SimBA-uw-tf

Release March 2020

Simon Nilsson, Jia Jie Choong, Sam Golden

Oct 21, 2021

CONTENTS:

This page hosts the documentation for SimBA-uw-tf

CHAPTER

ONE

INTRODUCTION

1.1 What is SimBA?

Several excellent computational frameworks exist that enable high-throughput and consistent tracking of freely moving unmarked animals. Here we introduce and distribute a plug-and play pipeline that enabled users to use these pose-estimation approaches in combination with behavioral annotation and generation of supervised machine-learning behavioral predictive classifiers. We have developed this pipeline for the analysis of complex social behaviors, but have included the flexibility for users to generate predictive classifiers across other behavioral modalities with minimal effort and no specialized computational background.

SimBA does not require computer science and programing experience, and SimBA is optimized for wide-ranging video acquisition parameters and quality. SimBA is written for Microsoft Windows. We may be able to provide support and advice for specific use instances, especially if it benefits multiple users and advances the scope of SimBA. Feel free to post issues and bugs here or contact us directly and we'll work on squashing them as they appear. We hope that users will contribute to the community!

- The SimBA pipeline requires no programing knowledge
- Specialized commercial or custom-made equipment is not required
- Extensive annotations are not required
- The pipeline is flexible and can be used to create and validate classifiers for different behaviors and environments
- Currently included behavioral classifiers have been validated in mice and rats
- SimBA is written for Windows

SimBA provides several validated classifer libraries using videos filmed from above at 90° angle with pose-estimation data from 8 body parts per animal; please see our OSF repository for access to all files. SimBA now accepts any user-defined pose-estimation annotation schemes with the inclusion of the Flexible Annotation Module in v1.1. SimBA now supports maDLC and SLEAP for similar looking animals with the release of maDLC/SLEAP module in v1.2.

Installation note: SimBA can be installed either with TensorFlow compatability (for generating DeepLabCut, DeepPoseKit and SLEAP pose-estimation models), or without TensorFlow (for stand-alone use with classifiers and other functions). Please choose the appropriate branch for your needs, using pip install. More details are found in the Installation Documentation.

Listserv for release information: If you would like to receive notification for new releases of SimBA, please fill out this form and you will be added to the listserv.

1.2 Need help

Please join our gitter community for help and questions.

https://gitter.im/SimBA-Resource/community

1.3 Installation

If you are having trouble installing SimBA, please reach out to us on gitter.

1.3.1 Requirements

1. Python 3.6.8
2. Git
3. FFmpeg
4. Windows OS

1.3.2 Installing on windows machine

1. Open up command prompt and type, `pip install simba-uw-tf-dev`

1.3.3 Install SimBA using anaconda (does not support dlc)

This will install the latest development version of SimBA.

1. Open up command prompt.
2. Make a new environment with python 3.6, `conda create -n simbaenv python=3.6.10`
3. Make sure you activate your environment, `conda activate simbaenv`
4. In the terminal type `pip install simba-uw-tf-dev`
5. It will error out because of shapely. Uninstall shapely by `pip uninstall shapely`
6. Then reinstall shapely with conda command: `conda install -c conda-forge shapely`

1.3.4 Install SimBA with built in DeepLabCut (anaconda)

This will install SimBA with built in deeplabcut.

> **Warning:** We are are no longer supporting the legacy SimBAxTF branch, which required TensorFlow. To receive support, you must download the most up-to-date SimBA version (pip install simba-uw-tf-dev). If you are using the legacy SimBAxTF branch, please transition to the currently supported branch. If you require access to download the legacy branch, please contact us on Gitter for access.

1. Open up command prompt.
2. Make a new environment with python 3.6, `conda create -n simbaenv python=3.6.10`

3. Make sure you activate your environment, `conda activate simbaenv`
4. Downgrade pip to version 19.0.1 by `pip install pip==19.0.1`.
5. Install simba tensorflow version, `pip install simba-uw-tf`
6. It will error out because of shapely. Uninstall shapely by `pip uninstall shapely`
7. Then reinstall shapely with conda command: `conda install -c conda-forge shapely`

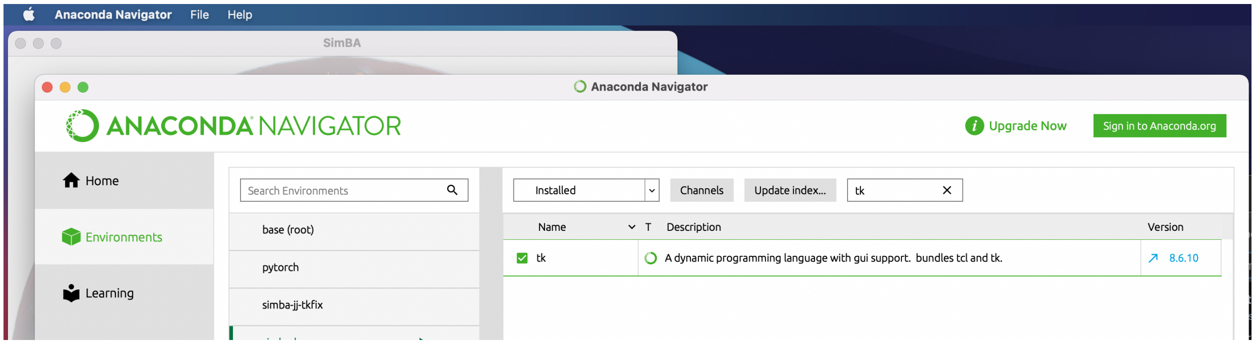
1.3.5 Installing on MacOS

Not recommended but it is possible.

Requirements

- XCode installed
- Homebrew
- ffmpeg
- Python 3.6.13
- Anaconda

Please make sure that in your environment, the version of `tk` is `8.6.10`



base (root)
pytorch
simba-jj-tkfix

	Name	T	Description	Version
☑	tk	○	A dynamic programming language with gui support. bundles tcl and tk.	8.6.10

Installation process

1. Create an environment for simba using anaconda terminal.
2. In the terminal type, `pip install simba-uw-tf-dev`
3. Then, `conda install -c anaconda python.app`
4. Then, `conda install matplotlib`
5. Then, `conda uninstall shapely`
6. Then, `conda install -c conda-forge shapely`
7. Then, `pip install shap`
8. Lastly, `pip install h5py`
9. In the terminal, type in `simba` to test if it works.

1.4 License

1.5 Contact

Questions? Please contact goldenneurolab@gmail.com

CHAPTER

TWO

GETTING STARTED

2.1 Step 1: Processing videos in batch

It is often helpful, and sometimes necessary, to pre-process experimental videos prior to analysis. This is generally accomplished through the use of open-source approaches like **FFmpeg** or commercial software, but can be a time consuming and cumbersome when applying to numerous similar videos. To streamline this process, SimBA incorporates **FFmpeg** into a a batch-process GUI.

This video pre-processing tool allows you to change multiple video parameters (clip/trim/crop, etc.) for multiple videos in a batch-process that is rapidly configured and then executed when ready. Once the required parameters has been set for all the videos, the user press *Execute* and the new videos will be generated according to the user input. Videos are processed using **FFmpeg**. Click here to learn how to install FFmpeg on your computer.

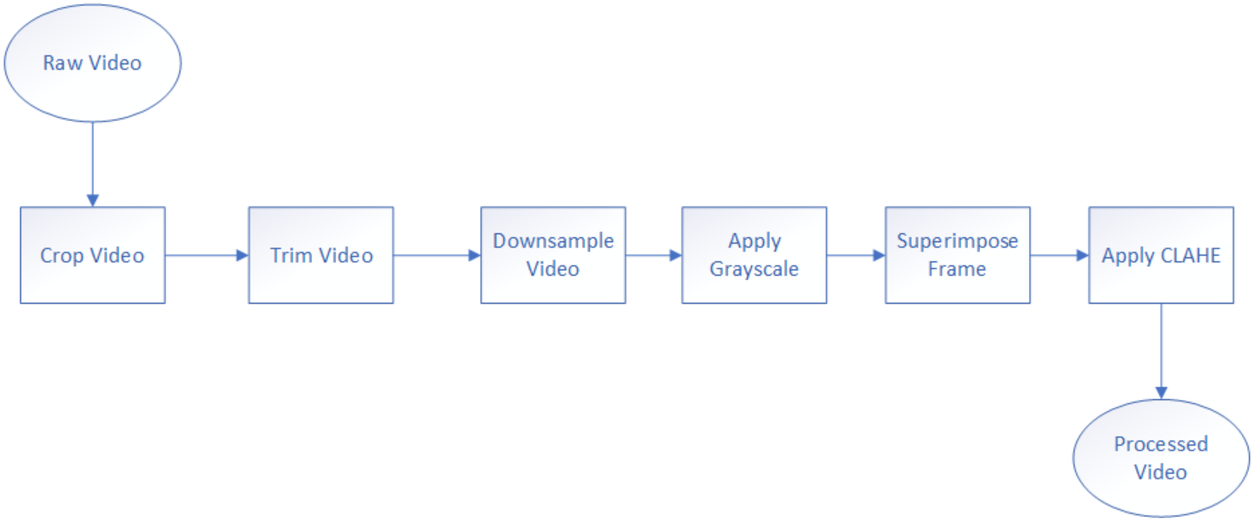
Note: Processing numerous high-resolution or long-duration videos takes time. We strongly suggest that you not execute a batch until you are ready to commit computational time to the process.

We suggest pre-processing videos in the following scenarios:

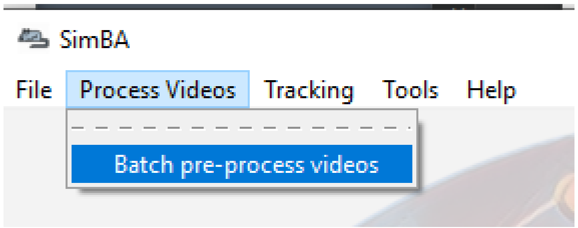
1) Red-light: If you have recorded in red-light conditions, we suggest converting to gray-scale and using CLAHE.
2) Reflections: If your recordings include reflections on the walls of your testing apparatus, we suggest cropping to remove the reflections.
3) Artifacts: If your recordings include frames where your hands (or other unintended features) are included in the video, at either the start or the end, we suggest trimming the videos to remove these frames.
4) Resolution: If your recordings are significantly higher resolution than needed, we suggest down-sampling to decrease processing time.

2.1.1 Pipeline

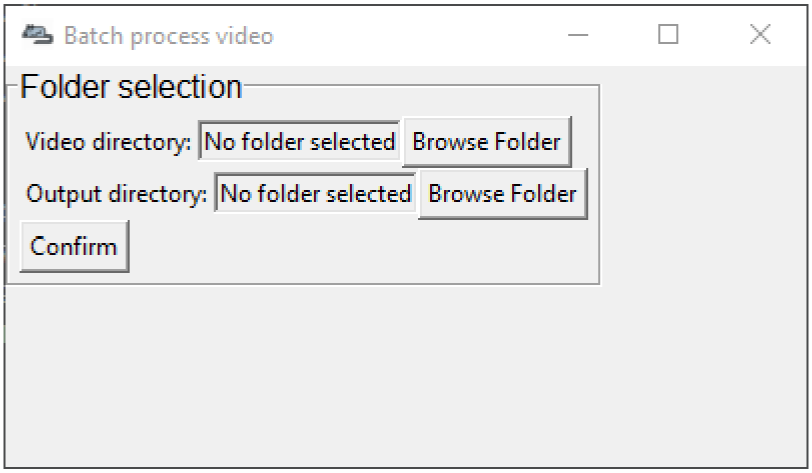
The video parameters that you specify will be processed in the following sequence. If the user leaves certain parameters unchanged, then they are ignored in the pipeline.



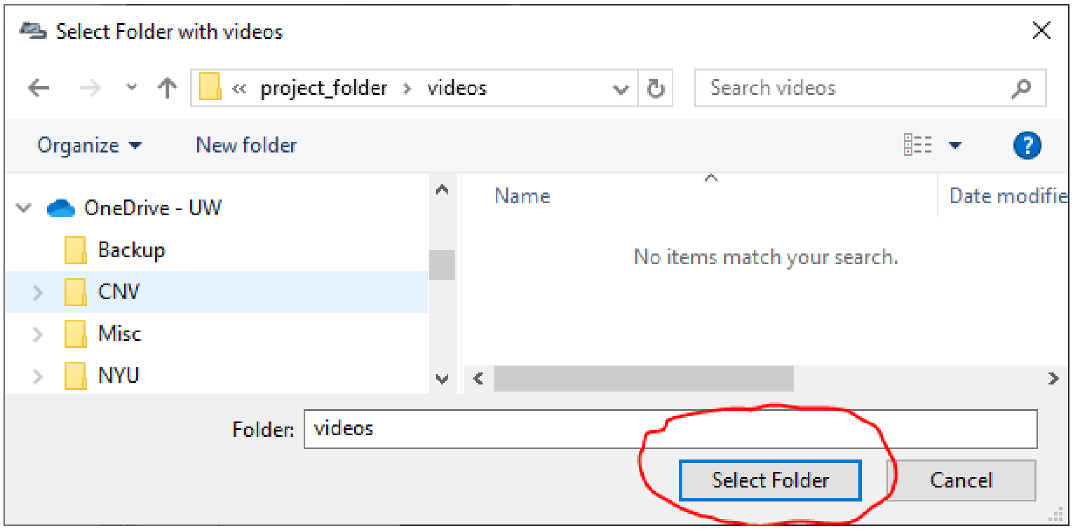
2.1.2 Step 1: Folder Selection



1. To begin batch pre-processing, in the main SimBA window click on `Process Videos` –> `Batch pre-process videos`. The window shown below will display.



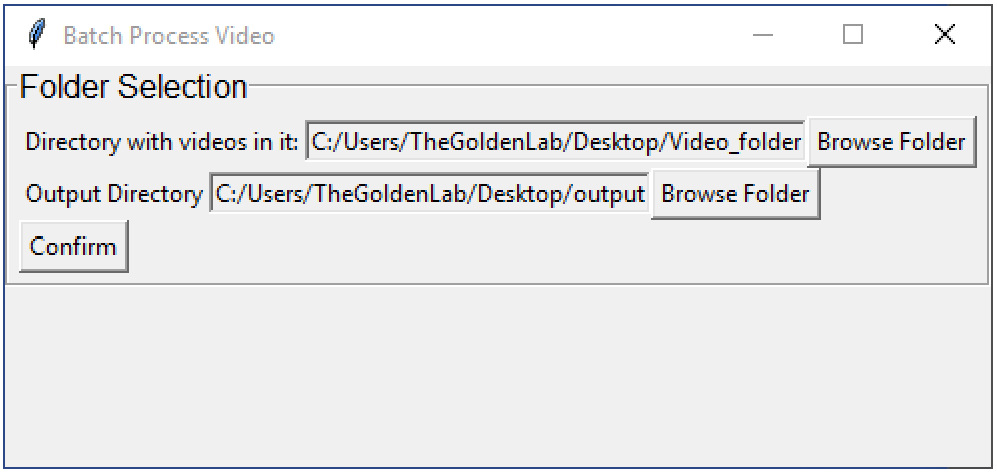
2. Under **Folder Selection** heading and next to `Video directory`, click on `Browse Folder` and navigate to a folder that contains the videos that should be batch processed and click on `Select Folder`. All vidoes that you would like to process must be present in this directory.



Folder: videos

3. Next to `Output Directory`, click on `Browse Folder` and navigate to a folder *(usually a new, empty, folder)* that should store the processed videos and click on `Select Folder`.
4. Click to `Confirm` the two selected directories.

Note: Please make sure there is no spaces in your folder names or video names, use underscores if needed.

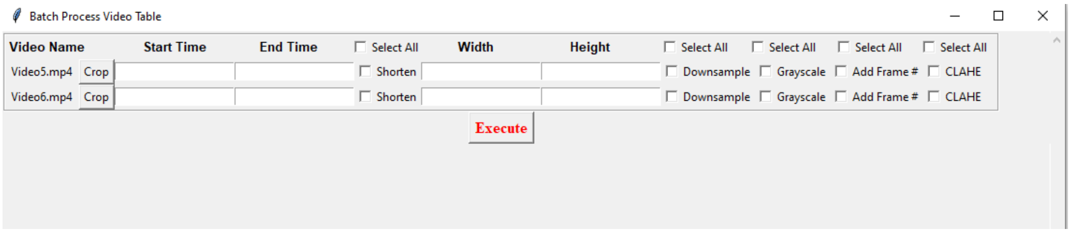


Directory with videos in it: C:/Users/TheGoldenLab/Desktop/Video_folder

Output Directory C:/Users/TheGoldenLab/Desktop/output

2.1.3 Step 2: Select parameters

1. Once you select `Confirm`, a table listing the videos in the `Video directory` folder will auto-generate and display. This sequence of videos will now be processed with the parameters you selected, started with the cropping function.



Video Name		Start Time	End Time	□ Select All	Width	Height	□ Select All	□ Select All	□ Select All	□ Select All
Video5.mp4	Crop			□ Shorten			□ Downsample	□ Grayscale	□ Add Frame #	□ CLAHE
Video6.mp4	Crop			□ Shorten			□ Downsample	□ Grayscale	□ Add Frame #	□ CLAHE

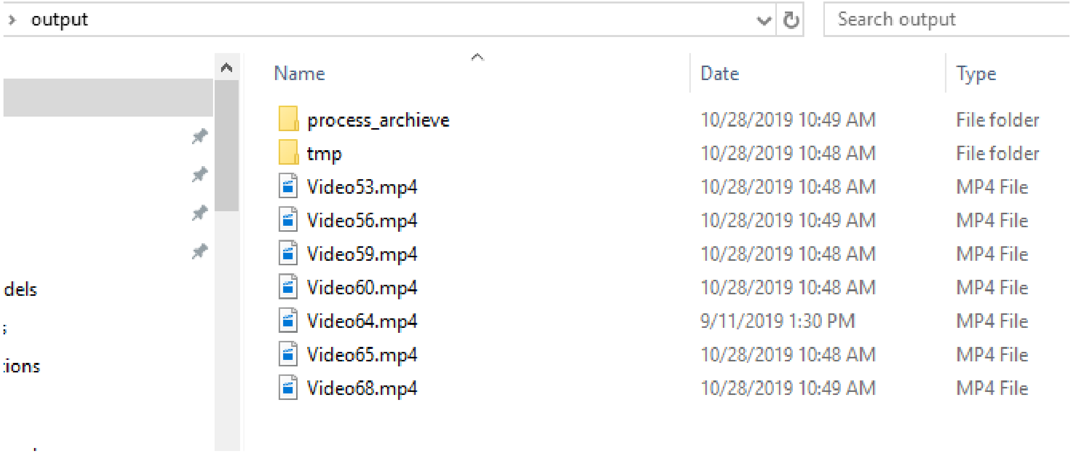
2. If you wish to crop your videos, click on the `Crop` button. A frame from the video will pop up. Left click on the frame and drag the rectangular bounding box to mark the region of the video you wish to keep. Once the rectangle is marked, double tap "Enter" on your keyboard. *(The relevant crop button will turn red once you have selected to crop a video)*

3. If you wish to trim a specific video, check the `Shorten` box and enter the **Start Time** and **End Time** in the HH:MM:SS format.
4. If you wish to downsample a specific video, check the `Downsample` box and enter the **Width** and **Height** in pixels.
5. If you wish to change a specific video to grayscale, check the `Grayscale` box.
6. If you wish to superimpose the specific frame numbers onto the frames of the video, check on the `Add Frame #` box. For an example output video with frame numbers overlaid, click here
7. If you wish to apply CLAHE, check on the `CLAHE` box. For more information on CLAHE, click here
8. In the first row of the table, there are `Select All` checkboxes. Use these checkboxes to apply a manipulation to all of the videos in the folder.

Note: We know that the *Select All* checkbox might be slightly off position in the table. We are working on a fix.

2.1.4 Step 3: Execute

1. Once all the parameters are set, click on `Execute`.
2. The final output videos will be saved in the `Output Directory` that you selected in *Step 1*.



Name	Date	Type
process_archieve	10/28/2019 10:49 AM	File folder
tmp	10/28/2019 10:48 AM	File folder
Video53.mp4	10/28/2019 10:48 AM	MP4 File
Video56.mp4	10/28/2019 10:49 AM	MP4 File
Video59.mp4	10/28/2019 10:48 AM	MP4 File
Video60.mp4	10/28/2019 10:48 AM	MP4 File
Video64.mp4	9/11/2019 1:30 PM	MP4 File
Video65.mp4	10/28/2019 10:48 AM	MP4 File
Video68.mp4	10/28/2019 10:49 AM	MP4 File

3. A subfolder in the `Output Directory` called **tmp** will contain the step-by-step processsed videos.
4. The **process_archieve** folder contains a **.txt** file that lists the processes that were run.
5. The **Output Directory** will contain all the final processed videos.

NEXT STEP: Create tracking model

2.2 Step 2: Create Tracking Data

Using SimBA to create/edit DeepLabCut

> **Warning: IF YOU ALREADY HAVE GENERATED A DLC TRACKING MODEL AND CREATED .CSV DATAFRAMES FOR YOUR EXPERIMENTAL VIDEOS (USING THE ANNOTATION CONFIGURATION DETAILED IN THE FIGURE BELOW), PLEASE SKIP THIS ENTIRE PAGE AND MOVE ON TO STEP 3: BUILDING CLASSIFIERS**

See **"Pose estimation body-part labelling"** below for supported rodent annotation formats. You **must use** one of these formats if you would like to use our included feature list for creating predictive classifiers. We **strongly** suggest using 8 body parts per rodent.

Again, if you have generated your DLC tracking using another body part labelling configuration, it will not work "right out of the box"!

