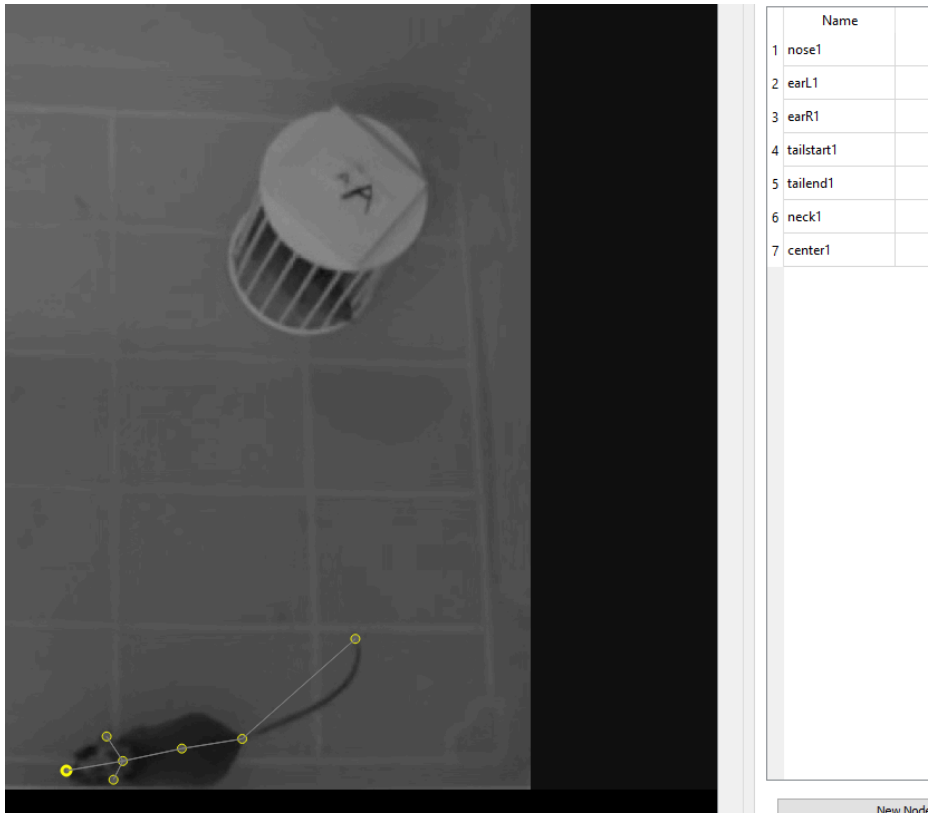
SimBA Protocol

- **Summary**
  - See: SLEAP Protocol ;
  - GitHub - sgoldenlab/simba: SimBA (Simple Behavioral Analysis), a pipeline and GUI for developing supervised behavioral classifiers
  - simba/docs/tutorial.md at master
  - https://app.gitter.im/#/room/#SimBA-Resource_community:gitter.im
  - Download Windows 10 Media Feature Pack (microsoft.com)
  - Download Visual C++ Redistributable for Visual Studio 2015 from Official Microsoft Download Center
- **Installation**
  - When installing simba into a new conda environment, you MUST FOLLOW THIS!!!
  - Install new stable version
    - pip install simba-uw-tf==1.3.12
  - Install new dev version, specific or latest version. First, activate the conda environment.
    - Conda activate simba
    - pip install simba-uw-tf-dev --upgrade
    - pip install simba-uw-tf-dev==0.80.5
  - Simba
  - Cd /d C:/
    - git clone -b master https://github.com/sgoldenlab/simba.git
    - pip3 install -r simba/SimBA/requirements.txt
      - This didn't work for me. Try…
      - Python -m pip install -r simba/SimBA/requirements.txt
  - Alternative install
    - Conda create simba_env
    - Activate conda simba_env
    - Install all the dependencies listed on the website: simba/installation.md at master · sgoldenlab/simba (github.com)
  - simba/FAQ.md at master · sgoldenlab/simba (github.com)
- **Running/activating simba**
  - Check location of simba conda installation by opening different cmd window options and running: conda info –envs
    - Could be in standard Anaconda cmd, or in anaconda miniconda cmd
  - Run the following to open the GUI
    - Conda activate simba
    - Simba
  - Note: Simba operates differently than sleap. Instead of working on the pre-existing videos and tracked files, simba makes copies of all videos and tracked files in its own directory (project_folder), and then does all analysis within that folder. This effectively doubles the size of your working directory, but so be it.
- **Creating a new project for sleap tracked videos**

