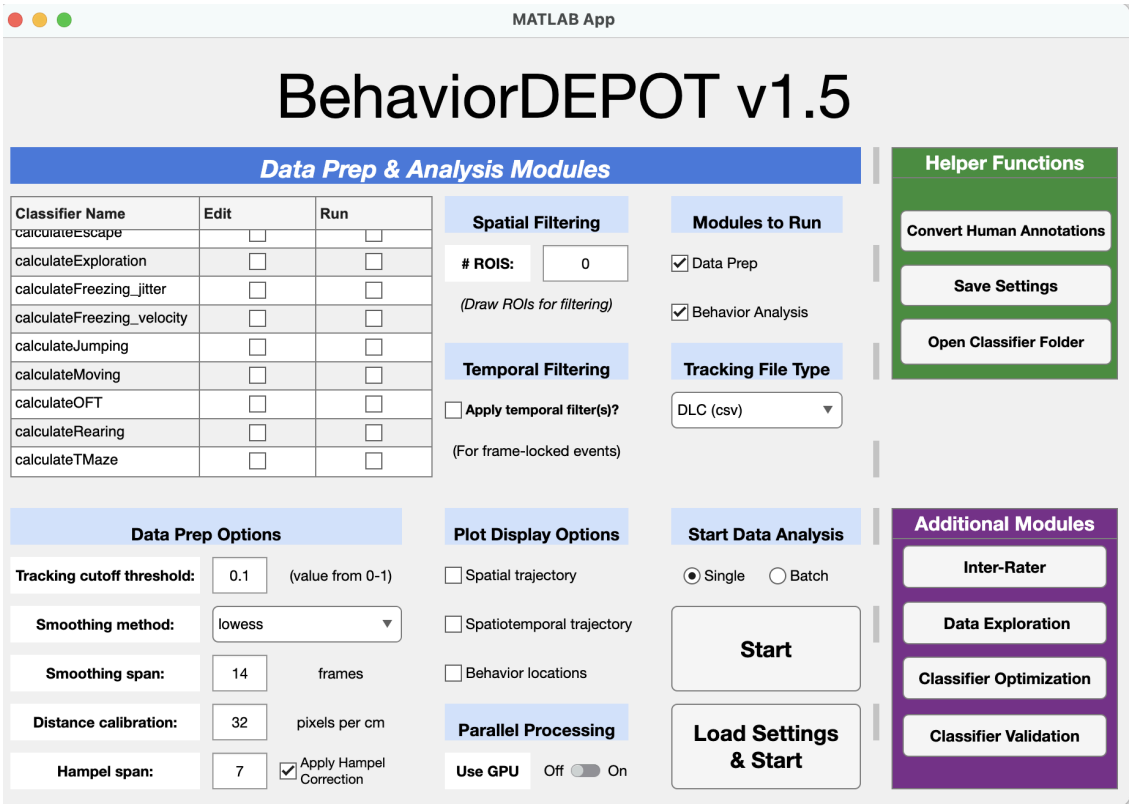
a. conda activate deeplabcut

3. Activate the Graphic User Interface (GUI)
    a. python -m deeplabcut

4. New Project
    a. Select videos from H file
    b. Click extract frames tab
    c. Select KMeans clustering (anxious vs exploratory behavior)
    d. *7/27/23left other parameters at default* then click extract frames (progress can be seen in anaconda terminal)
    e. Click label frames and select folder with new iteration file and naprari (basic functions video here) should open: **Open File Functions Crashes Nampari atm**

Behavior DEPOT

Upon generating pose estimation data with DeepLabCut, you can leverage BehaviorDEPOT, a specialized platform for animal behavior analysis, to conduct further examination of your Elevated Zero Maze (EZM) experiment recordings. BehaviorDEPOT is capable of distinguishing freezing versus exploratory behavior in the open arms of the EZM, differentiating zones, and calculating movement, given its built-in support for EZM. However, it is crucial to figure out an effective method for assigning zones in DeepLabCut, as the utility of the secondary analysis in BehaviorDEPOT heavily depends on this. Hence, accurate zone assignment is a prerequisite for the successful calculation of metrics such as time spent in open versus closed arms and total distance traveled.



BehaviorDEPOT is not a Python package, but rather a MATLAB application.