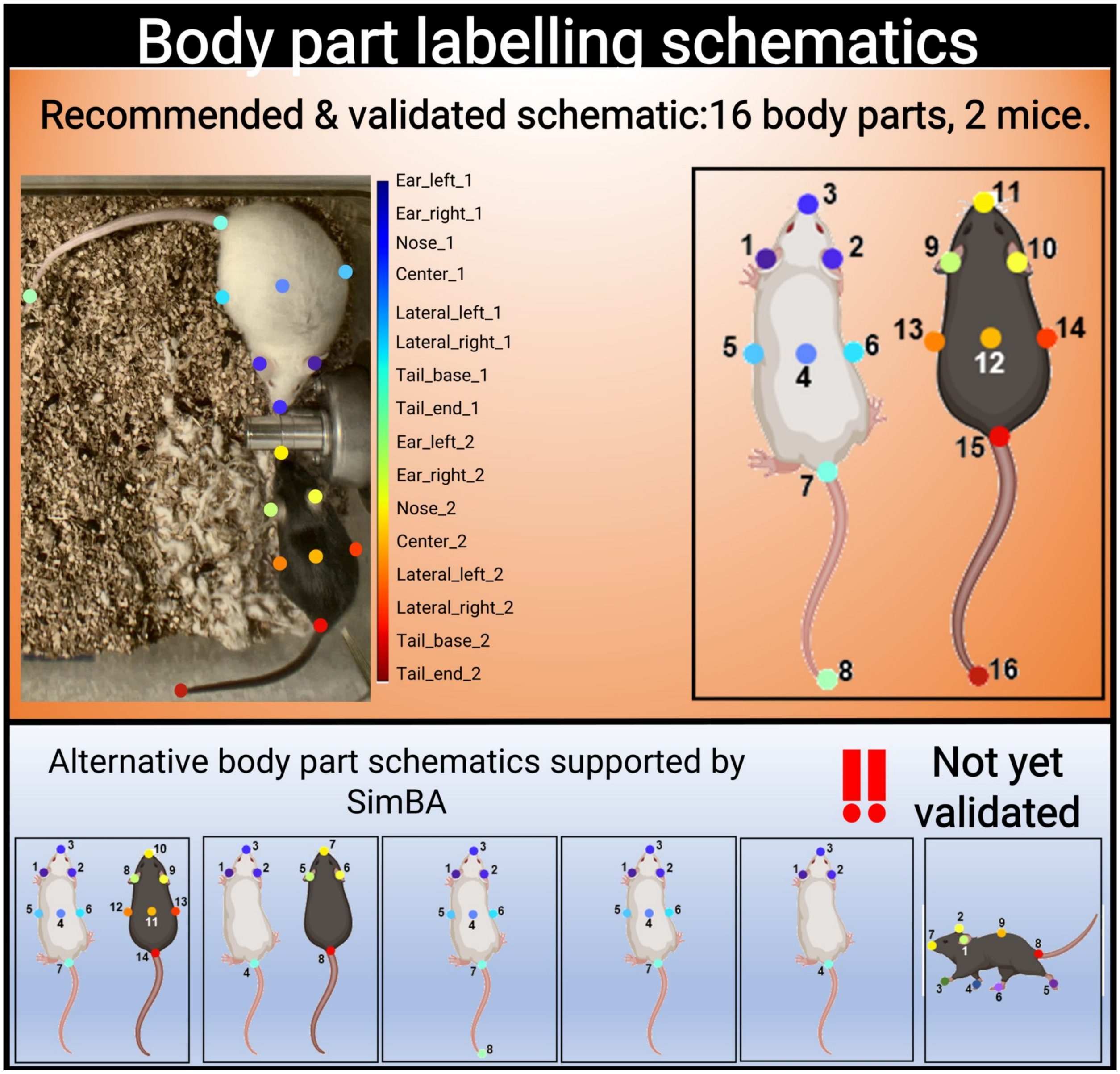
Note: Some versions of DeepLabCut saves data in *hdf5* format by default. For information on how to output DeepLabCut data in csv file format, go to the DeepLabcut repository

DeepLabCut is currently fully supported by the SimBA pipeline. We **strongly** suggest that users generate their DLC tracking model using the DLC GUI. This will provide the most up-to-date and pleasurable annotation experience, followed by loading the generated DLC tracking data and raw videos into SimBA. Future iterations will include tracking support for mask R-CNN, LEAP, DeepPoseKit, etc, as requested by the community.

IF YOU STILL WOULD LIKE TO GENERATE YOUR DLC TRACKING MODEL AND .CSV DATAFRAMES VIA SIMBA, CONTINUE…

Note: This section of SimBA was written early on - when DeepLabCut was accessed exclusively by the command line and did not come with the excellent graphical user interface it comes with today. At that early point we added DeepLabCut commands to SimBA to make many of the important DeepLabCut functions accessible for all members of the lab, regardless of previous programming experience. We have left these features in the current build, as an option for use.

2.2.1 Pose estimation body-part labelling



SimBA uses the tracking data to construct a battery of *features* (for a list of features with example descriptions, click here. The number of labelled body parts, and which body parts are labelled and tracked, determines which features can be calculated.

We have validated SimBA by labelling 16 body parts (8 body parts on two individual mice) and we **strongly** advice users to use the same body part labelling schematic.

Please note that the **order** of body part annotation is important, and matches the labelling schematic above.

If using SimBA, set the DeepLabCut config to use the 16 body part schematic by checking the `Apply Golden Aggression Config` box (See **Step 1**, below). Pre-trained DLC tracking models using the 16 body part schematic and two animals can also be downloaded through OSF, covering a range of acquisition parameters (color video, blank and white video, etc).

However, SimBA will also accept alternative body labelling schematics (see the above figure). These alternative schematics currently include:

- 7 body parts on two individual mice recorded 90° above angle
- 4 body parts on two individual mice recorded 90° above angle
- 8 body parts on a single mouse recorded 90° above angle
- 7 body parts on a single mouse recorded 90° above angle
- 4 body parts on a single mouse recorded 90° above angle
- 9 body parts on a single mouse recorded 45° side angle

Note: So far we have only validated machine learning models that use features calculated from 16 labelled body parts on two individual animals. The other configurations are "use at your own risk" and may require trouble-shooting.

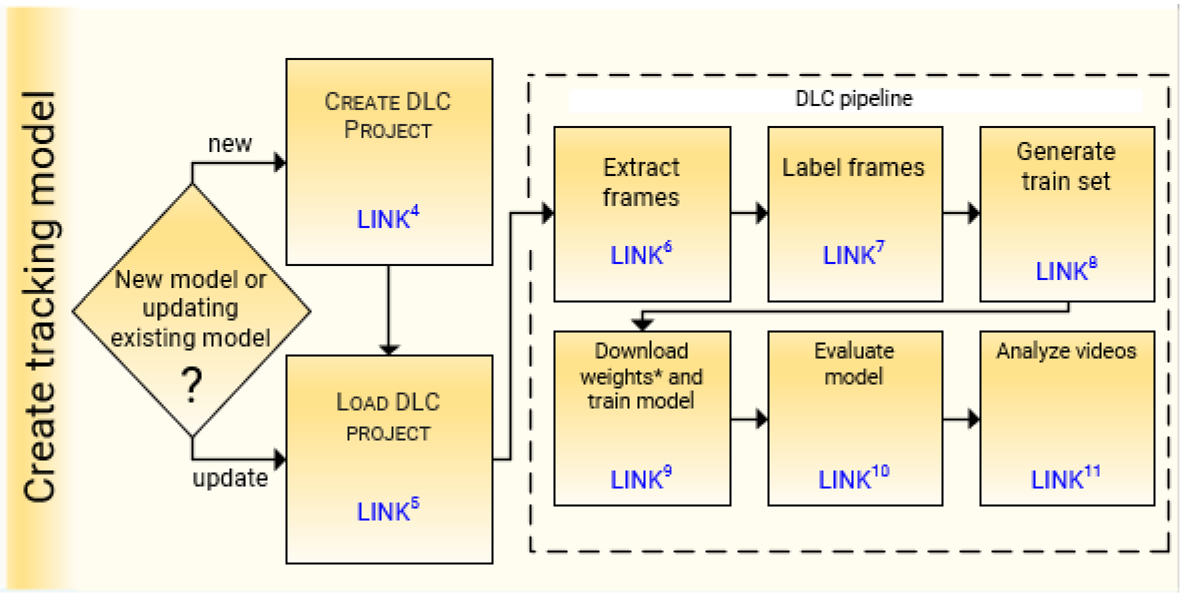
Note: SimBA now also supports user-created body-part annotation schemes. These annotations schemes can be created in the SimBA project configuration menu - click HERE to learn more.

2.2.2 SimBA-DLC Pipeline breakdown

For detailed information on the DeepLabCut workflow, see the DeepLabCut repository

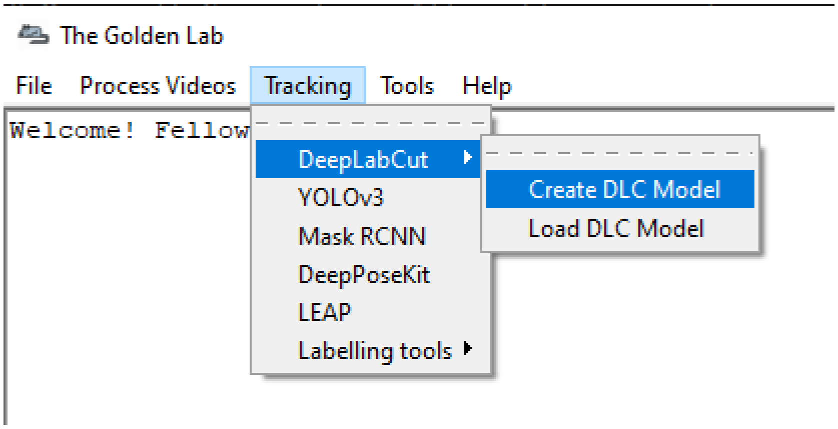
The SimBA-DLC pipeline is confirmed compatible with DLC v2.0.9.

Note: All videos should have been pre-processed, see Step 1: Video pre-processing, prior to DLC tracking.



2.2.3 Part 1: Create DLC Model

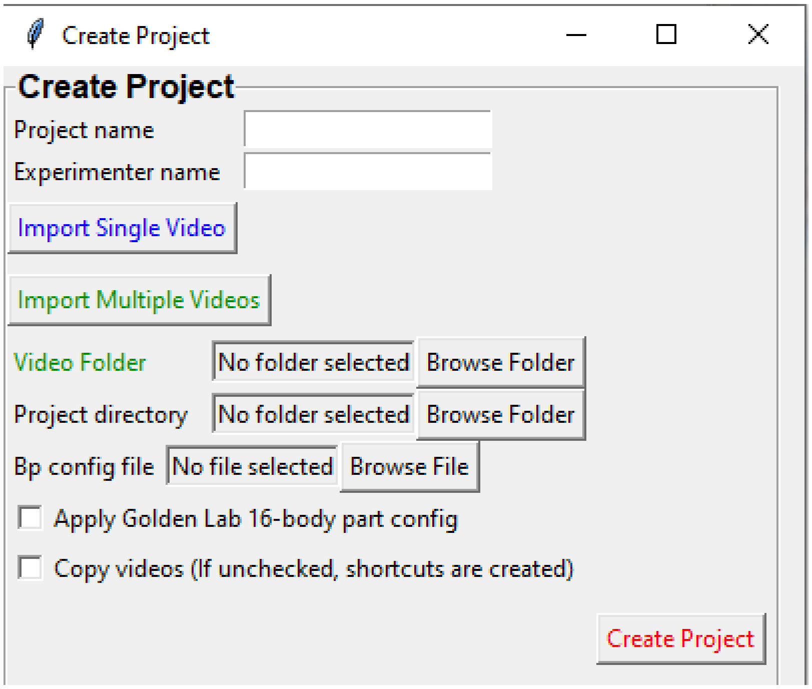
Use this menu to create a new project for your DeepLabCut pose-estimation tracking analysis.



Step 1: Generate a new DLC model.

This step generates a new DeepLabCut project.

1. In the main SimBA window, click on `Tracking` –> `DeepLabCut` –> `Create DLC Model`. The following window will pop up.



2. Enter the name of your project in the `Project name` entry box.
3. Enter the name of the experimenter in the `Experimenter name` entry box.

Note: Spaces are not allowed in the project name or the experimenter name.

4. Next, import the videos into your project. If you are adding only one video, you can click on `Import Single Video` and the `Video Folder` in **green** should change to `Video Path` **blue**. Next, click on *Browse File* to select a video file.

5. To import multiple videos into your project, click on `Import Multiple Video` and the **green** `Video Folder` will appear. Click on *Browse Folder* and choose a folder containing the videos to import into your project.

Note: The default settings is to **import multiple videos**.

6. Next, select the main directory that your project will be located in. Next to `Project directory`, click on `Browse Folder` and choose a directory.

7. **Optional**: If you have previously created a SimBA project, **and** a user-defined body-part configuration schematic in SimBA, you may want to use this configuration in your DeepLabCut tracking project. If so, go ahead and click on `Browse File` next to the text `Bp config file`. These body-part configurations are saved in you SimBA project, in a CSV file located in the `project_folder/logs/measures/pose_configs/bp_names\project_bp_names.csv` file. Navigate to this file and select it. If you **do not** want to use a SimBA-created bosy-part configuration file, leave the `Bp config file` entry box empty.

8. If you wish to use the settings used by the Golden lab (this setting is used to track two mice, and eight body parts on each of the two mice) check the `Apply Golden Aggression Config` box. For more information, see above section and figure. If you wish to generate your own DeepLabCut tracking config, using a different annotation configuration, please leave this box un-ticked.

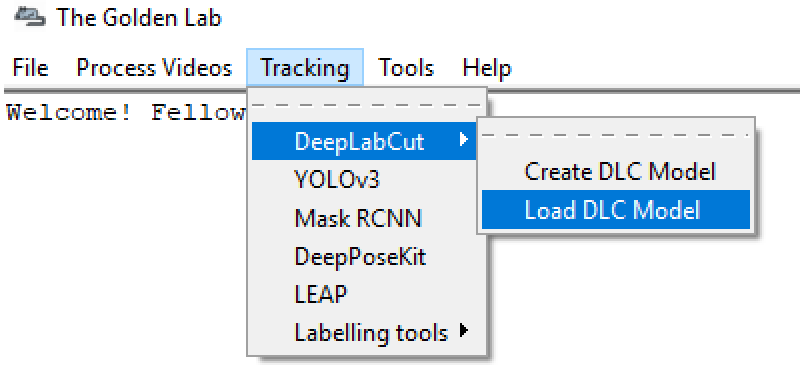
9. You can either copy all the videos to your DeepLabCut project folder, or create shortcuts to the videos. By checking the `Copy Videos` checkbox, the videos will be copied to your project folder. If this box is left un-ticked, shortcuts to your videos will be created. This can be useful if you wish to save storage space, or if you have very large or numerous video files.

10. Click on `Create Project` to create your DeepLabCut project. The project will be located in the chosen **working directory**.

11. You will need to Load this project to use it. Move to Part 2 below.

2.2.4 Part 2: Load DLC Project

These menus are used to load created DeepLabCut projects.



Step 1: Check the settings in the DeepLabCut config.yaml file

1. Go to the project folder.
2. Double click on the config.yaml file and open it in a text editor like Notepad.
3. Change the settings if necessary.

Step 2: Load the DeepLabCut config.yaml file

1. In the main SimBA window, click on `Tracking` –> `DeepLabCut` –> `Load DLC Model`. The following window will pop up.

2. Under the **Load Model** tab, click on `Browse File` and load the *config.yaml* file from your project folder.

Step 3: Add additional videos to the project (optional)

This step can be useful if you would like to add new videos to refine your tracking model that were not avilable previously.

Single Video

1. Under the **Add videos into project** tab and **Single Video** heading, click on `Browse File` and select the video you wish to add to the project.

2. Click on `Add single video`.

Multiple Videos

1. Under the **Add videos into project** tab and **Multiple Videos** heading, click on `Browse Folder` and select the folder containing the videos you wish to add to the project.

2. Click on `Add multiple videos`.

Step 4: Extract frames for labeling body parts using DLC. For more details, see the DeepLabcut repository

1. Under the **Extract/label frames tab** tab and **Extract Frames DLC** heading, enter the number of frames you wish to extract from the videos in the `numframes2pick` entry box.

2. Select the **Mode** of extracting the frames.
   - `Automatic` selects the frames to extract automatically.
   - `Manual` allows you to select the frames to extract.
3. Select the **Algorithm** used to pick the frames to extract.
   - `Uniform` selects the frames uniformly in a series format.
   - `KMeans` uses k-means clustering algorithm to select the frames randomly based on cluster.

- `Cluster Resize Width` is the setting for KMeans. The default is set to 30.
- `Cluster Step` is the setting for KMeans. The default is set to 1.

4. To use color information for clustering, check the `True` box next to **Cluster Color**.

5. If you wish to use **OpenCV** to extract frames, check the `True` box. If you wish to use **ffmpeg**, keep this box un-checked.

6. Click on `Extract Frames` to begin to extract the frames.

Step 5: Label Frames

1. Under **Label Frames** heading in the **Extract/label frames tab** tab, click on `Label Frames` and the *DeepLabCut-Labelling ToolBox* will pop up.

2. Click on *Load Frames* located in the bottom left-hand corner.

3. Navigate to the folder with the video name where your extracted frames are saved, and click `Select Folder`. *They should be located at workingdirectory/yourproject/labeled-data/*

4. Now you can start labelling frames. For more details, see the DeepLabCut repository

5. Once all the frames are labelled, move on to **Step 6**.

Step 6: Check Labels

1. Under **Check Labels**, click on `Check Labelled Frames` to check the labelled frames.

Step 7: Generate training set

1. Under **Generate Training Set**, click on the `Generate training set` button to generate a new training set.

Step 8: Train Network

Train the model using a training set. Note that there are two scenarios:

1. **Training from scratch**: In the `iteration` entry box, fill in an integer (e.g., 0) representing the model iteration number. Once done, click on **Update iteration**. If left blank, then the iteration number will be the most recently used. If the iteration number isn't specified, and you are working with a new project, it will default to 0.

2. **Training from previous weights**: In the `init_weight` box, specify the path to the initial weights. If this is left blank it will default to resnet-50. If you want to use other weights, click on `Browse File`.

Pre-trained weights for mouse and rat resident-intruder protocols using 16 body-parts, as well as other pre-trained weights, can be downloaded here. Downloaded weights may consist of three files (*.index, .meta, .data*). In the `init_weight` box, specify the path to any of the three downloaded files.

Step 9: Evaluate Network

1. Click on `Evaluate Network` to evaluate the trained model. For more details, see the DeepLabCut repository

Step 10: Video Analysis

Single Video Analysis

1. Under the **Video Analysis** tab and the **Single Video Analysis** header, click on `Browse File` and select one video file.

2. Click on `Single Video Analysis`.

Multiple Videos Analysis

1. Under the **Video Analysis** tab and the **Multiple Videos Analysis** header, click on *Browse Folder* and select a folder containing the videos you wish to analyze.

2. Enter the video format in the `Video type` entry box (eg: mp4, avi). *Do* ***not** include any " . " dot in the entry box (e.g,: for example, do not enter .mp4, .flv, or .avi. Instead enter mp4, flv, or avi*).

3. Click on `Multi Videos Analysis`.

Step 11: Plot Video Graph

1. Click on `Browse File` to select the video.

2. Click on `Plot Results`. For more details, see the DeepLabCut documentation

Step 12: Create Video

This step will generate a video with labelled tracking points.

1. Click on `Browse File` to select the video.

2. You can choose to save the frames with tracking points on the video, and not just the compressed video file, by checking the `Save Frames` box. This process takes longer if you want to render to whole video. However, if you just want to check on a few frames how the model is doing, you can render a few images and terminate the process and save time.

3. Click on `Create Video`.

2.2.5 Part 3: Improve the Model

There are two ways that you can improve DeepLabCut generated tracking models.

1. Extract more frames to train the model.
2. Correct the predictions made by the model on new frames.

Extract more frames

This step automates the **Extract Frames** process to only extract frames from select videos. This function will copy the settings from the DeepLabCut *config.yaml* and remove all the videos in the *config.yaml* file. The function will then add the new videos that the user specifies into a new, temporary yaml file, and extract frames from only the videos included in the temporary yaml file.

Step 1: Generate temp.yaml from the original config.yaml

Single Video

1. Under **Load Model**, click `Browse File` and select the *config.yaml* file.
2. Under `[Generate temp yaml]` tab –> `Single video`, click on *Browse File* and select the video.
3. Click on `Add single video`
4. A *temp.yaml* file will be generated in the same folder as the project folder.

Multiple Videos

1. Under **Load Model**, click `Browse File` and select the *config.yaml* file.

2. Under `[Generate temp yaml]` tab –> `Multiple videos`, click on `Browse Folder` and select the folder containing *only* the videos you want to add to your peoject and extract additional frames from.

3. Click on `Add multiple videos`.
4. A *temp.yaml* file will be generated in the same folder as the project folder.

Step 2: Load temp.yaml

1. Under **Load Model**, click `Browse File` and select the *temp.yaml* that you have just created.
2. Now you can extract frames of the videos that you have just added. Return to **Part 1, Step 4** to update the model with the new frames.

Tracking Point Correction

Step 1: Load config.yaml

1. Go to `Tracking` –> `DeepLabCut` –> `Load DLC Model`.
2. Under **Load Model**, click on *Browse File* and load the *config.yaml* file from the project folder.

Step 2: Extract Outliers

1. Under **Extract Outliers**, click `Browse File` to select the videos to you want to extract outlier frames from.
2. Click on `Extract Outliers`.

Step 3: Label Outliers

1. Under **Label Outliers**, click on `Refine Outliers`. The *DeepLabCut - Refinement ToolBox* will pop open.

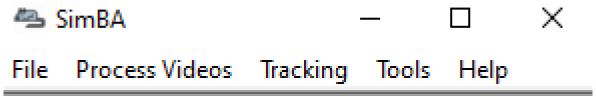
2. In the bottom left-hand corner, click on *Load labels* and select the *machinelabels* file to start to correct the tracking points. For more information on how to correct outliers in DeepLabCut, go to the DeepLabCut repository

Step 4: Merge Labelled Outliers

1. Under **Merge Labeled Outliers**, click on *Merge Labelled Outliers* to add the labelled outlier frames to your dataset.