- Generally speaking, if you have a large number of videos in your project, you should split your data into separate folders based on the **filming environment,** i.e. that all your videos should be shot by the same camera at the same position (or at least that they are so similar that they are essentially indistinguishable, especially in terms of the height of the camera from the subject). This allows you later on to create a uniform pixel/mm value for all videos, instead of having to manually set it for each video.
- Create custom pose config based on still image of skeleton and with exact same naming convention in the sleap project folder. You can open the sleap skeleton.json with notepad to get node names.
    - Name of the body part config should reflect the parent directory ideally.
- Grab the skeleton.jpg from the sleap folder (should be a screenshot that shows the animal with all body points labeled and window displaying the skeleton layout and ordering. Note that in simba, the ordering of the body points must match the ordering of the sleap data.
    - 

- Create project
    - Select skeleton you created
- Project name: simba
    - This should be in the folder where videos/sleap prediction files are stored, as well as the “sleap” folder. This will create a new folder called “simba”. If you have to restart the project for whatever reason, just delete the simba folder and start over.

- **Import videos**

    - ***Important: Simba fundamentally breaks when animals are not visible in the first frames of the video. The devs strongly recommend pre-processing your videos prior to sleap tracking AND simba to trim the videos to just where the mouse is present in the first frames of the video***
- **Import tracking data**
    - first, you must ensure that your tracking.h5 file and corresponding video file have the exact same name to import the data into simba. Use my batch_sleap_rename.py script to remove any prefix or appendix that sleap may have added to the .h5 file name
    - Settings
        - Data type = H5 (sleap)
        - Interpolation method: None
        - Smoothing: none
            - Visual of what the interpolation steps do:
                - https://github.com/sgoldenlab/simba/blob/master/images/Interpolation_comparison.png
            - **Critical:** if you choose an interpolation method, it assumes there MUST be an animal in frame at all times in the video, and will thus create rows of data, even if there are none. Importantly, a good model in sleap will CORRECTLY not show a skeleton instance in frames where there are none at the outset of a video.
            - When set to none, the created csv will not try to fill in the empty frames of the video at the beginning of the video. This is critical.
            - If you choose none, you can still run my interpolation script on these videos for interpolation on frames exported from sleap as having instances in the video frame.
    - Imported csv will be placed into the project_folder > csv > input_csv folder
        - Notably, these imported csvs will have probability estimates of 1 for every row.
        - If you want to start over, you can go into this folder and delete the csvs and re-import with new settings
- **Interpolate input_csv .csv files using custom script**
    - Run simba_input_csv_interpolation.R script in the input_csv folder to convert the csvs. Check the files.
    - Files will be saved to the “proc” folder. Copy these csvs into the
- **Visualize the imported videos with tracking data (optional)**
    - Optional: You can visualize the import_csv results (and whatever interpolation/smoothing methods you may have deployed on import).
        - Tools > Visualize pose-estimation in folder
            - Set input directory to input_csv folder
            - Set output to simba > project_folder > frames > output folder
            - **DO NOT put in the directory where your videos are stored, it will overwrite them.**
- **Once video and tracking data (optionally interpolated) are imported, load the project**