Is installed, needs MATLAB Toolkits:

1. Open MATLAB.

2. In the MATLAB toolstrip (the toolbar at the top of the MATLAB window), locate and click the "Add-Ons" button, which should be on the right side of the Home tab. This will open the Add-On Explorer.

3. In the Add-On Explorer, you can search for and install the required toolboxes. The toolboxes needed for Behavior DEPOT are:

    a. Signal Processing Toolbox
    b. Curve Fitting Toolbox
    c. Image Processing Toolbox
    d. Statistics and Machine Learning Toolbox

4. You can find these toolboxes by typing their names in the search bar at the top of the Add-On Explorer. Click on the desired toolbox in the search results to view its details.

5. To install a toolbox, click the "Install" button on the toolbox details page. You may need to sign in with your MathWorks Account if you haven't done so already. Ensure that you have a valid license for the toolbox you're trying to install.

6. Once the installation is complete, the new toolbox will be added to your MATLAB environment, and you can start using its functions. To check if the toolboxes have been installed successfully, type ver in the MATLAB command window. The list of installed toolboxes should now include the ones you just installed.

EXPLORE Software for Novel Object Recognition Video Processing

Into:
EXPLORE is an open-source software designed to process and analyze videos for novel object recognition testing. It uses machine learning algorithms to identify and track objects in video recordings, providing valuable data for behavioral studies.

EXPLORE is hosted on GitHub and can be accessed via the following steps:

1. Navigate to the EXPLORE GitHub repository.
2. Click on the green "Code" button and copy the URL.
3. Open a terminal on your computer.
4. Navigate to the directory where you want to download EXPLORE.
5. Type `git clone` followed by the URL you copied and press Enter.

Using EXPLORE:

1. Once you have cloned the EXPLORE repository, navigate to the directory containing the software.
2. Run `main_training.py` to train the model. This will generate a `.h5` file which contains the trained model.
3. Run `main_prediction.py` to process your videos. This will use the trained model to identify and track objects in the videos

Current Issues:

1. **Multiple Objects in Trials:** EXPLORE currently does not accommodate multiple frame display during object cropping. When using two objects at a time and more than two different objects across a series of related trials, the recordings have to be separated by present objects rather than variables of interest.
2. **Older Package Dependencies:** EXPLORE relies on older packages that are no longer supported on current Mac systems, specifically Python 3.6. This makes it difficult to install and use the software on newer systems, especially those with the M1 chip.

3. **Issues with main_training.py:** It is possible to use `main_training.py` with some minor edits while using Python 3.8 or newer, but progressing further is currently challenging due to the aforementioned compatibility issues.

EXPLORE is currently crashing during main prediction. Why?:

Notes

General: try to time the recording sessions within their wakefulness period. I wonder if we could make the Object recognition test darker for the mice to try and get more activity/less freezing?

Order of tests: It's recommended to start with the least invasive tests (e.g., open field test, novel object test) and gradually move on to more demanding tests (e.g., rotarod test, beam walk test). This helps minimize any carryover effects from previous tests.
This stands

Can behviorDEPOT be used to further analyze EXPLORE outputs?

Do I have to retrain networks as animals grow? Who has addressed this Q?

Can I move all of these applications to a google Colab, and convert the instructions such that others could identify their tests of interest and plug and play without needing local plugins?

NO not effectiv. Using a VM may not be better then amending code, packages, and providing feedback to the groups making the software. Running it on the lab PC may still be best voice even via remote desktop

Nature review paper that got recently published explaining why its useful to use more “natural” tests in rodents to understand and interpret their behavior rather than traditional switches, mazes or swims: whats relevant to us?
We using what everyone else using

Open field test:
Effect size for distance traveled or time spent in the center zone: small to medium (Cohen's d = 0.2 to 0.5)
Novel object test:
Effect size for discrimination index (DI) or exploration time ratio: small to medium (Cohen's d = 0.2 to 0.5)

Rotarod test:
Effect size for latency to fall or number of falls: medium to large (Cohen's d = 0.5 to 0.8)
Beam walk test:
Effect size for latency to traverse the beam or number of slips: medium to large (Cohen's d = 0.5 to 0.8)
Sucrose preference test:
Effect size for sucrose preference ratio or consumption volume: small to medium (Cohen's d = 0.2 to 0.5)