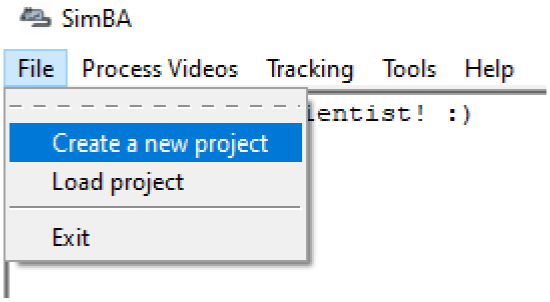
2.3 Step 3: SimBA pipeline

This is a generic step by step tutorial to start using SimBA to create behavioral classifiers. For more information about the menus and options and their use cases, see the SimBA Scenario tutorials



2.3.1 Part 1: Create a new project

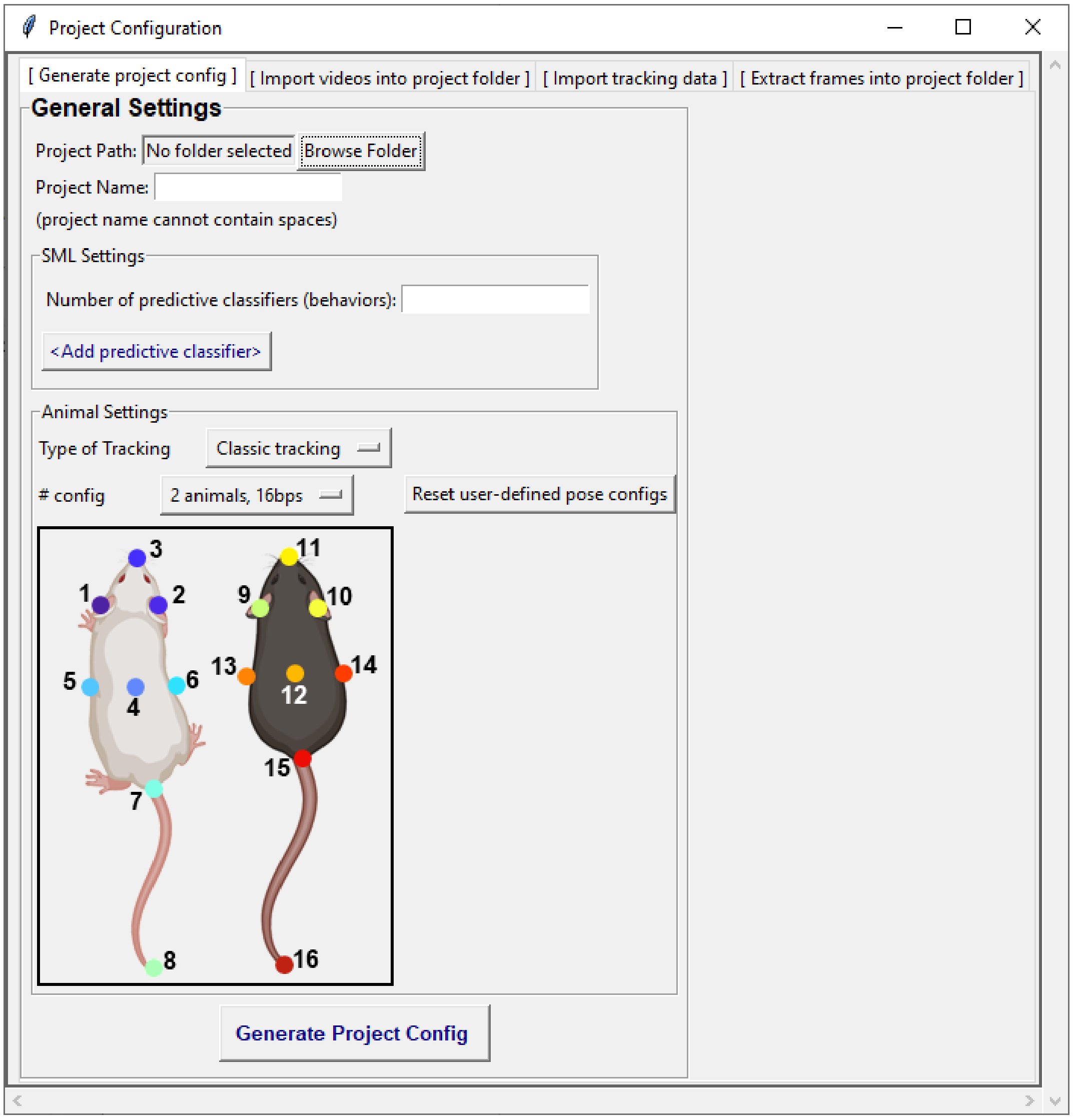
This section describes how to create a new project for your tracking analysis.



Step 1: Generate Project Config

In this step you create your main project folder with all the required sub-directories.

1. In the main SimBA window, click on `File` and and `Create a new project`. The following windows will pop up.



Project Configuration

[Generate project config] [Import videos into project folder] [Import tracking data] [Extract frames into project folder]

General Settings

Project Path: No folder selected

Project Name:

(project name cannot contain spaces)

SML Settings

Number of predictive classifiers (behaviors):

Animal Settings

Type of Tracking: Classic tracking

config: 2 animals, 16bps

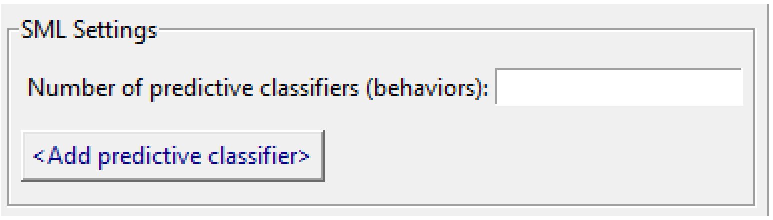
2. Navigate to the `[ Generate project config ]` tab. Under **General Settings**, specify a `Project Path` which is the directory that will contain your main project folder.

3. `Project Name` is the name of your project.

Note: *Keep in mind that the project name cannot contain spaces. Instead use underscore "_"*

4. Under *SML Settings*, put in the number of predictive classifiers that you wish to create. For an example, if you had three behaviors in your video, put 3 in the entry box.

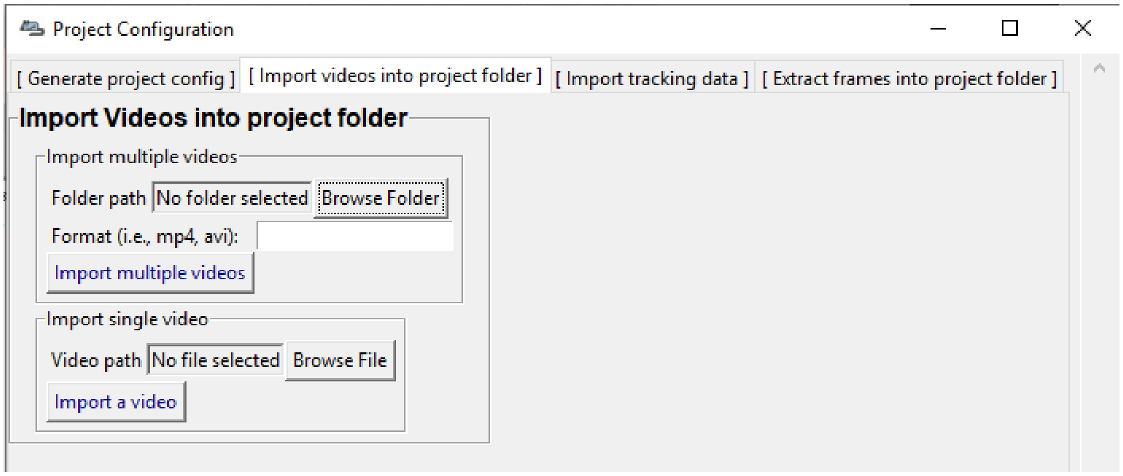
5. Click `Add Classifier` and it creates a row as shown in the following image. In each entry box, fill in the name of the behavior that you want to classify.


SML Settings
Number of predictive classifiers (behaviors):
<Add predictive classifier>

6. `Type of Tracking` allows the user to choose multi-animal tracking or the classic tracking.
7. `Animal Settings` is the number of animals and body parts that that the pose estimation tracking data contains. The default for **SimBA** is 2 animals and 16 body parts (*2 animals, 16bp*). There are a few other - ** yet not validated** - options, accessible in the dropdown menu.
8. Click on *Generate Project Config* to generate your project. The project folder will be located in the specified `Project Path`.

Step 2: Import videos into project folder

In this step, you can choose to import either one or multiple videos. The imported videos are used for visualizing predictions and standardizing distances across videos by calculating metric distances from pixel distances.



To import multiple videos

1. Navigate to the `[ Import videos into project folder ]` tab.
2. Under the `Import multiple videos` heading, click on `Browse Folder` to select a folder that contains all the videos that you wish to import into your project.
3. Enter the file type of your videos. (e.g., *mp4*, *avi*, *mov*, etc) in the `Video type` entry box.
4. Click on `Import multiple videos`.

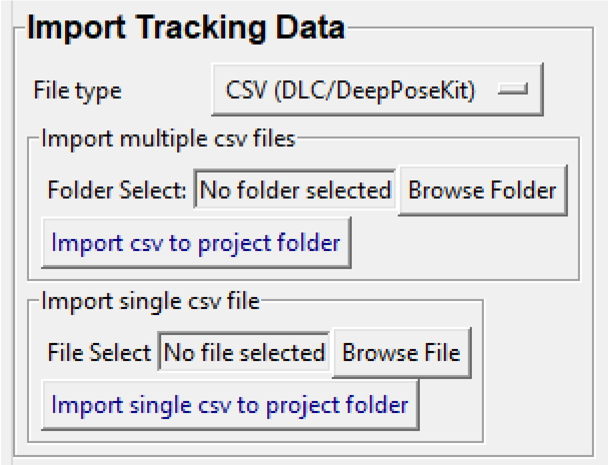
Note: If you have a lot of videos, it might take a few minutes before all the videos are imported.

To import a single video

1. Under the `Import single video` heading, click on `Browse File` to select your video.
2. Click on `Import a video`.

Step 3: Import Tracking Data

In this step, you will import your pose-estimation tracking data.



Import Tracking Data

File type: CSV (DLC/DeepPoseKit)

Import multiple csv files

Folder Select: No folder selected

Browse Folder
Import csv to project folder

Import single csv file

File Select: No file selected

Browse File
Import single csv to project folder

To import csv/json files

1. Navigate to the `[ Import tracking data ]` tab. Under the `Import tracking data` click on the `File type` drop down menu.
2. From the drop down menu, .csv files = `CSV (DLC/DeepPoseKit)`, and .json files = `JSON (BENTO)`.
3. To import multiple files, choose the folder that contains the files by clicking `Browse Folder`, then click `Import csv to project folder`.
4. To import a single file, choose the file by clicking `Browse File`, then click `Import single csv to project folder`.

To import h5 files (multi-animal DLC)

Please note that you can only import the h5 tracking data after you have imported the videos into the project folder.

1. From the `File type` drop down menu, select `H5 (multi-animal DLC)`.
2. Under `Animal settings`, enter the number of animals in the videos in the `No of animals` entry box, and click `Confirm`.
3. Enter the names for each of the animal in the video.
4. `Tracking type` is the type of tracking from DeepLabCut multi animal tracking.
5. Select the folder that contains all the h5 files by clicking `Browse Folder`.
6. Click `Import h5` to start importing.

To import slp files (SLEAP)

1. From the `File type` drop down menu, select `SLP (SLEAP)`.
2. Under `Animal settings`, enter the number of animals in the videos in the `No of animals` entry box, and click `Confirm`.
3. Enter the names for each of the animal in the video.
4. Select the folder that contains all the slp files by clicking `Browse Folder`.
5. Click on `Import .slp`.

Step 4: Extract frames into project folder

Tip: Users no longer have to extract frames into project folder as everything will be loaded using memory and the video files.

2.3.2 Part 2: Load project

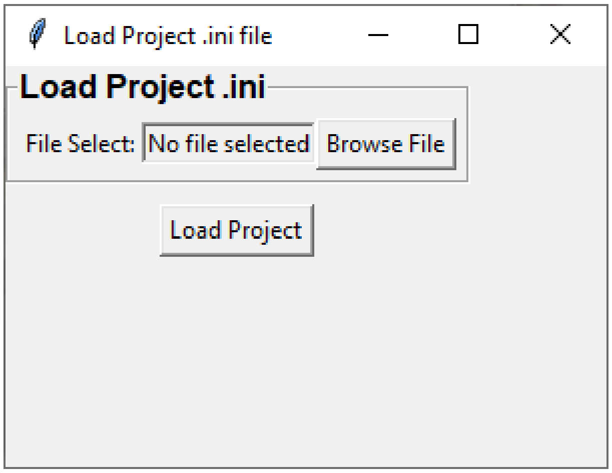
This section describes how to load and work with created projects.

Step 1: Load Project Config

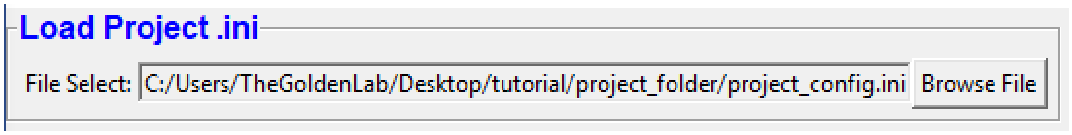
In this step you will load the *project_config.ini* file that was created.

Note: A project_config.ini should always be loaded before any other process.

1. In the main SimBA window, click on `File` and `Load project`. The following windows will pop up.



2. Click on `Browse File`. Then, go to the directory that you created your project in and click on your *project folder*. Locate the *project_config.ini* file and select it. Once this step is completed, it should look like the following, and you should no longer see the text *No file selected*.



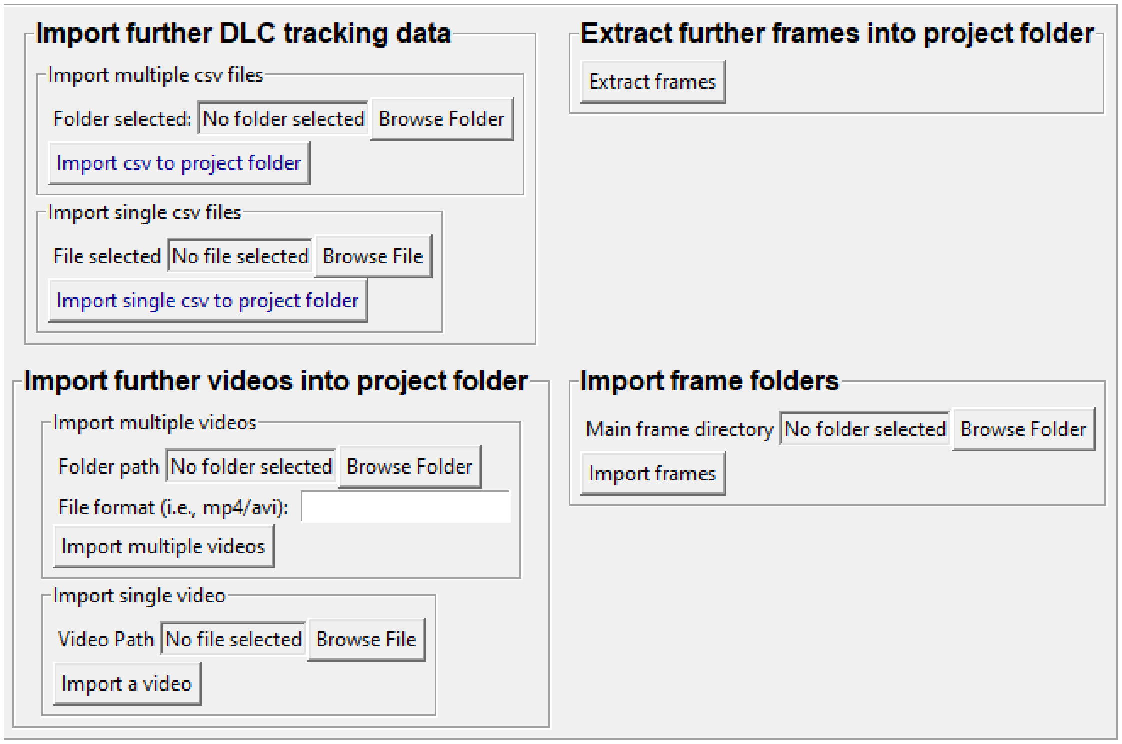
File Select: C:/Users/TheGoldenLab/Desktop/tutorial/project_folder/project_config.ini

In this image, you can see the `Desktop` is my selected working directory, `tutorial` is my project name, and the last two sections of the folder path is always going to be `project_folder/project_config.ini`.

3. Click on `Load Project`.

Step 2 (Optional) : Import more DLC Tracking Data or videos

In this step, you can choose to import more pose estimation data in csv file format and/or more videos. If this isn't relevant then you can skip this step.


Import further DLC tracking data
Import multiple csv files
Folder selected: No folder selected
Browse Folder
Import csv to project folder
Import single csv files
File selected No file selected
Browse File
Import single csv to project folder
Extract further frames into project folder
Extract frames
Import further videos into project folder
Import multiple videos
Folder path No folder selected
Browse Folder
File format (i.e., mp4/avi):
Import multiple videos
Import single video
Video Path No file selected
Browse File
Import a video
Import frame folders
Main frame directory No folder selected
Browse Folder
Import frames

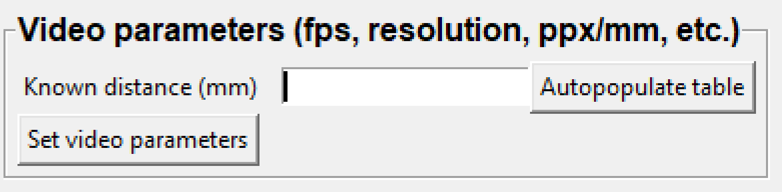
1. Click on the `[ Further imports (data/video/frames) ]` tab. From here you can import more data or videos into the project folder. The imported .csv files will be placed in the `project_folder/csv/input` directory, and the imported videos will be placed in the `project_folder/videos` directory.

2. Once the videos are imported, you can extract frames from the additional videos by clicking on `Extract frames` under the **Extract further frames into project folder** heading.

3. If you already have existing frames of the videos in the project folder, you can import the folder that contains the frames into the project. Under the **Import frame folders** heading, click on `Browse Folder` to choose the folder thar contains the frames, and click on `Import frames`. The frames will be imported into the `project_folder/frames/input` folder.

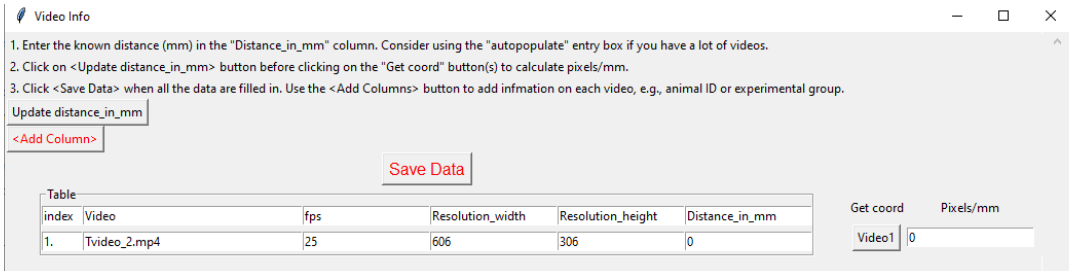
Step 3: Set video parameters

In this step, you can customize the meta parameters for each of your videos (fps, resolution, metric distances) and provide additional custom video information (Animal ID, group etc). You also set the **pixels per millimeter** for your videos. You will be using a tool that requires the known distance between two points (e.g., the cage width or the cage height) in order to calculate **pixels per millimeter**. The real life distance between the two points is called `Distance in mm`.



1. Under **Set video parameters(distances,resolution,etc.)**, the entry box named `Distance in mm` is the known distance between two points in the videos in millimeter. If the known distance is the same in all the videos in the project, then enter the value *(e.g,: 245)* and click on `Auto populate Distance in mm in tables`. and it will auto-populate the table in the next step (see below). If you leave the *Distance in mm* entry box empty, the known distance will default to zero and you will fill in the value for each video individually.