- **Video parameters**
    - Calculate mm/px using the simba internal tool. Output will be printed to the main simba program window. Copy that information and use it in the video parameters box.
- **Outlier correction (option to skip)**
    - ***If you copy interpolated input_csv files into outlier_movement_corrected folder, skip this tab entirely***
    - Skip outlier correction
        - If you skip the outlier correction, copies of the the input_csv files will be moved to the ouliter_corrected_movement_location.
    - Run outlier correction: Settings
        - Nose + tail start, criterion = 2
- **ROI analysis and visualization**
    - Set shape thickness, name different ROIs, etc.
    - Analyze nose first > visualize ROI tracking of nose (defaults to last analyzed body part) > analyze center > visualize ROI tracking of center
    - Create tracked videos
        - Use multi-threading, at least 2 cpu cores.
            - Doing so results in an mp4 video that is smaller size (but of equivalent quality). Without multi-threading you get a larger avi file.
        - Tracked videos stored in project_folder > frames > output > ROI_analysis
    - Note: Using the probability threshold will basically hide the data from the ROI analysis/pause it. In the output video, if a keypoint falls below the threshold, the time and entry counter will disappear, indicating the data is not being counted (i.e. time doesnt increase, nor does the entries). Importantly, the disappearing dot will NOT result in artificial registered “entries”.
    - Note: If there is no ROI in a video, the video will not be included in ANY analysis, including the distance traveled/velocity analysis. Therefore, you should always have at least one ROI per video (i.e. “arena”)
- **Extract Features**
- **Label Behavior**
- **Train Machine Model**
- **Run Machine Model**
- **Visualizations**
- **Add ons**

Issues and Errors

- **2023.8.18, 2022_LID_mong**
    - Liddell bucket size: long side, 40cm.
    - Ephys rigs: 40cm.

    - I notice that importing some sleap .h5 files, some body points wont be imported properly. I imported some csvs, and despite there being clear center1 tracking info in sleap, it shows up as xy coordinates 0,0 for all rows. Unclear why this is happening.
    - Skipping the outlier correction will reset the starting frames of the csvs back to 0, even if they were imported without interpolation.
    - Coping the input_csvs into the outlier_corrected_movement_location still results in ROI tracked videos where the tracked point appears at the 0 frame, even if the csv first row is a higher frame number.
        - All of this suggests that the animal really does need to appear in the first frame of the video. This is confirmed in the following steps of the documentation
            - https://github.com/sgoldenlab/simba/blob/master/docs/Scenario1.md#to-import-multiple-dlc-csv-files
    - Basically, I think that I dont want to use simba's native interpolation steps…. I need to do a custom approach. Use simba to import the h5s into csvs with NO interpolation and NO smoothing. Then I need to write a custom R function to fill in the missing frames with some kind of NA or NaN values.
        - I successfully created this script, huzzah.
- **Old issues**
    - ML Rig 2 throws cv2 DLL error after recent windows update.
        - Windows key + I to go into Settings, then navigate to Apps > Manage optional features, then click Add a feature. Provided you see Windows Media Player, click that, then click Install.
        - Also do media Feature Pack
        - Downloading them directly from the websites failed for me.
    - This fixed my issue. I posted on the SIMBA gitter sharing this jewel of knowledge.
    - **Converting sleap .h5 files to csvs with simba and interpolation**
        - When running interpolation on videos where animals briefly disappear from view, you should NOT run the outlier correction step because it erases the interpolated frames.
    - Simba was working fine, but then deeplabcut would not work
        - Threw module not found error
    - Uninstalled deeplabcut, and reinstalled using downloaded yaml file. Had to DELETE the previous dlc-windowsGPU conda environment
    - Now, deeplabcut works fine
    - But attempting to run Simba.py throws an error (module not found error,
        - ModuleNotFoundError: No module named 'yellowbrick'
    - I remember this error from before!!! But I did not write down how I fixed it, so fuck me. Had a vague recollection that I needed to install different version of DLC?
    - So Now I am deleting my previous C:/DeepLabCut/Simba folder
    - … and reinstalling via
        - git clone -b master https://github.com/sgoldenlab/simba.git
        - pip install -r simba/simba/requirements.txt
    - And Simba launches just fine. And once again, DEEPLABCUT DOESN'T WORK!!!
    - So it appears Simba is somehow affecting the way DLC opens

- Deleted simba folder
    - Deleted dlc-windowsGPU environment
    - Reactivated dlc-windowsGPU.yaml folder in C:/DLC
    - C:\SIMBA>pip install -r simba/simba/requirements.txt
        - Yields Command "python setup.py egg_info" failed with error code 1 in C:\Users\CALEBV~1\AppData\Local\Temp\pip-install-48perxz1\ruamel.yaml\
            - This is familiar. And as I recall, I had to fix this by activating the dlc environment
- ## OFT CHAMBER: 17.5 x 17.5" (444.5 x 444.5mm) / 18" x 18" (457.2 x 457.2mm)
- SPP CHAMBER: 23 x 8.5" (584 x 216mm)