2. Click on `Set Video Parameters` and the following windows will pop up.



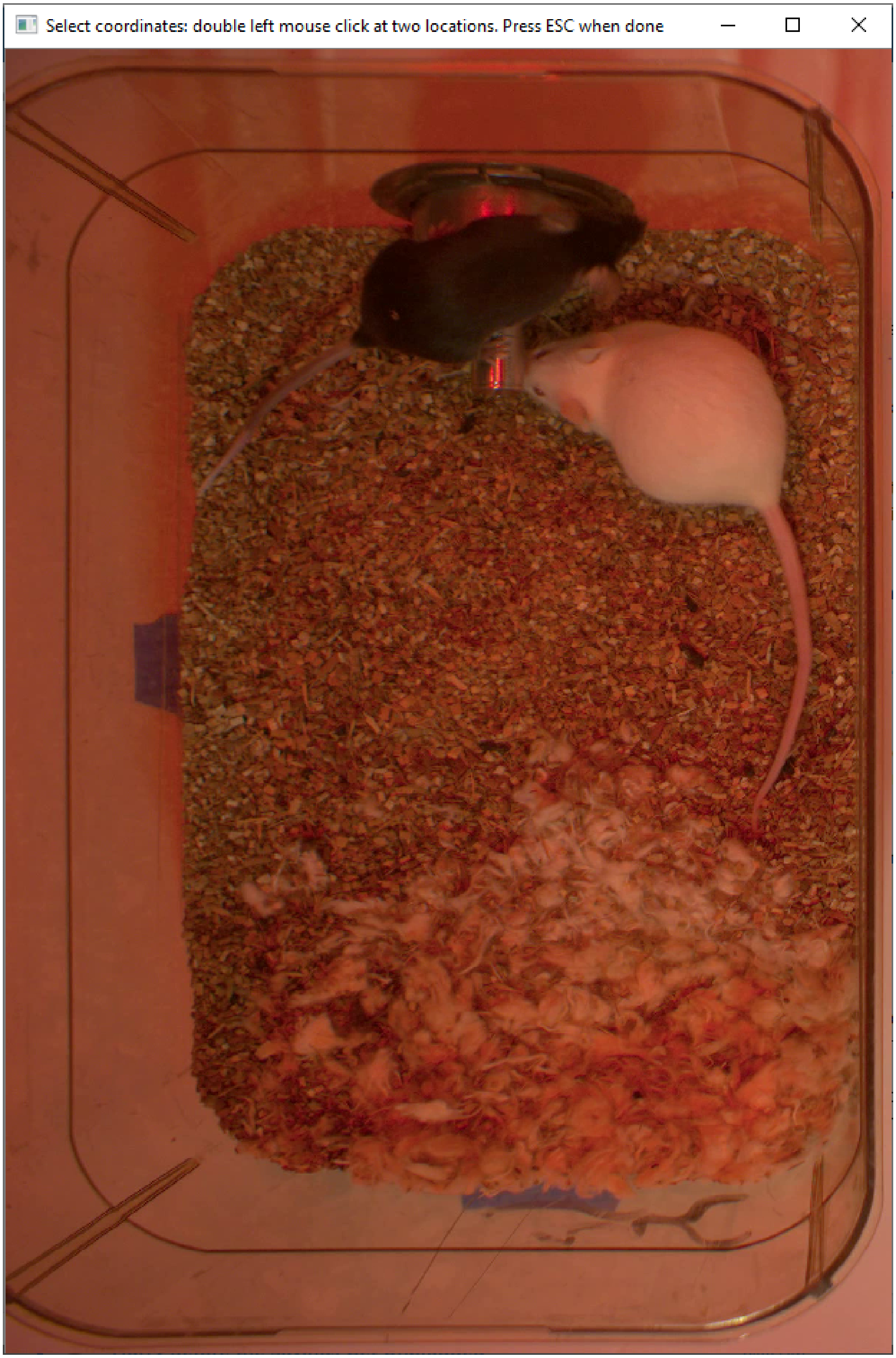
1. Enter the known distance (mm) in the "Distance_in_mm" column. Consider using the "autopopulate" entry box if you have a lot of videos.
2. Click on <Update distance_in_mm> button before clicking on the "Get coord" button(s) to calculate pixels/mm.
3. Click <Save Data> when all the data are filled in. Use the <Add Columns> button to add infmation on each video, e.g., animal ID or experimental group.

index	Video	fps	Resolution_width	Resolution_height	Distance_in_mm
1.	Tvideo_2.mp4	25	606	306	0

3. In the above example I imported four videos and their names are listed the leftmost `Video` column. I auto-populated the known distance to 10 millimeter in the previous step, and this is now displayed in the `Distance in mm` column.

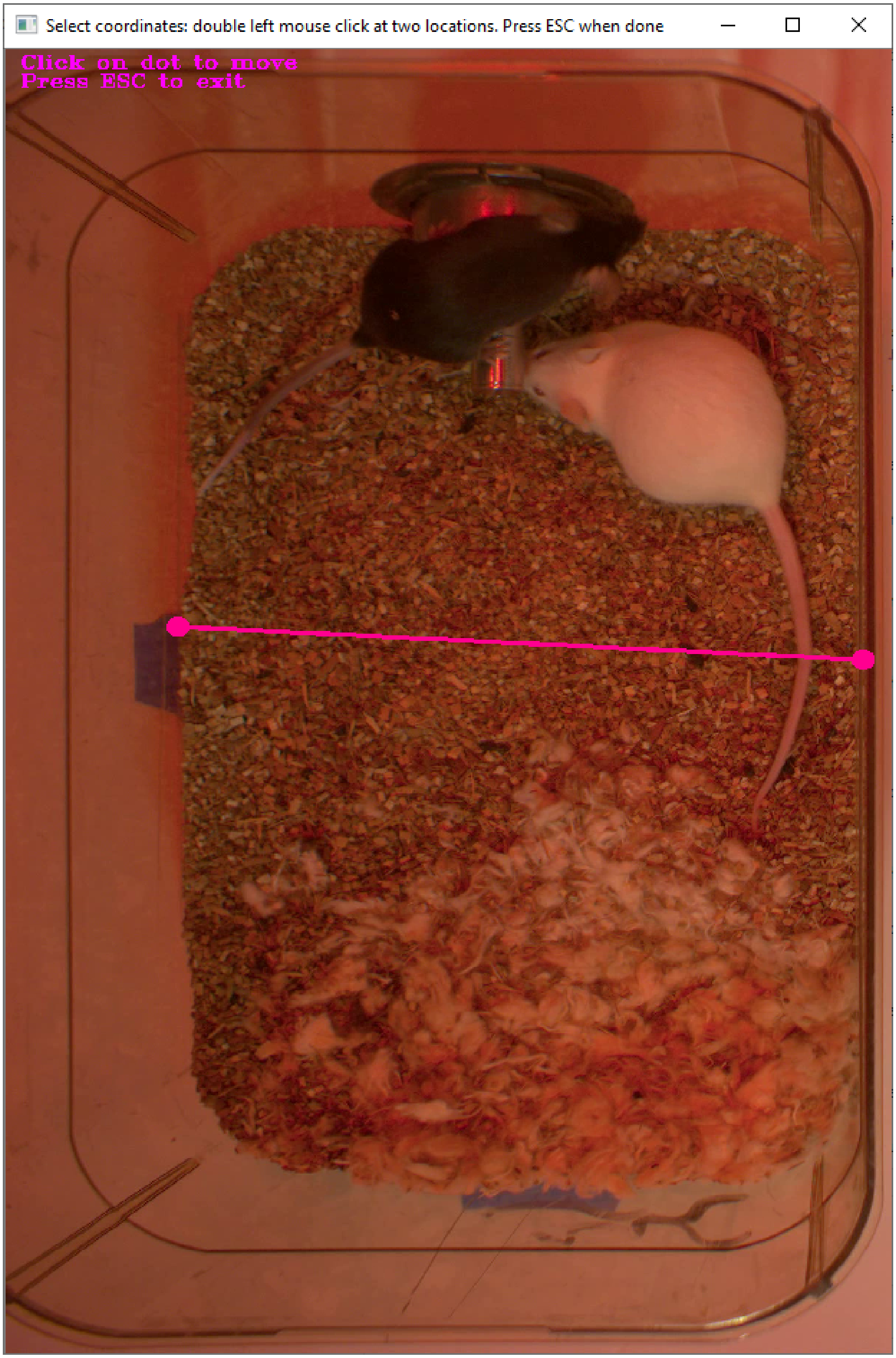
4. I can click on the values in the entry boxes and change them until I am satisfied. Then, I click on `Update distance_in_mm` and this will update the whole table.

5. Next, to get the `Pixels/mm` for the first video, click on `Video1` and the following window will pop up. The window that pops up displays the first frame of `Video1`.



Select coordinates: double left mouse click at two locations. Press ESC when done

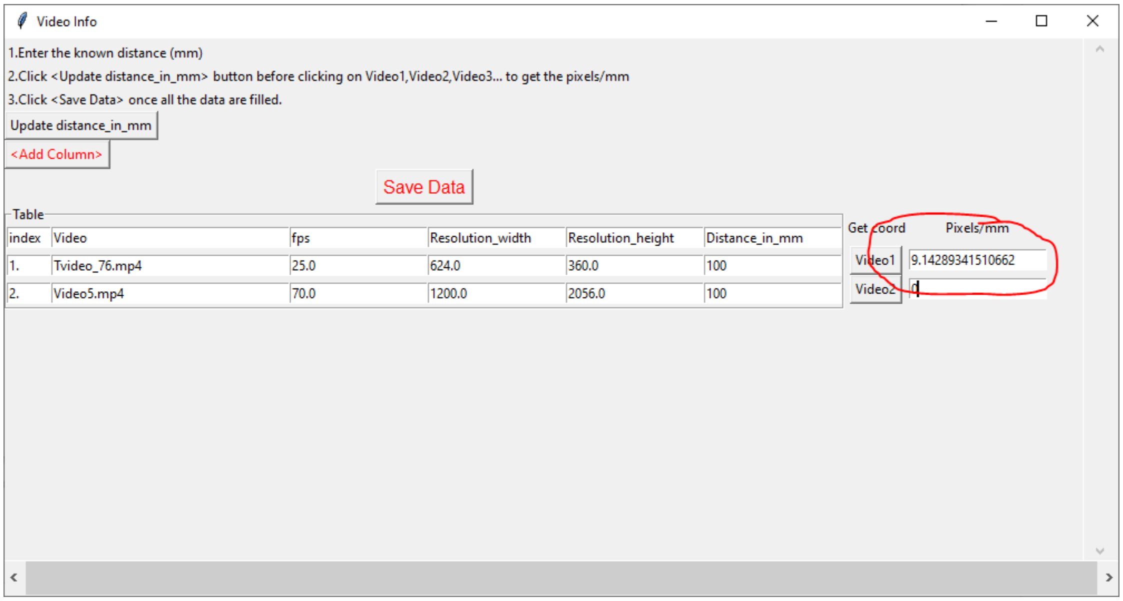
6. Now, double **left** click to select two points that defines the known distance in real life. In this case, I know that the two **pink connected dots** represent a distance of 10 millimeter in real life.



Select coordinates: double left mouse click at two locations. Press ESC when done

7. If you misplaced one or both of the dots, you can double click on either of the dots to place them somewhere else in the image. Once you are done, hit `Esc`.

8. If every step is done correctly, the `Pixels/mm` column in the table should populate with the number of pixels that represent one millimeter,



1.Enter the known distance (mm)
2.Click <Update distance_in_mm> button before clicking on Video1,Video2,Video3... to get the pixels/mm
3.Click <Save Data> once all the data are filled.

index	Video	fps	Resolution_width	Resolution_height	Distance_in_mm
1.	Tvideo_76.mp4	25.0	624.0	360.0	100
2.	Video5.mp4	70.0	1200.0	2056.0	100

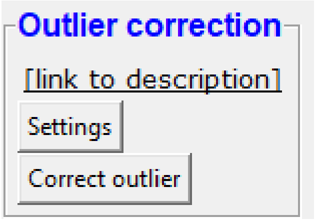
Get coord	Pixels/mm
Video1	9.14289341510662
Video2	0

9. Repeat the steps for every video in the table, and once it is done, click on `Save Data`. This will generate a csv file named **video_info.csv** in `/project_folder/log` folder that contains a table with your video meta data.

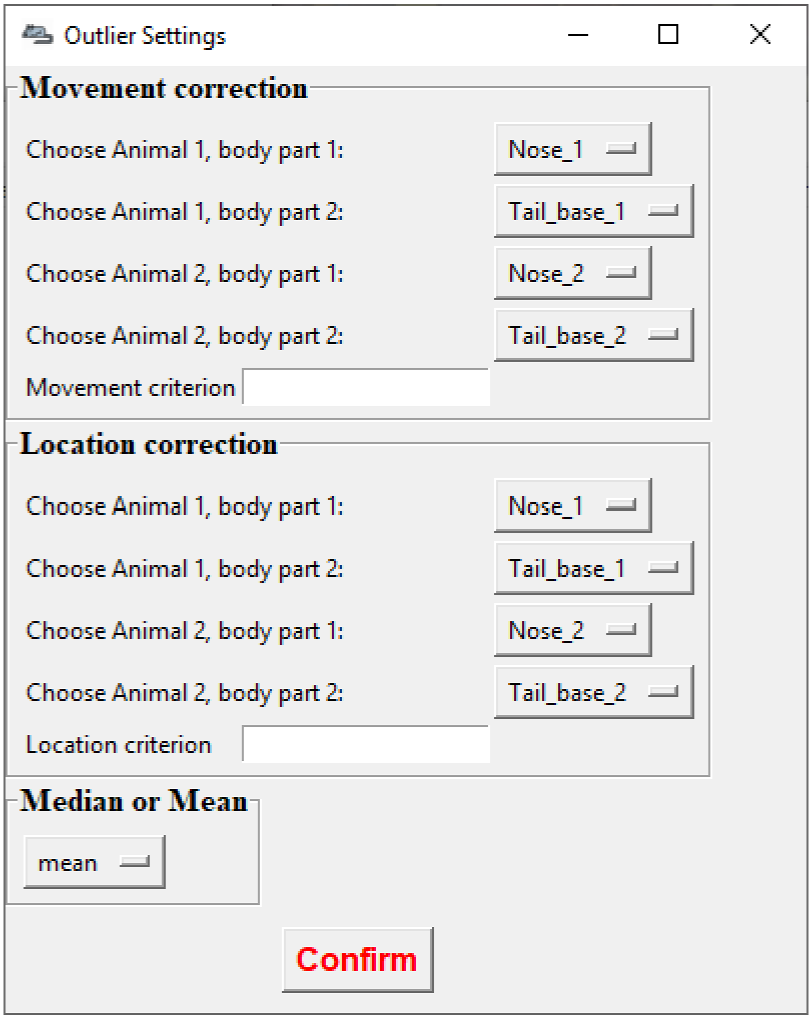
10. You can also chose to add further columns to the meta data file (e.g., AnimalID or experimental group) by clicking on the `Add Column` button. This information will be saved in additional columns to your **video_info.csv** file.

Step 4: Outlier Correction

Outlier correction is used to correct gross tracking inaccuracies by detecting outliers based on movements and locations of body parts in relation to the animal body length. For more details, please click here



1. Click on `Settings` and the following window will pop up. The Outlier Settings window varies with the number of animals in the project. The images below shows settings for two animals.



Outlier Settings

Movement correction

Choose Animal 1, body part 1: Nose_1

Choose Animal 1, body part 2: Tail_base_1

Choose Animal 2, body part 1: Nose_2

Choose Animal 2, body part 2: Tail_base_2

Movement criterion

Location correction

Choose Animal 1, body part 1: Nose_1

Choose Animal 1, body part 2: Tail_base_1

Choose Animal 2, body part 1: Nose_2

Choose Animal 2, body part 2: Tail_base_2

Location criterion

Median or Mean

mean

2. Select the body parts for Animal 1 and Animal 2 that you want to use to calculate a reference value. The reference value will be the mean or median Euclidian distance in millimeters between the two body parts of the two animals in all frames.

3. Enter values for the `Movement criterion` and the `Location criterion`.

   - `Movement criterion`. A body part coordinate will be flagged and corrected as a "movement outlier" if the body part moves the *reference value times the criterion value* across two sequential frames.
   - `Location criterion`. A body part coordinate will be flagged and correct as a "location outlier" if the distance between the body part and at least two other body parts belonging to the same animal are longer than the *reference value times the criterion value* within a frame.

Body parts flagged as movement or location outliers will be re-placed in their last reliable coordinate.

4. Chose to calculate the median or mean Euclidian distance in millimeters between the two body parts and click on `Confirm Config`.

5. Click to run the outlier correction. You can follow the progress in the main SimBA window. Once complete, two new csv log files will appear in the `/project_folder/log` folder. These two files contain the number of body parts corrected following the two outlier correction methods for each video in the project.

Step 5: Extract Features

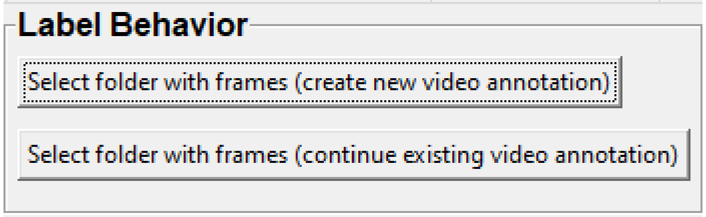
Based on the coordinates of body parts in each frame - and the frame rate and the pixels per millimeter values - the feature extraction step calculates a larger set of features used for behavioral classification. Features are values such as metric distances between body parts, angles, areas, movement, paths, and their deviations and rank in individual frames and across rolling windows. This set of features will depend on the body-parts tracked during pose-estimation (which is defined when creating the project). Click here for an example list of features when tracking 2 mice and 16 body parts.

1. Click on `Extract Features`.

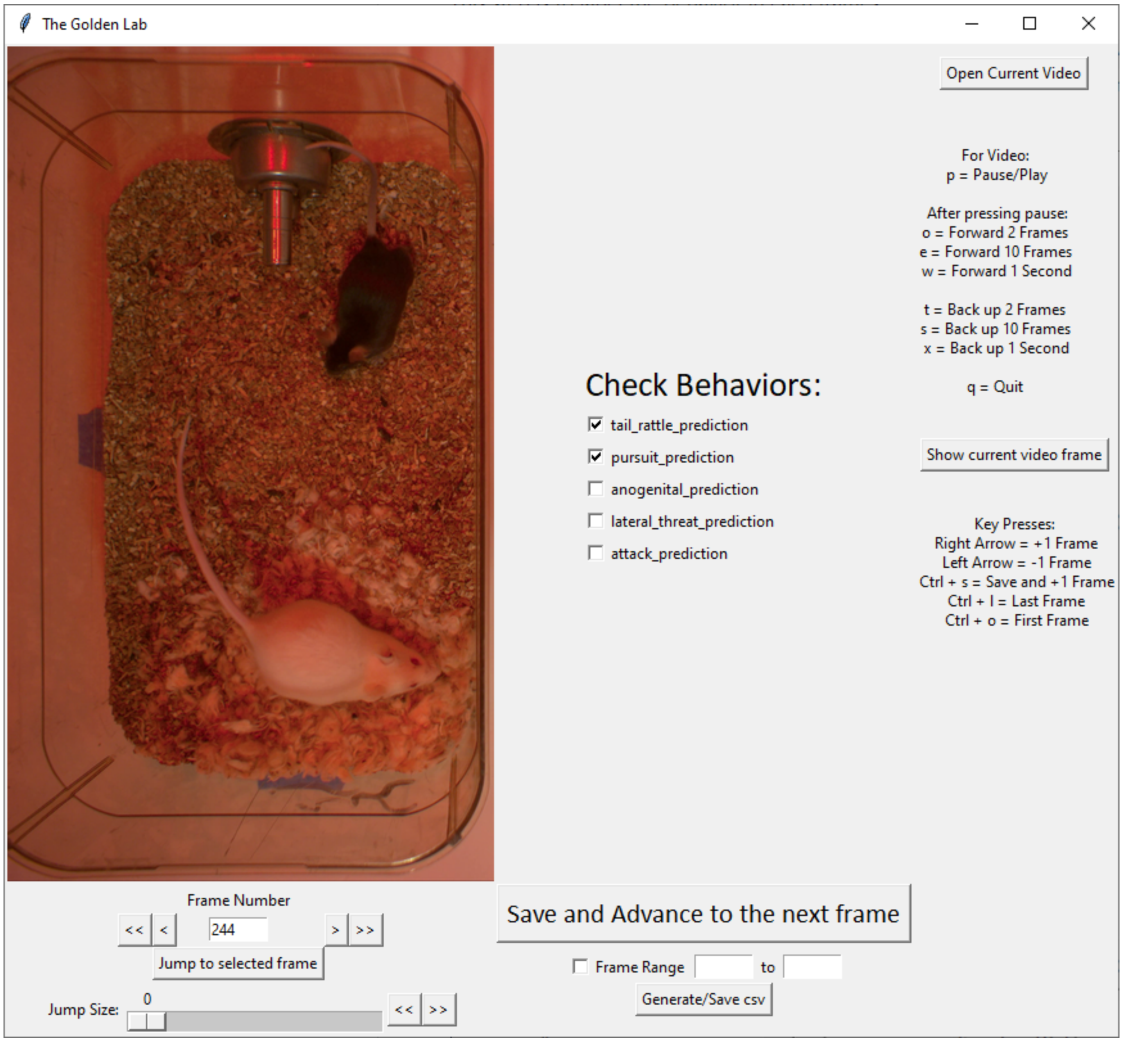
Step 6: Label Behavior

This step is used for label the behaviors in each frames of a video. This data will be concatenated with the features and used for creating behavioral classifiers.

There are two options, one is to start a **new video annotation** and one is to **continue on where you last left off**. Both are essentially the same, except the latter will start with the frame where you last saved. For example, one day, you started a new video by clicking `Select video (create new video annotation)` and you feel tired and sick of annotating the videos. You can now click `Generate/Save` button to save your work for your coworker to continue. Your coworker can continue by clicking ` Select folder with frames(continue existing video annotation)` and select the the video folder that you have annotated half way and take it from there!


Label Behavior
Select folder with frames (create new video annotation)
Select folder with frames (continue existing video annotation)

1. Click on `Select video`. In your project folder navigate to the `/project_folder/videos/` folder, and you should select the videos that you wished to annotate.



For Video:
p = Pause/Play

After pressing pause:
o = Forward 2 Frames
e = Forward 10 Frames
w = Forward 1 Second

t = Back up 2 Frames
s = Back up 10 Frames
x = Back up 1 Second

q = Quit

Check Behaviors:

- ☑ tail_rattle_prediction
- ☑ pursuit_prediction
- □ anogenital_prediction
- □ lateral_threat_prediction
- □ attack_prediction

Key Presses:
Right Arrow = +1 Frame
Left Arrow = -1 Frame
Ctrl + s = Save and +1 Frame
Ctrl + l = Last Frame
Ctrl + o = First Frame

Frame Number: 244

□ Frame Range ____ to ____

2. Please click here to learn how to use the behavior annotation interface.
3. Once finished, click on `Generate/Save` and it will generate a new *.csv* file in */csv/targets_inserted* folder.

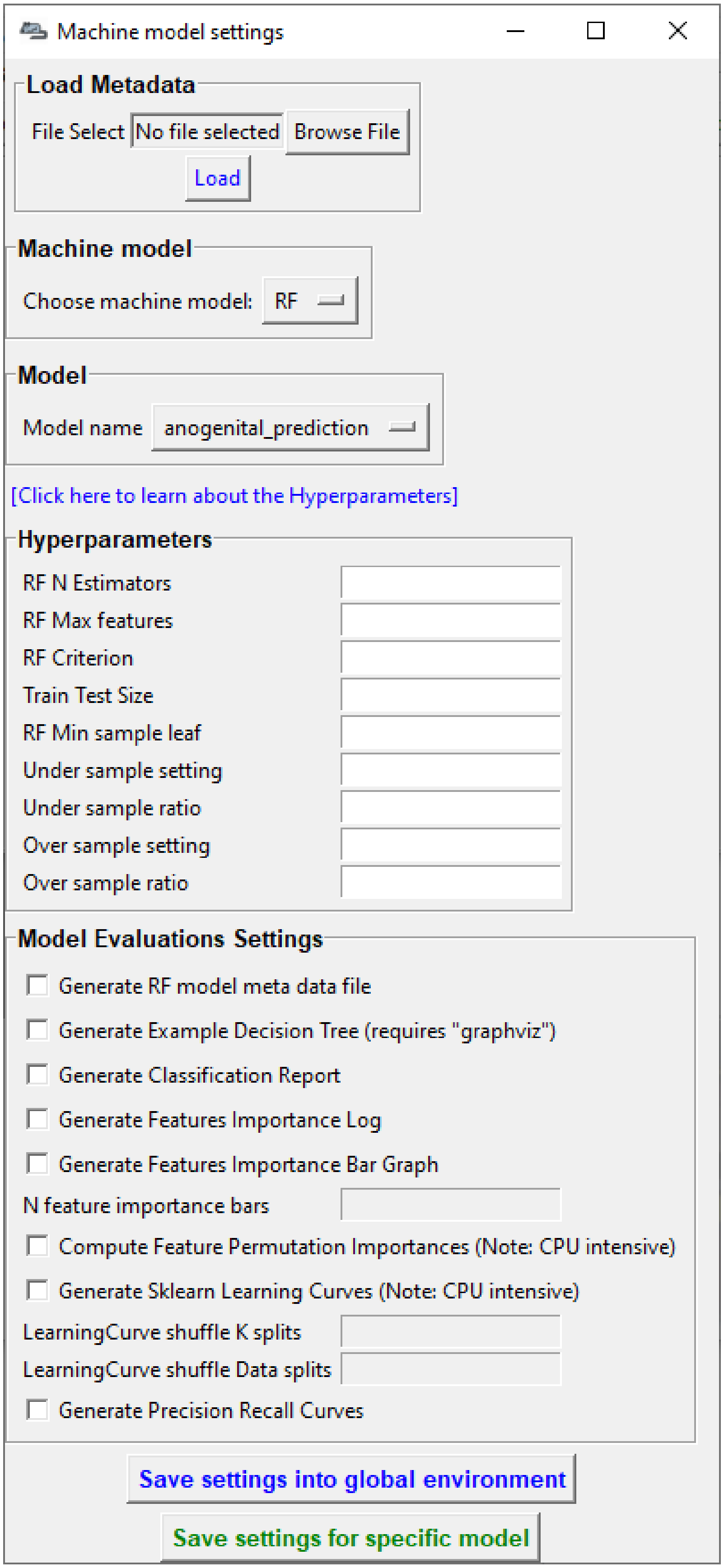
Step 7: Train Machine Model

This step is used for training new machine models for behavioral classifications.

Note: If you import existing models, you can skip this step and go straight to **Step 8** to run machine models on new video data.

Train single model

1. Click on `Settings` and the following window will pop up.



Machine model settings

Load Metadata

File Select: No file selected | Browse File

Load

Machine model

Choose machine model: RF

Model

Model name: anogenital_prediction

[Click here to learn about the Hyperparameters]

Hyperparameters

RF N Estimators	
RF Max features	
RF Criterion	
Train Test Size	
RF Min sample leaf	
Under sample setting	
Under sample ratio	
Over sample setting	
Over sample ratio	

Model Evaluations Settings

- ☐ Generate RF model meta data file
- ☐ Generate Example Decision Tree (requires "graphviz")
- ☐ Generate Classification Report
- ☐ Generate Features Importance Log
- ☐ Generate Features Importance Bar Graph

N feature importance bars

- ☐ Compute Feature Permutation Importances (Note: CPU intensive)
- ☐ Generate Sklearn Learning Curves (Note: CPU intensive)

LearningCurve shuffle K splits

LearningCurve shuffle Data splits

- ☐ Generate Precision Recall Curves

Save settings into global environment

Save settings for specific model

Note: If you have a .csv file containing hyper-parameter meta data, you can import this file by clicking on `Browse File` and then click on `Load`. This will autofill all the hyper-parameter entry boxes and model evaluation settings.

2. Under **Machine Model**, choose a machine model from the drop down menu: `RF` , `GBC`, `XGboost`.
   - `RF` : Random forest
   - `GBC` : Gradient boost classifier
   - `XGboost` : eXtreme Gradient boost
3. Under the **Model** heading, use the dropdown menu to select the behavioral classifier you wish to define the hyper-parameters for.
4. Under **Hyperparameters**, select the hyper-parameter settings for your model. For more details, please click here. Alternatively, import the recommended settings from a meta data file (see above, **Step 1**).
   - `RF N estimators`: Number of decision trees in the decision ensemble.
   - `RF Max features`: Number of features to consider when looking for the best split.
   - `RF Criterion`: The metric used to measure the quality of each split, i.e "gini" or "entropy".
   - `Train Test Size`: The ratio of the dataset withheld for testing the model (e.g., 0.20).
   - `RF Min sample leaf`: The minimum number of samples required to be at a leaf node.
   - `Under sample setting`: "Random undersample" or "None". If "Random undersample", a random sample of the majority class will be used in the train set. The size of this sample will be taken as a ratio of the minority class and should be specified in the "under sample ratio" box below. For more information, click here
   - `Under sample ratio`: The ratio of samples of the majority class to the minority class in the training data set. Applied only if "Under sample setting" is set to "Random undersample". Ignored if "Under sample setting" is set to "None" or NaN.
   - `Over sample setting`: "SMOTE", "SMOTEEN" or "None". If "SMOTE" or "SMOTEEN", synthetic data will be generated in the minority class based on k-means to balance the two classes. For more details, click here. Alternatively, import recommended settings from a meta data file (see **Step 1**).
   - `Over sample ratio`: The desired ratio of the number of samples in the minority class over the number of samples in the majority class after over sampling.
5. Under **Model Evaluation Settings**.

- `Generate RF model meta data file`: Generates a .csv file listing the hyper-parameter settings used when creating the model. The generated meta file can be used to create further models by importing it in the **Load Settings** menu (see above, **Step 1**).
- `Generate Example Decision Tree`: Saves a visualization of a random decision tree in .pdf and .dot formats. Requires graphviz. For more information, click here
- `Generate Classification Report`: Saves a classification report truth table in .png format. Depends on yellowbrick. For more information, click here
- `Generate Features Importance Log`: Creates a .csv file that lists the importance's gini importances of all features for the classifier.
- `Generate Features Importance Bar Graph`: Creates a bar chart of the top N features based on gini importances. Specify N in the `N feature importance bars` entry box below.
- `N feature importance bars`: Integer defining the number of top features to be included in the bar graph (e.g., 15).

- `Compute Feature Permutation Importance's`: Creates a .csv file listing the importance's (permutation importance's) of all features for the classifier. For more details, please click here. **Note:** Calculating permutation importance's is computationally expensive and takes a long time.
- `Generate Sklearn Learning Curve`: Creates a .csv file listing the f1 score at different test data sizes. For more details, please click here. This is useful for estimating the benefit of annotating further data.
- `LearningCurve shuffle K splits`: Number of cross validations applied at each test data size in the learning curve.
- `LearningCurve shuffle Data splits`: Number of test data sizes in the learning curve.
- `Generate Precision Recall Curves`: Creates a .csv file listing precision at different recall values. This is useful for titration of the false positive vs. false negative classifications of the models.

6. Click on the `Save settings into global environment` button to save your settings into the *project_config.ini* file and use the settings to train a single model.
7. Alternatively, click on the `Save settings for specific model` button to save the settings for one model. To generate multiple models - for either multiple different behaviors and/or using multiple different hyper-parameters - re-define the Machine model settings and click on `Save settings for specific model` again. Each time the `Save settings for specific model` is clicked, a new config file is generated in the */project_folder/configs* folder. In the next step (see below), a model for each config file will be created if pressing the **Train multiple models, one for each saved settings** button.
8. If training a single model, click on `Train Model`.

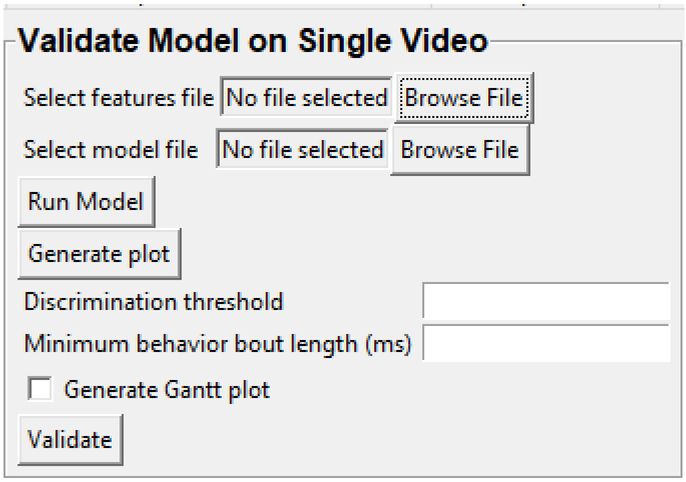
To train multiple models

1. Click on `Settings`.
2. Under **Machine Model**, choose the machine model from the drop down menu, `RF` , `GBC`, `Xboost`.
3. Under **Model**, select the model you wish to train from the drop down menu.
4. Then, set the **Hyperparameters**.
5. Click the `Save settings for specific model` button. This generates a meta.csv file, located in your `project_folder/configs` directory, which contains your selected hyperparameters. Repeat the steps to generate multiple models. On model will be generated for each of the meta.csv files in the `project_folder/configs` directory.
6. Close the `Machine models settings` window.
7. Click on the green `Train Multiple Models, one for each saved settings` button.

Optional step before running machine model on new data

The user can validate each model *(saved in .sav format)* file. In this validation step the user specifies the path to a previously created model in .sav file format, and a .csv file containing the features extracted from a video. This process will (i) run the classifications on the video, and (ii) create a video with the predictions overlaid together with a gantt plot showing predicted behavioral bouts. Click here for an example validation video.

1. Click `Browse File` and select the *project_config.ini* file and click `Load Project`.
2. Under **[Run machine model]** tab –> **validate Model on Single Video**, select your features file (.csv). It should be located in `project_folder/csv/features_extracted`.



Validate Model on Single Video

Select features file: No file selected

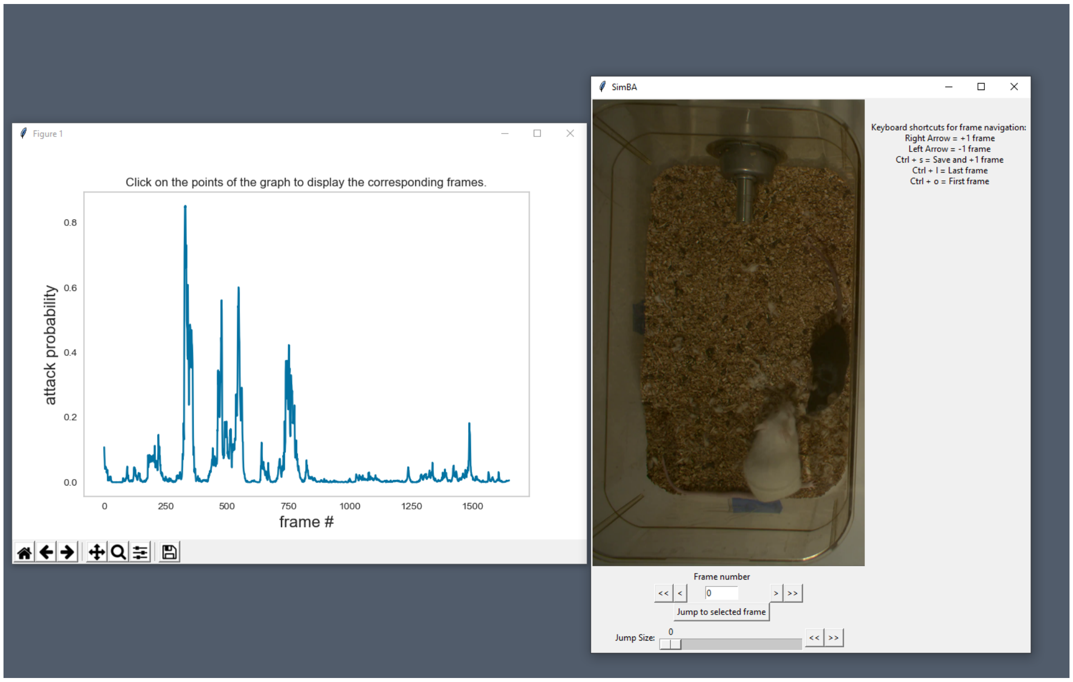
Select model file: No file selected

Discrimination threshold:

Minimum behavior bout length (ms):

□ Generate Gantt plot

3. Under `Select model file`, click on `Browse File` to select a model *(.sav file)*.
4. Click on `Run Model`.
5. Once, it is completed, it should print *"Predictions generated."*, now you can click on `Generate plot`. A graph window and a frame window will pop up.
    - `Graph window`: model prediction probability versus frame numbers will be plot. The graph is interactive, click on the graph and the frame window will display the selected frames.
    - `Frame window`: Frames of the chosen video with controls.



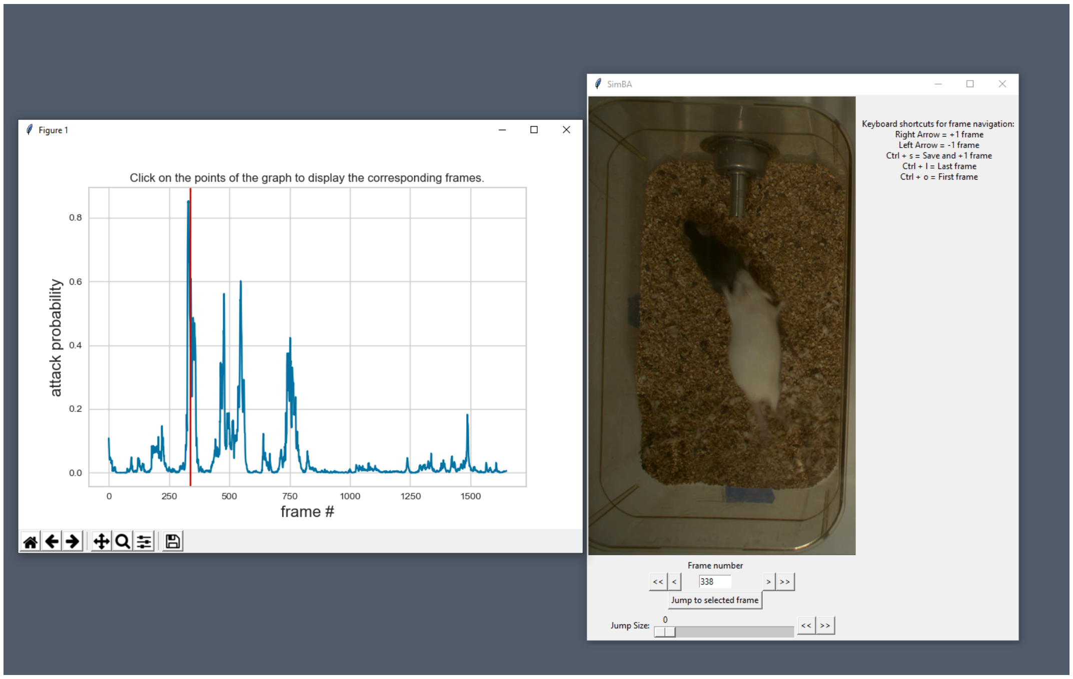
Click on the points of the graph to display the corresponding frames.

Keyboard shortcuts for frame navigation:
Right Arrow = +1 frame
Left Arrow = -1 frame
Ctrl + s = Save and +1 frame
Ctrl + l = Last frame
Ctrl + o = First frame

Frame number: 0

Jump Size: 0

7. Click on the points on the graph and picture displayed on the other window will jump to the corresponding frame. There will be a red line to show the points that you have clicked.



Click on the points of the graph to display the corresponding frames.

Keyboard shortcuts for frame navigation:
Right Arrow = +1 frame
Left Arrow = -1 frame
Ctrl + s = Save and +1 frame
Ctrl + l = Last frame
Ctrl + o = First frame

Frame number
338

8. Once it jumps to the desired frame, you can navigate through the frames to determine if the behavior is present. This step is to find the optimal threshold to validate your model.

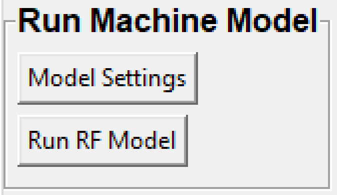
9. Once the threshold is determined, enter the threshold into the `Discrimination threshold` entry box and the desire minimum behavior bouth length into the `Minimum behavior bout lenght(ms)` entrybox.

   - `Discrimination threshold`: The level of probability required to define that the frame belongs to the target class. Accepts a float value between 0.0-1.0. For example, if set to 0.50, then all frames with a probability of containing the behavior of 0.5 or above will be classified as containing the behavior. For more information on classification threshold, click here
   - `Minimum behavior bout length (ms)`: The minimum length of a classified behavioral bout. **Example**: The random forest makes the following attack predictions for 9 consecutive frames in a 50 fps video: 1,1,1,1,0,1,1,1,1. This would mean, if we don't have a minimum bout length, that the animals fought for 80ms (4 frames), took a brake for 20ms (1 frame), then fought again for another 80ms (4 frames). You may want to classify this as a single 180ms attack bout rather than two separate 80ms attack bouts. With this setting you can do this. If the minimum behavior bout length is set to 20, any interruption in the behavior that is 20ms or shorter will be removed and the behavioral sequence above will be re-classified as: 1,1,1,1,1,1,1,1,1 - and instead classified as a single 180ms attack bout.

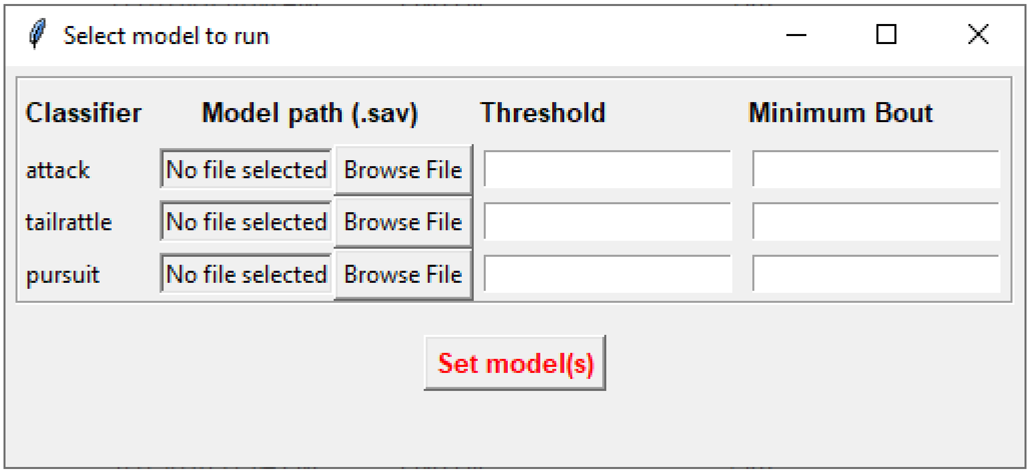
10. Click `Validate` to validate your model. **Note that this step will take a long time as it will generate a lot of frames.**

Step 8: Run Machine Model

This step runs behavioral classifiers on new data.



1. Under the **Run Machine Model** heading, click on `Model Selection`. The following window with the classifier names defined in the *project_config.ini* file will pop up.

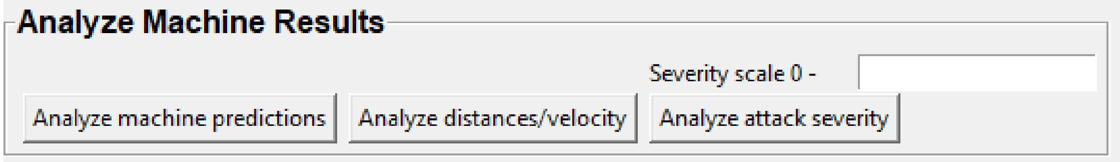


Classifier	Model path (.sav)		Threshold	Minimum Bout
attack	No file selected	Browse File		
tailrattle	No file selected	Browse File		
pursuit	No file selected	Browse File		

2. Click on `Browse File` and select the model (*.sav*) file associated with each of the classifier names.
3. Once all the models have been chosen, click on `Set Model` to save the paths.
4. Fill in the `Discrimination threshold`.
   - `Discrimination threshold`: The level of probability required to define that the frame belongs to the target class (see above).
5. Fill in the `Minimum behavior bout length`.
   - `Minimum behavior bout length (ms)`: The minimum length of a classified behavioral bout(see above).
6. Click on `Set model(s)` and then click on `Run RF Model` to run the machine model on the new data.

Step 9: Analyze Machine Results

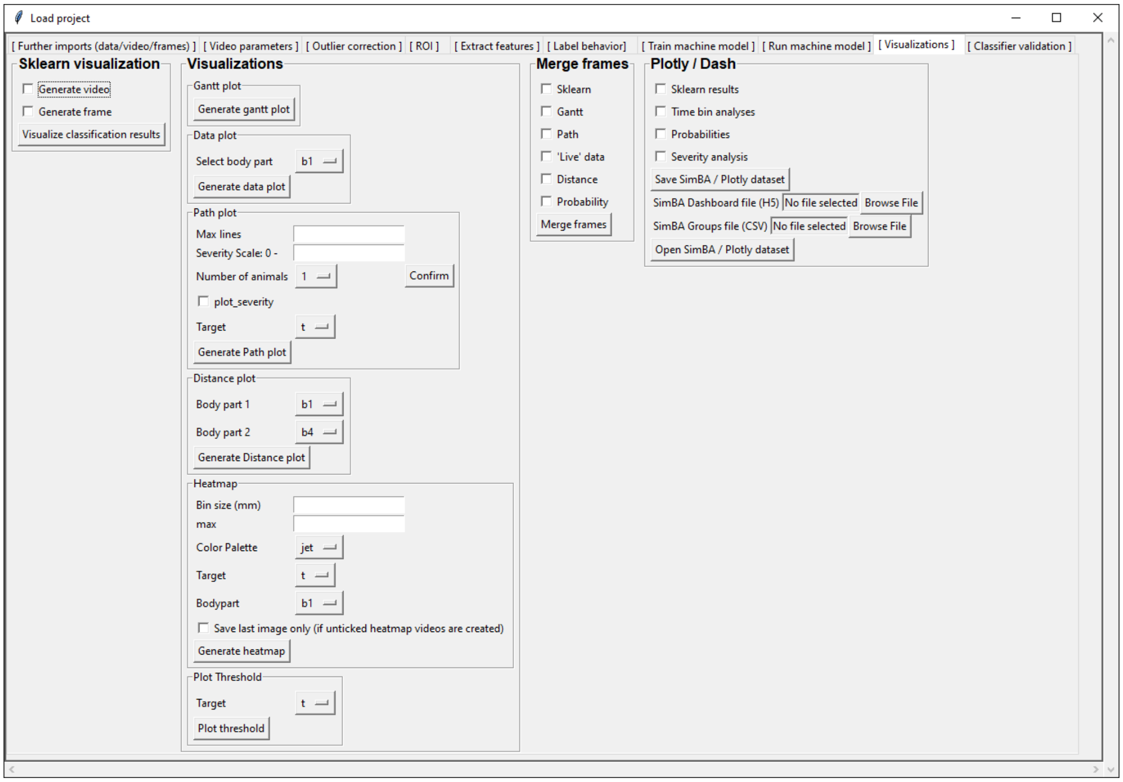
Access this menu through the `Load project` menu and the `Run machine model` tab. This step performs summary analyses and presents descriptive statistics in .csv file format. There are three forms of summary analyses: `Analyze`, `Analyze distance/velocity`, and `Analyze severity`.



- `Analyze`: This button generates descriptive statistics for each predictive classifier in the project, including the total time, the number of frames, total number of 'bouts', mean and median bout interval, time to first occurrence, and mean and median interval between each bout. A date-time stamped output csv file with the data is saved in the `/project_folder/log` folder.
- `Analyze distance/velocity`: This button generates descriptive statistics for mean and median movements and distances between animals. The date-time stamped output csv file with the data is saved in the `/project_folder/log` folder.
- `Analyze severity`: Calculates the 'severity' of each frame classified as containing attack behavior based on a user-defined scale. **Example:** the user sets a 10-point scale. One frame is predicted to contain an attack, and the total body-part movements of both animals in that frame is in the top 10% percentile of movements in the entire video. In this frame, the attack will be scored as a 10 on the 10-point scale. A date-time stamped output .csv file containing the 'severity' data is saved in the `/project_folder/log` folder.
- `Severity scale 0 -`:

Step 10: Sklearn Visualization

These steps generate visualizations of features and machine learning classification results. This includes images and videos of the animals with prediction overlays, gantt plots, line plots, paths plots and data plots. In this step the different frames can also be merged into video mp4 format.



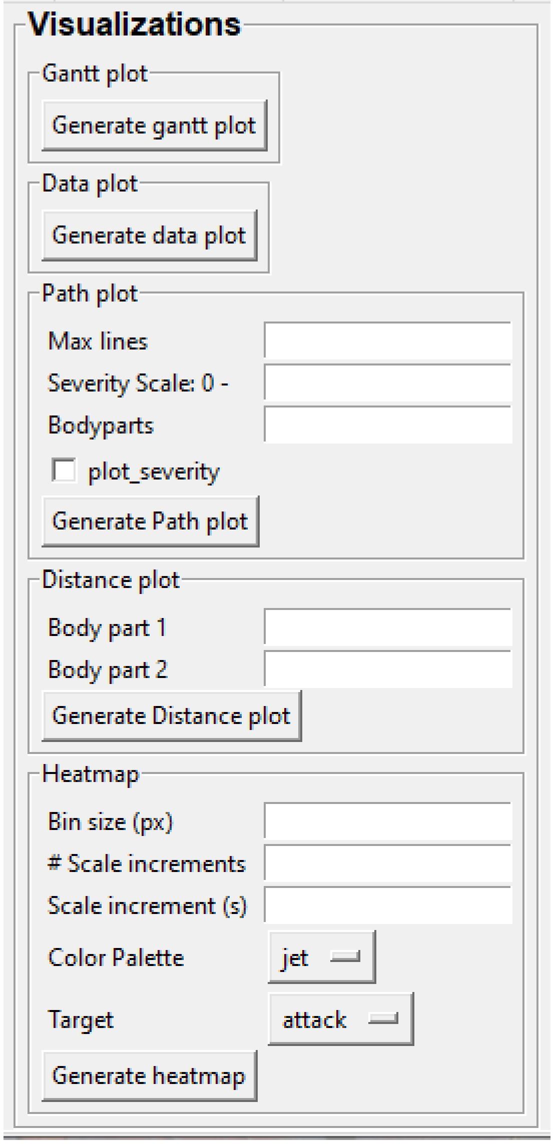
1. Under the **Sklearn visualization** heading, check on the box and click on `Visualize classification results`.
   - `Generate video`: This generates a video of the classification result
   - `Generate frame`: This generates frames(images) of the classification result

Note: Generate frames are required if you want to merge frames into videos in the future.

This step grabs the frames of the videos in the project, and draws circles at the location of the tracked body parts, the convex hull of the animal, and prints the behavioral predictions on top of the frame. For an example, click here

Step 11: Visualizations

The user can also create a range of plots: **gantt plot**, **Data plot**, **Path plot**, **Distance plot**, and **Heatmap**.



Color Palette: jet

Target: attack

Gantt plot

Gantt plot generates gantt plots that display the length and frequencies of behavioral bouts for all the videos in the project.

1. Under the **Gantt plot** heading, click on `Generate Gantt plot` and gantt plot frames will be generated in the `project_folder/frames/output/gantt_plots` folder.

Data plot

Generates 'live' data plot frames for all of the videos in the project that display current distances and velocities.

1. Under the **Data plot** heading, click on `Generate Data plot` and data plot frames will be generated in the `project_folder/frames/output/live_data_table` folder.

Path plot

Generates path plots displaying the current location of the animal trajectories, and location and severity of attack behavior, for all of the videos in the project.

1. Under the **Path plot** heading, fill in the following user defined values.
    - `Max Lines`: Integer specifying the max number of lines depicting the path of the animals. For example, if 100, the most recent 100 movements of animal 1 and animal 2 will be plotted as lines.
    - `Severity Scale`: Integer specifying the scale on which to classify 'severity'. For example, if set to 10, all frames containing attack behavior will be classified from 1 to 10 (see above).
    - `Bodyparts`: String to specify the bodyparts tracked in the path plot. For example, if Nose_1 and Centroid_2, the nose of animal 1 and the centroid of animal 2 will be represented in the path plot.
    - `plot_severity`: Tick this box to include color-coded circles on the path plot that signify the location and severity of attack interactions.
2. Click on `Generate Path plot`, and path plot frames will be generated in the `project_folder/frames/output/path_plots` folder.

Distance plot

Generates distance line plots between two body parts for all of the videos in the project.

1. Fill in the `Body part 1` and `Body part 2`
    - `Body part 1`: String that specifies the the bodypart of animal 1. Eg., Nose_1
    - `Body part 2`: String that specifies the the bodypart of animal 1. Eg., Nose_2
2. Click on `Generate Distance plot`, and the distance plot frames will be generated in the `project_folder/frames/output/line_plot` folder.

Heatmap

Generates heatmap of behavior that happened in the video.

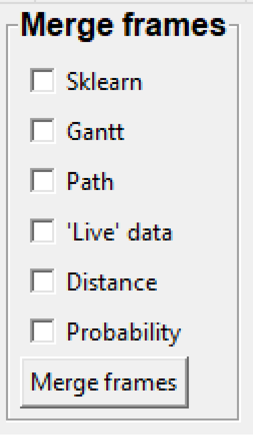
To generate heatmaps, SimBA needs several user-defined variables:

- `Bin size(mm)` : Pose-estimation coupled with supervised machine learning in SimBA gives information on the location of an event at the single pixel resolution, which is too-high of a resolution to be useful in heatmap generation. In this entry box, insert an integer value (e.g., 100) that dictates, in pixels, how big a location is. For example, if the user inserts *100*, and the video is filmed using 1000x1000 pixels, then SimBA will generate a heatmap based on 10x10 locations (each being 100x100 pixels large).
- `max` (integer, or auto): How many color increments on the heatmap that should be generated. For example, if the user inputs *11*, then a 11-point scale will be created (as in the gifs above). If the user inserts auto in this entry box, then SimBA will calculate the ideal number of increments automatically for each video.
- `Color Palette` : Which color pallette to use to plot the heatmap. See the gifs above for different output examples.
- `Target`: Which target behavior to plot in the heatmap. As the number of behavioral target events increment in a specific location, the color representing that region changes.
- `Bodypart`: To determine the location of the event in the video, SimBA uses a single body-part coordinate. Specify which body-part to use here.
- `Save last image only`: Users can either choose to generate a "heatmap video" for every video in your project. These videos contain one frame for every frame in your video. Alternative, users may want to generate a **single image** representing the final heatmap and all of the events in each video - with one png for every video in your project. If you'd like to generate single images, tick this box. If you do not tick this box, then videos will be generated (which is significantly more time-consuming).

2. Click `Generate heatmap` to generate heatmap of the target behavior. For more information on heatmaps based on behavioral events in SimBA - check the tutorial for scenario 2 - visualizing machine predictions

Step 12: Merge Frames

Merge all the generated plots from the previous step into a single frame and generate a **video** as an **output**.



Note: All the frames must be generated in the previous step for this to work. This step combines all the frames(images) that are generated and merge them together and make a video.**

1. Check on the plot that you wish to merge together and output as a single video.
2. Under **Merge Frames**, click `Merge Frames` and frames with all the generated plots will be combined and saved in the `project_folder/frames/output/merged` folder in a video format.

2.3.3 Plotly

Please click here to learn how to use plotly.

CHAPTER

THREE

TUTORIALS

3.1 IBANGS ML Social Behavior Workshop Tutorial

This is a step by step tutorial to start using SimBA to create behavioral classifiers for the IBANGS ML Social Behavior Workshop. For more information about the menus and options and their use cases, see the SimBA Scenario tutorials

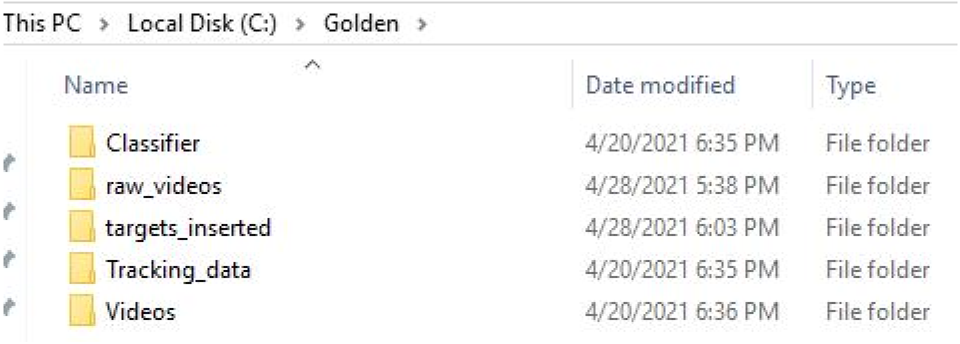
3.1.1 Installation

We recommend using anaconda and install simba in an anaconda environment. Please click here to learn how to install SimBA.

3.1.2 Logistics prior to workshop start

Prior to the workshop please familiarize yourself with the folder structure provided.

Note: If you are in this workshop, the files are in **C:\Golden**. There is a shortcut to the folder structure located on the desktop named **SimBA-IBANG**, double click on it to get access to **C:\Golden**.



This PC > Local Disk (C:) > Golden >

Name	Date modified	Type
Classifier	4/20/2021 6:35 PM	File folder
raw_videos	4/28/2021 5:38 PM	File folder
targets_inserted	4/28/2021 6:03 PM	File folder
Tracking_data	4/20/2021 6:35 PM	File folder
Videos	4/20/2021 6:36 PM	File folder

- `Classifier` : This folder contains pre-trained classifiers from the Golden Lab (*Attack.sav*, *Defensive.sav*) and the classifier settings used for training (*model_settings.csv*).
- `raw_videos` : This folder contains preprocessed videos that will be used in **Part 1**
- `Videos` : This folder contains the videos that will be used in **Part 2** to train a classifier.
- `Tracking_data` : This folder contains the tracking data of the videos that are generated using DeepLabCut.
- `targets_inserted`: This folder contains our hand annotated csvs for training a classifier in **Part 2**

Please click here to download our sample data to follow through this tutorial.

3.1.3 Starting simba

We will be using miniconda in this workshop tutorial and the environment has been set up to use.

1. Open up command prompt, type in `conda activate simbaenv` and press `"Enter"`.
2. Type `simba` and press `"Enter"`.

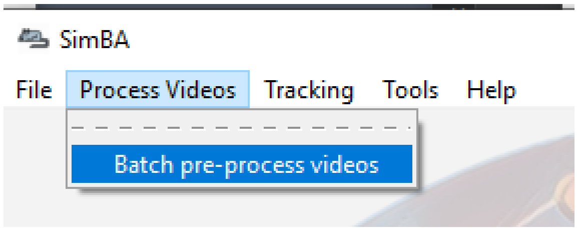
3.1.4 Part 1: Video Preprocessing with SimBA

This step will preprocess the videos for SimBA to analyze. There are multiple ways you can do this:

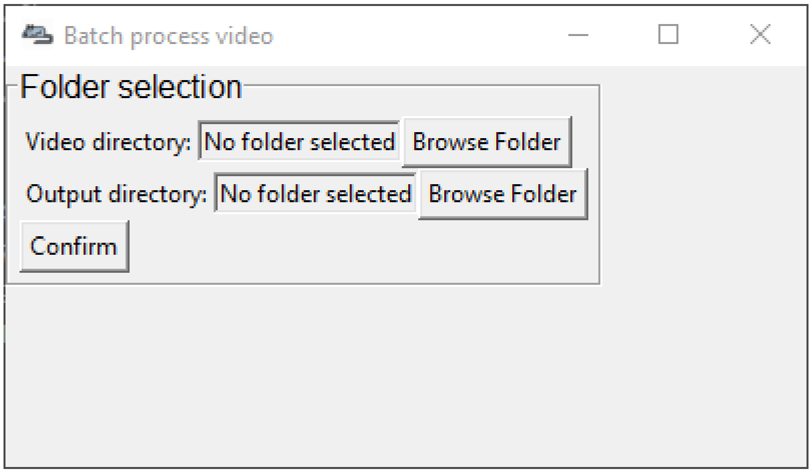
- Built-in Tools in SimBA
- Batch videos processing

In this workshop we are going to use batch videos pipeline.

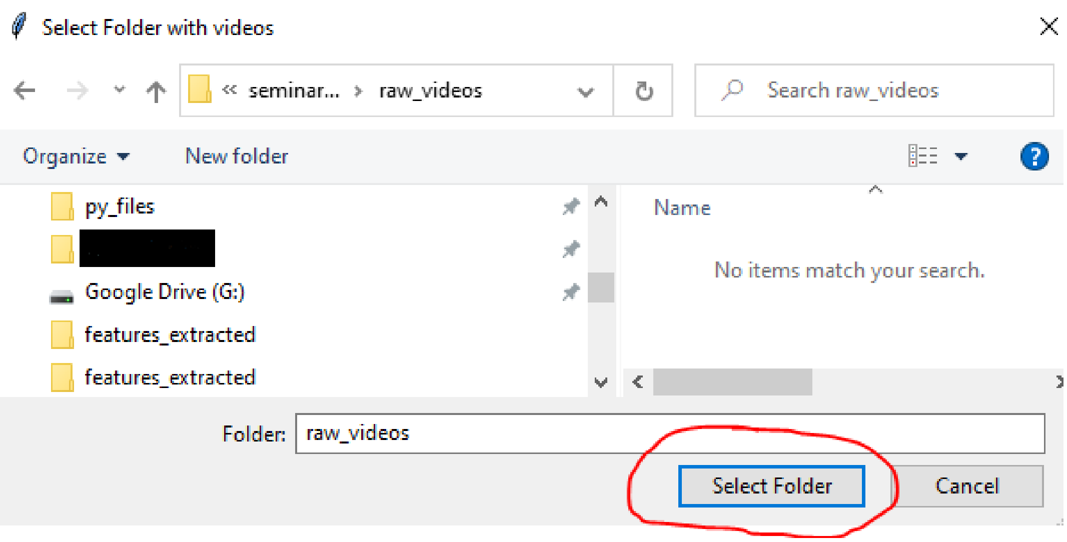
Step 1: Folder Selection



1. In the main SimBA window click on `Process Videos` –> `Batch pre-process videos`. The window shown below will display.



2. Under **Folder Selection** heading and next to `Video directory`, click on `Browse Folder` and navigate to **C:/Golden/raw_videos** and click on `Select Folder`.



Folder: raw_videos

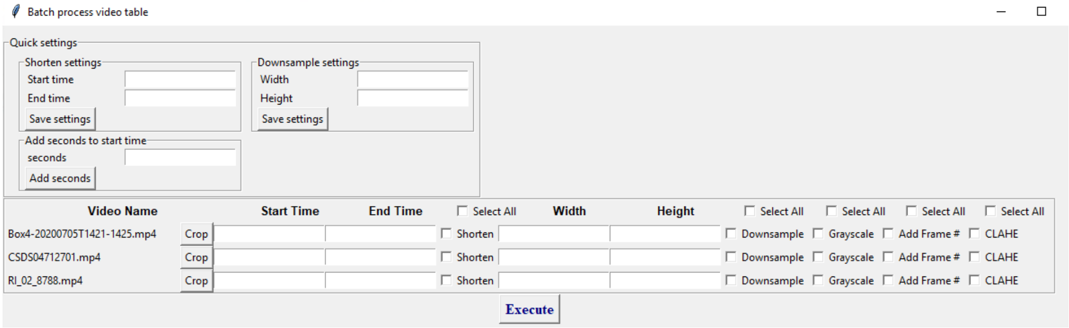
3. Next to `Output Directory`, click on `Browse Folder` and create a folder **C:/Golden/output** that should store the processed videos and click on `Select Folder`.
4. Click to `Confirm` the two selected directories.

Warning:

- Please make sure there are no spaces in your folder names or video names, use underscores if needed.
- If you select the same video directory as the output directory, your video will be **overwritten**. Best practice is to select a different output directory.

Step 2: Select parameters

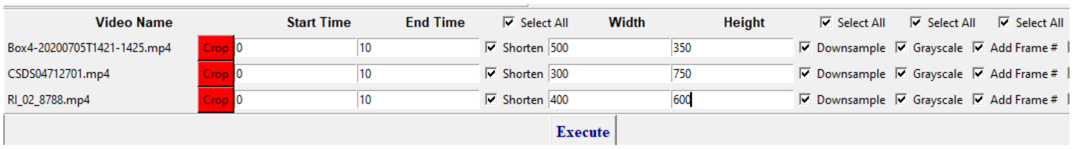
1. Once you select `Confirm`, a table listing the videos in the `Video directory` folder will auto-generate and display. This sequence of videos will now be processed with the parameters you selected, starting with the cropping function.



Video Name		Start Time	End Time	☐ Select All	Width	Height	☐ Select All	☐ Select All	☐ Select All	☐ Select All
Box4-20200705T1421-1425.mp4	Crop			☐ Shorten			☐ Downsample	☐ Grayscale	☐ Add Frame #	☐ CLAHE
CSDS04712701.mp4	Crop			☐ Shorten			☐ Downsample	☐ Grayscale	☐ Add Frame #	☐ CLAHE
RI_02_8788.mp4	Crop			☐ Shorten			☐ Downsample	☐ Grayscale	☐ Add Frame #	☐ CLAHE

2. To crop your videos, click on the `Crop` button. A frame from the video will pop up. Left click on the frame and drag the rectangular bounding box to mark the region of the video you wish to keep. Once the rectangle is marked, double tap "Enter" on your keyboard. *(The relevant crop button will turn red once you have selected to crop a video)*

3. Please refer to the image below as input parameters:



Video Name		Start Time	End Time	☑ Select All	Width	Height	☑ Select All	☑ Select All	☑ Select All
Box4-20200705T1421-1425.mp4	Crop	0	10	☑ Shorten	500	350	☑ Downsample	☑ Grayscale	☑ Add Frame #
CSDS04712701.mp4	Crop	0	10	☑ Shorten	300	750	☑ Downsample	☑ Grayscale	☑ Add Frame #
RI_02_8788.mp4	Crop	0	10	☑ Shorten	400	600	☑ Downsample	☑ Grayscale	☑ Add Frame #

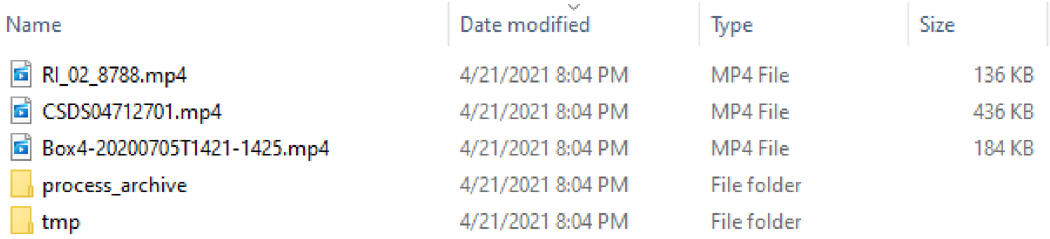
- To trim a specific video, check the `Shorten` box and enter the **Start Time** and **End Time** in the **HH:MM:SS** format. In our case, try to trim all the videos to 10 seconds by entering 0 for `Start Time` and 10 for `End time` .
- To downsample a specific video, check the `Downsample` box and enter the **Width** and **Height** in pixels as shown in the image.
- To change a specific video to grayscale, check the `Grayscale` box.
- To superimpose the specific frame numbers onto the frames of the video, check on the `Add Frame #` box. For an example output video with frame numbers overlaid, click here
- We are not going to do this as it is computationally expensive but to apply CLAHE, check on the `CLAHE` box. For more information on CLAHE, click here

4. In the first row of the table, there are `Select All` checkboxes. Use these checkboxes to apply a manipulation to all of the videos in the folder.

Note: We know that the *Select All* checkbox might be slightly off position in the table. We will work on fixing it soon.

Step 3: Execute

1. Once all the parameters are set, click on `Execute`.
2. The final output videos will be saved in the `Output Directory` that you selected in *Step 1*.



Name	Date modified	Type	Size
RI_02_8788.mp4	4/21/2021 8:04 PM	MP4 File	136 KB
CSDS04712701.mp4	4/21/2021 8:04 PM	MP4 File	436 KB
Box4-20200705T1421-1425.mp4	4/21/2021 8:04 PM	MP4 File	184 KB
process_archive	4/21/2021 8:04 PM	File folder	
tmp	4/21/2021 8:04 PM	File folder	

3. A subfolder in the `Output Directory` called **tmp** will contain the step-by-step processed videos.
4. The **process_archive** folder contains a **.txt** file that lists the processes that were run.
5. The **Output Directory** will contain all the final processed videos.

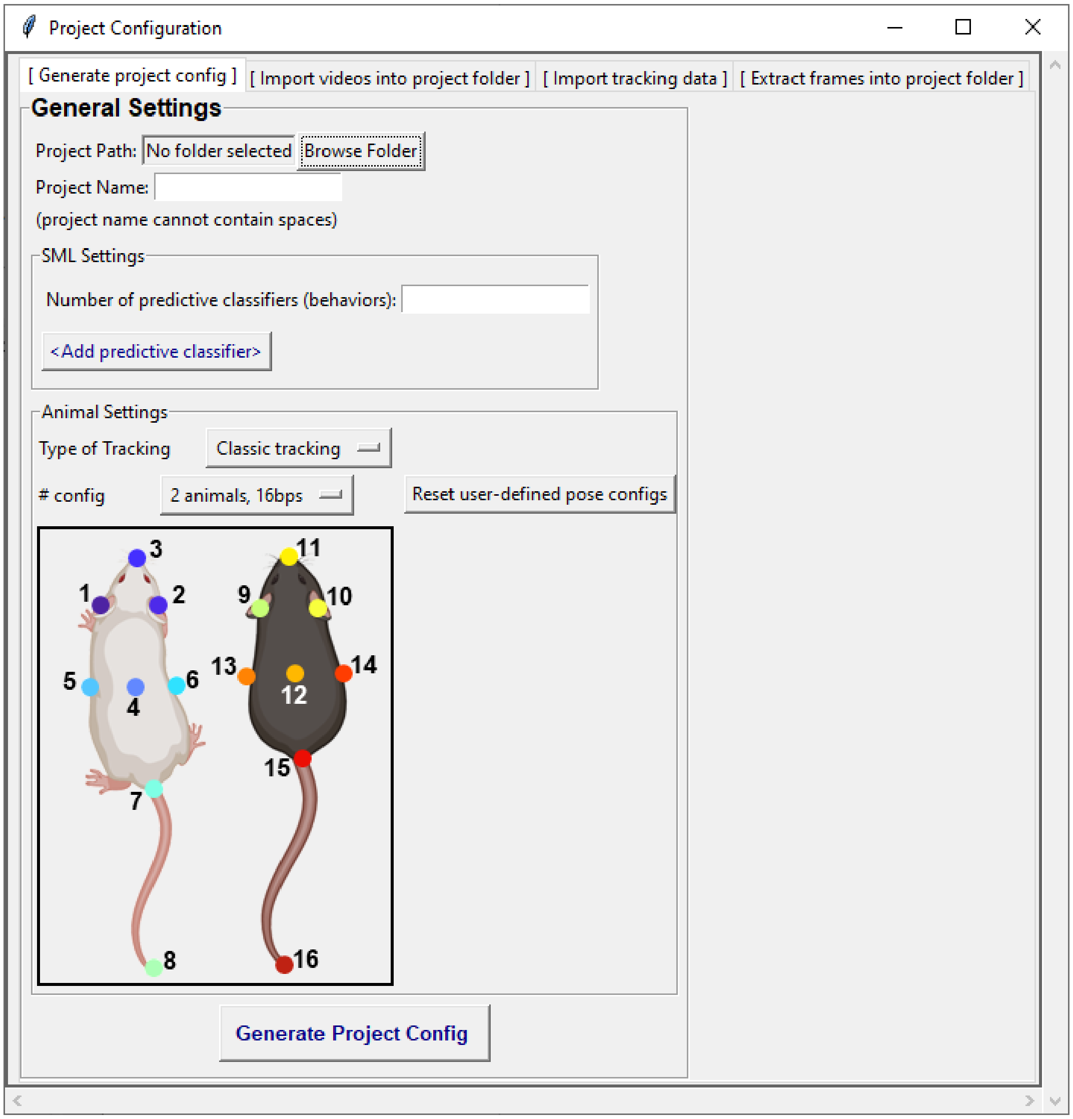
3.1.5 Part 2: Create Project

Now that we have the videos in the optimal format, we can start using SimBA. In this part, we are going to create an “Attack” classifier.

Step 1: Generate Project Config

In this step you create your main project folder which will contain all of the required sub-directories.

1. In the main SimBA window, click on `File` and `Create a new project`. The following windows will pop up.


Project Configuration
[Generate project config]
[Import videos into project folder]
[Import tracking data]
[Extract frames into project folder]
Browse Folder
<Add predictive classifier>
Reset user-defined pose configs
3
1
2
5
4
6
7
8
11
9
10
13
12
14
15
16
Generate Project Config

General Settings

Project Path: No folder selected

Project Name:

(project name cannot contain spaces)

SML Settings

Number of predictive classifiers (behaviors):

Animal Settings

Type of Tracking: Classic tracking

config: 2 animals, 16bps

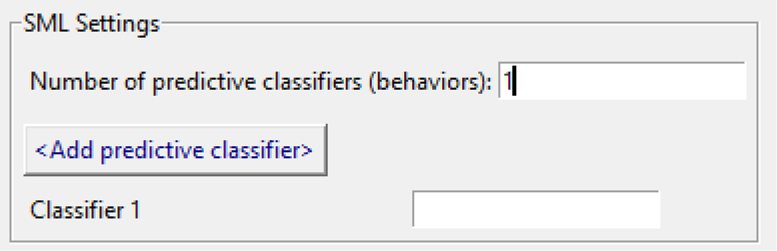
2. Navigate to the `[ Generate project config ]` tab. Under **General Settings**, click on the `Browse Folder` on `Project Path` and select **C:/Golden** and click `Select Folder`

3. `Project Name` is the name of your project, in this case type in **ibangs_attack**

> **Note:** *Keep in mind that the project name cannot contain spaces. Instead use underscore “_”*

4. Under *SML Settings*, put in the number of predictive classifiers that you wish to create. In this case, put **1**.

5. Click `Add Classifier` and it creates a row as shown in the following image. In each entry box, fill in the name of the behavior that you want to classify. In this case type **Attack**.



SML Settings

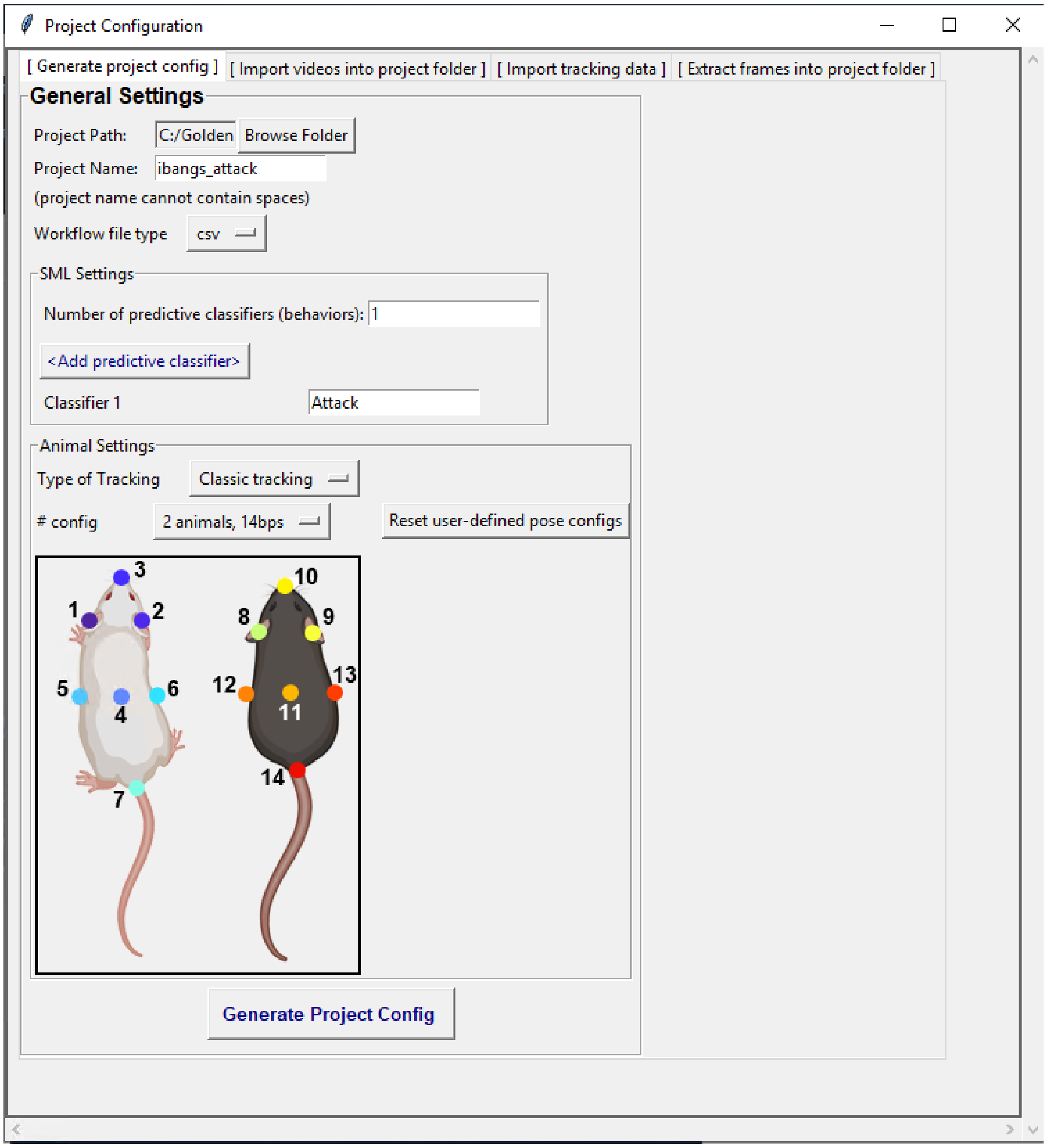
Number of predictive classifiers (behaviors): 1

Classifier 1

6. `Type of Tracking` allows the user to choose (same color) multi-animal tracking or the classic tracking, and we are going to select `classic tracking`.

7. `Animal Settings` is the number of animals and body parts that that the pose estimation tracking data contains. The default for **SimBA** is 2 animals and 16 body parts (`2 animals, 16bp`). There are a few other - ** yet not validated** - options, accessible in the dropdown menu. Here, we are going to select `2 animals, 14bp`.

8. Your windows should look like the image as shown below.



Project Configuration

General Settings

Project Path: C:/Golden

Project Name: ibangs_attack

(project name cannot contain spaces)

Workflow file type csv

SML Settings

Number of predictive classifiers (behaviors): 1

Classifier 1 Attack

Animal Settings

Type of Tracking Classic tracking

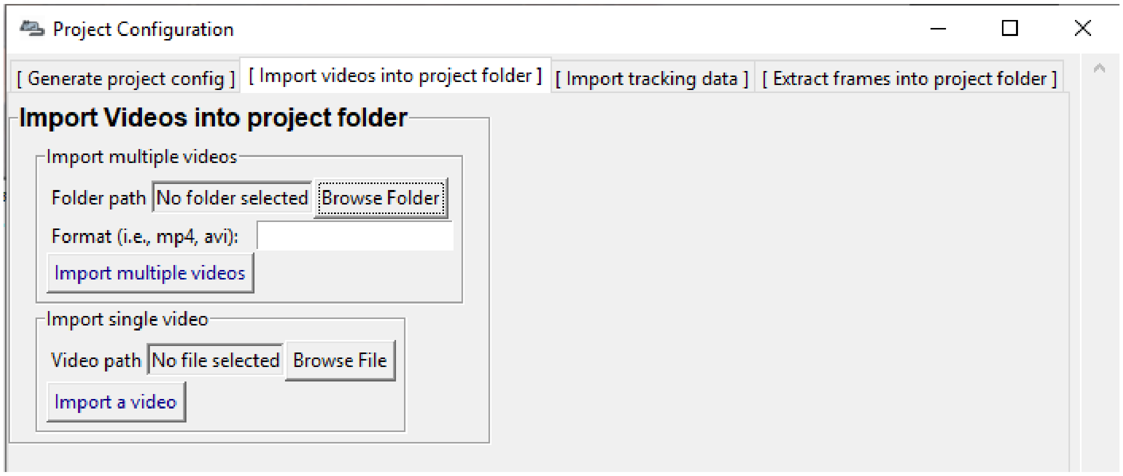
config 2 animals, 14bps

[Generate project config]
[Import videos into project folder]
[Import tracking data]
[Extract frames into project folder]
Browse Folder
<Add predictive classifier>
Reset user-defined pose configs
3
1
2
5
6
4
7
10
8
9
12
13
11
14
Generate Project Config

8. Click on `Generate Project Config` to generate your project. The project folder will be located in **C:/Golden/ibangs_attack**.

Step 2: Import videos into project folder for training

In this step, there are options to import either one or multiple videos. The imported videos are used for visualizing predictions and standardizing distances across videos by calculating metric distances from pixel distances. We are going to import **multiple** videos.



Import multiple videos

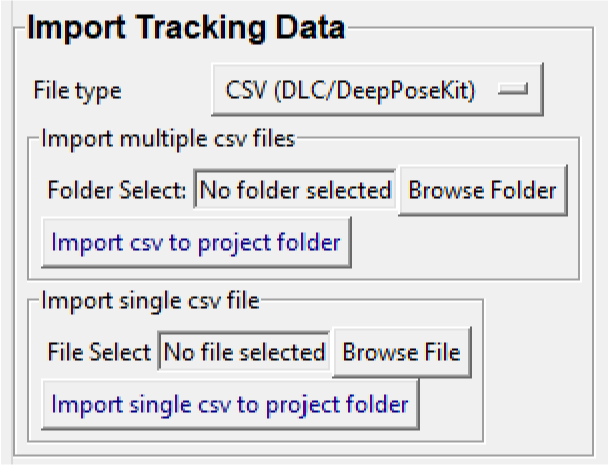
1. Navigate to the `[ Import videos into project folder ]` tab.
2. Under the `Import multiple videos` heading, click on `Browse Folder` to select a folder that contains all the videos that you wish to import into your project. In our case, we go to **C:/Golden/Videos**
3. Enter the file type of your videos, `mp4` in the `Video format` entry box.
4. Click on `Import multiple videos`.
5. The videos will be imported into **C:/Golden/ibangs_attack/project_folder/videos**

Step 3: Import Tracking Data

In this step, you will import your pose-estimation tracking data. We support tracking from most common pose estimation platforms and these can be imported from: click here In this workshop, we are going to use DeepLabCut's tracking data and they will be in **.csv** format.

For more information about using DeepLabCut to generate tracking data:

- DeepLabCut Github page
- DeepLabCut Preprint
- DeepLabCut Community on Gitter



Import Tracking Data

File type: CSV (DLC/DeepPoseKit)

Folder Select: No folder selected

File Select: No file selected

Import multiple csv files
Browse Folder
Import csv to project folder
Import single csv file
Browse File
Import single csv to project folder

Import tracking data (.csv)

1. Navigate to the `[ Import tracking data ]` tab. Under the `Import tracking data` click on the `File type` drop down menu.
2. From the drop down menu, .csv files = `CSV (DLC/DeepPoseKit)`.
3. To import multiple files, choose the folder that contains the files by clicking `Browse Folder`. In our case, go to *C:/Golden/Tracking_data*, then click `Import csv to project folder`.
4. The csvs will be located in **C:/Golden/ibangs_attack/project_folder/csv/input_csv**

3.1.6 Part 3: Load project

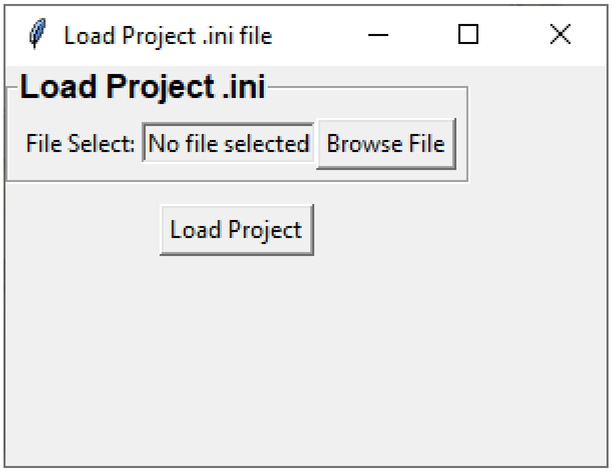
To use SimBA, you will load the project you have created.

Step 1: Load Project Config

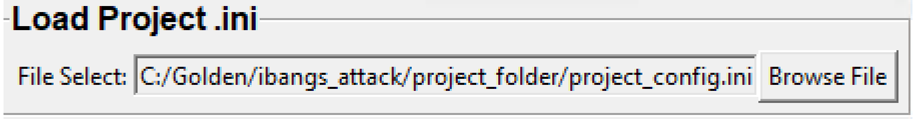
In this step you will load the *project_config.ini* file that was created. It is located in **C:/Golden/ibangs_attack/project_folder**

Note: A project_config.ini should always be loaded before any other process.

1. In the main SimBA window, click on `File` and `Load project`. The following windows will pop up.



2. Click on `Browse File`. Then, go to **C:/Golden/ibangs_attack/project_folder** and click on *project_config.ini* file. Once this step is completed, it should look like the following, and you should no longer see the text **No file selected**.

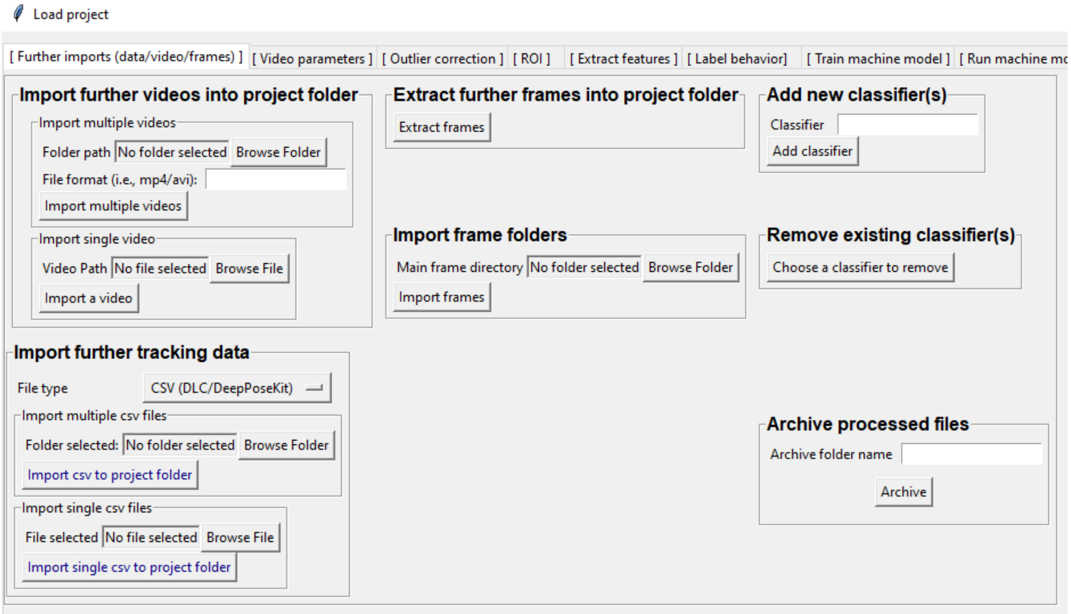


File Select: C:/Golden/ibangs_attack/project_folder/project_config.ini

In this image, you can see the **C:/Golden** is my selected working directory, **ibangs_attack** is my project name, and the last two sections of the folder path is always going to be **project_folder/project_config.ini**.

3. Click on `Load Project`.

4. The following window will pop out.

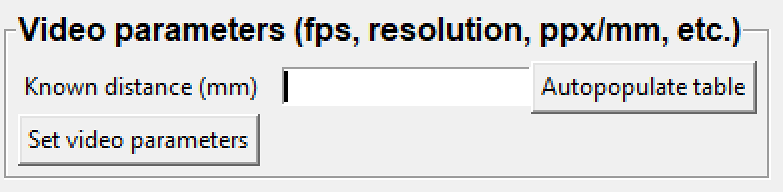


Step 2: Set video parameters

Meta data such as FPS and resolution are automatically pulled for your videos.

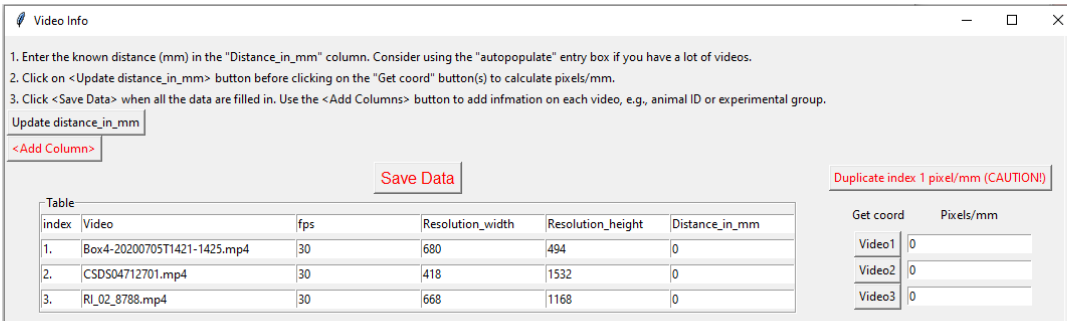
In order to standardize your arena size, you will specify a known distance in your videos. SimBA then calculates the pixels per millimeter for each video.

You will be using a tool that requires the known distance between two points (e.g., the cage width or the cage height) in order to calculate **pixels per millimeter**. The real life distance between the two points is called `Distance in mm`.



1. Click on the **[Video parameters]** tab, under **Set video parameters(distances,resolution,etc.)**, the entry box named `Known distance(mm)` is the known distance between two points in the videos in millimeters. If the known distance is the same in all the videos in the project, then enter the value *(e.g,: 245)* and click on `Autopopulate table`. and it will auto-populate the table in the next step (see below). If you leave the *Known distance(mm)* entry box empty, the known distance will default to zero and you will fill in the value for each video individually.

2. In this case, we are going to leave `Known distance (mm)` empty and click on `Set Video Parameters`. The following will pop up.



1. Enter the known distance (mm) in the "Distance_in_mm" column. Consider using the "autopopulate" entry box if you have a lot of videos.
2. Click on <Update distance_in_mm> button before clicking on the "Get coord" button(s) to calculate pixels/mm.
3. Click <Save Data> when all the data are filled in. Use the <Add Columns> button to add infmation on each video, e.g., animal ID or experimental group.

Table

index	Video	fps	Resolution_width	Resolution_height	Distance_in_mm
1.	Box4-20200705T1421-1425.mp4	30	680	494	0
2.	CSDS04712701.mp4	30	418	1532	0
3.	RI_02_8788.mp4	30	668	1168	0

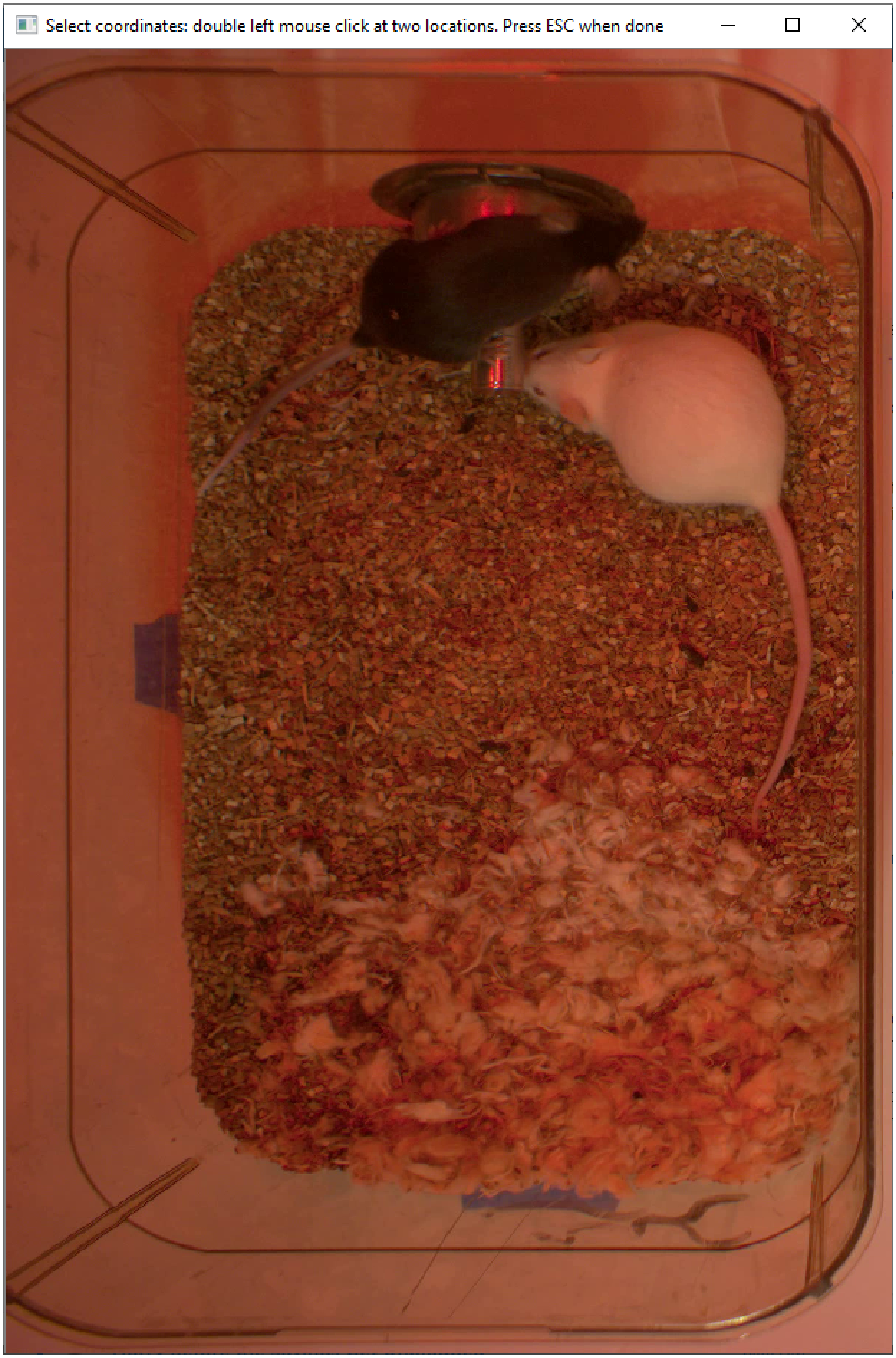
Get coord	Pixels/mm
Video1	0
Video2	0
Video3	0

3. We have imported 3 videos and their names are listed on the `Video` column. In our case, the `Distance in mm` for the videos are the following:

   - **Box4-20200705T1421-1425** = `190`
   - **RI_02_8788** = `190`
   - **CSDS04712701** = `127`

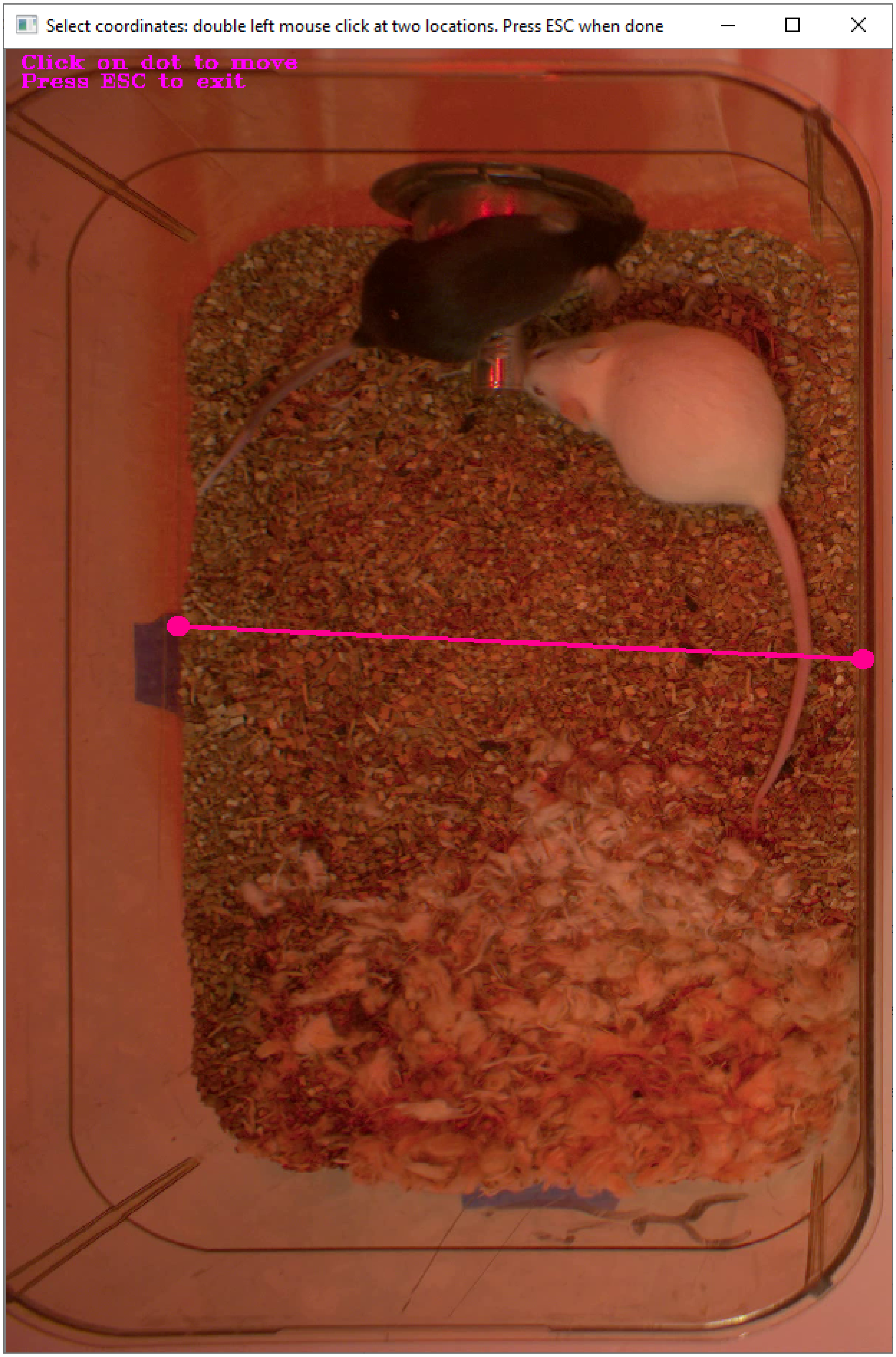
4.Enter the values in the entry boxes sand click on `Update distance_in_mm` and this will update the whole table.

5. Next, to get the `Pixels/mm` for the first video, click on `Video1` and the following window will pop up. The window that pops up displays the first frame of `Video1`.



Select coordinates: double left mouse click at two locations. Press ESC when done

6. Now, double **left** click to select two points that defines the known distance in real life. In this case click on width of the box for the ones in cages (RI_02_8788, CSDS04712701), and the height for the operant chamber(Box4-20200705T1421-1425)

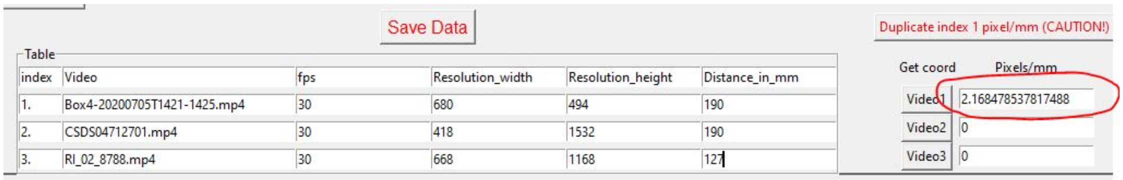


Select coordinates: double left mouse click at two locations. Press ESC when done

Click on dot to move
Press ESC to exit

7. If you misplaced one or both of the dots, you can double click on either of the dots to place them somewhere else in the image. Once you are done, hit `Esc`.

8. If every step is done correctly, the Pixels/mm column in the table should populate with the number of pixels that represent one millimeter,



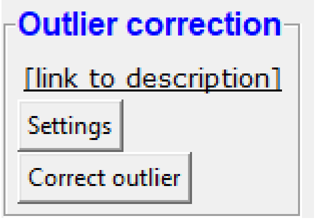
index	Video	fps	Resolution_width	Resolution_height	Distance_in_mm
1.	Box4-20200705T1421-1425.mp4	30	680	494	190
2.	CSDS04712701.mp4	30	418	1532	190
3.	RI_02_8788.mp4	30	668	1168	127

Get coord	Pixels/mm
Video1	2.168478537817488
Video2	0
Video3	0

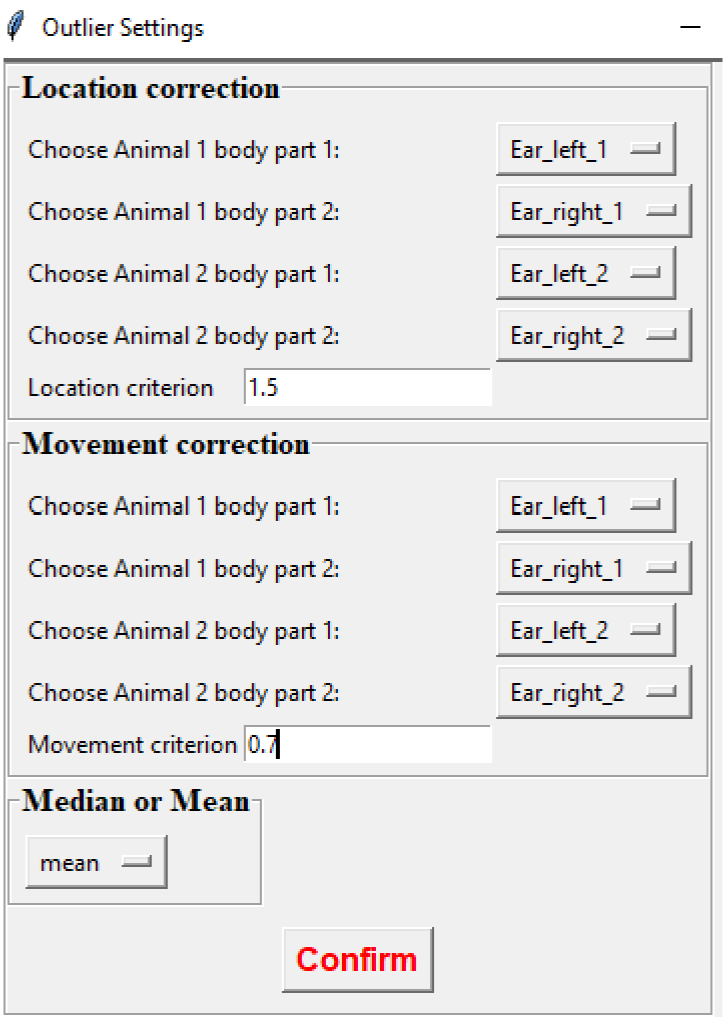
9. Repeat the steps for every video in the table, and once it is done, click on Save Data. This will generate a csv file named **video_info.csv** in /project_folder/log folder that contains a table with your video meta data.

Step 3: Outlier Correction

Outlier correction is used to correct gross tracking inaccuracies by detecting outliers based on movements and locations of body parts in relation to the animal body length. For more details, please click here



1. Under Outlier correction click on Settings, and the following window will pop up. The image shows the settings we used in our lab.



Outlier Settings

Location correction

Choose Animal 1 body part 1: Ear_left_1

Choose Animal 1 body part 2: Ear_right_1

Choose Animal 2 body part 1: Ear_left_2

Choose Animal 2 body part 2: Ear_right_2

Location criterion 1.5

Movement correction

Choose Animal 1 body part 1: Ear_left_1

Choose Animal 1 body part 2: Ear_right_1

Choose Animal 2 body part 1: Ear_left_2

Choose Animal 2 body part 2: Ear_right_2

Movement criterion 0.7

Median or Mean

mean

2. Make sure your settings matches the image. Enter `1.5` for `Location criterion` and `0.7` for `Movement criterion`
3. For `Median or Mean` select `mean`, and hit `Confirm`.
4. Click `Run outlier correction`.
5. This step will correct the outliers and store the new csvs in **C:/Golden/ibangs_attack/project_folder/csv/outlier_corrected_move** If you are confident with your tracking data you can skip outlier correction by clicking `Skip outlier correction (CAUTION)`, but we do not recommend it.

Step 4: Extract Features

Based on the coordinates of body parts in each frame - and the frame rate and the pixels per millimeter values - the feature extraction step calculates a large set of features per frame (~500) used for behavioral classification. Features are values such as metric distances between body parts, angles, areas, movement, paths, and their deviations and rank in individual frames and across rolling windows. This set of features will depend on the body-parts tracked during pose-estimation (which is defined when creating the project).

- Click here for an example list of features when tracking 2 mice and 16 body parts.
- Click here for example list of features for user defined pose config.
- Click here for user-defined feature extraction script.

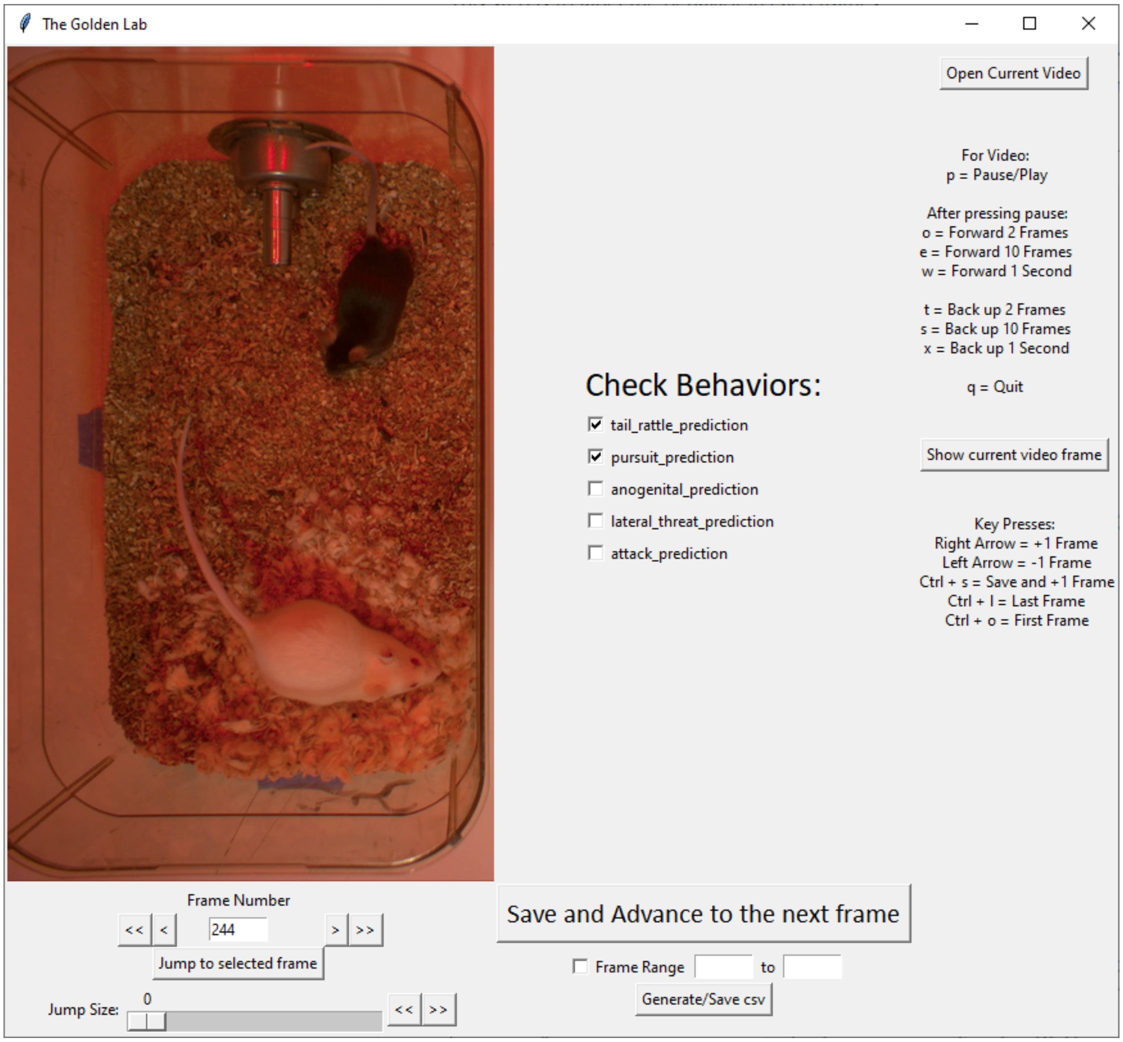
1. Click on `Extract Features`.

Step 5: Label Behavior

This step is used to label the behaviors in each frames of a video. This data will be concatenated with the features and used for creating behavioral classifiers.

There are two options, one is to start a **new video annotation** and one is to **continue on where you last left off**. Both are essentially the same, except the latter will start with the frame where you last saved. For example, one day, you started a new video by clicking `Select video (create new video annotation)` and you feel tired and sick of annotating the videos. You can now click `Generate/Save` button to save your work for your coworker to continue. Your coworker can continue by clicking ` Select folder with frames(continue existing video annotation)` and select the the video folder that you have annotated half way and take it from there!

1. Click on `Select video`. In your project folder navigate to the `/project_folder/videos/` folder, and you should select the videos that you wish to annotate.



For Video:
p = Pause/Play

After pressing pause:
o = Forward 2 Frames
e = Forward 10 Frames
w = Forward 1 Second

t = Back up 2 Frames
s = Back up 10 Frames
x = Back up 1 Second

q = Quit

Check Behaviors:

☑ tail_rattle_prediction
☑ pursuit_prediction
☐ anogenital_prediction
☐ lateral_threat_prediction
☐ attack_prediction

Key Presses:
Right Arrow = +1 Frame
Left Arrow = -1 Frame
Ctrl + s = Save and +1 Frame
Ctrl + l = Last Frame
Ctrl + o = First Frame

Frame Number 244

☐ Frame Range to

Jump Size: 0

2. Please click here to learn how to use the behavior annotation interface.

3. Once finished, click on `Generate/Save` and it will generate a new *.csv* file in **/csv/targets_inserted** folder.

Step 6: Train Machine Model

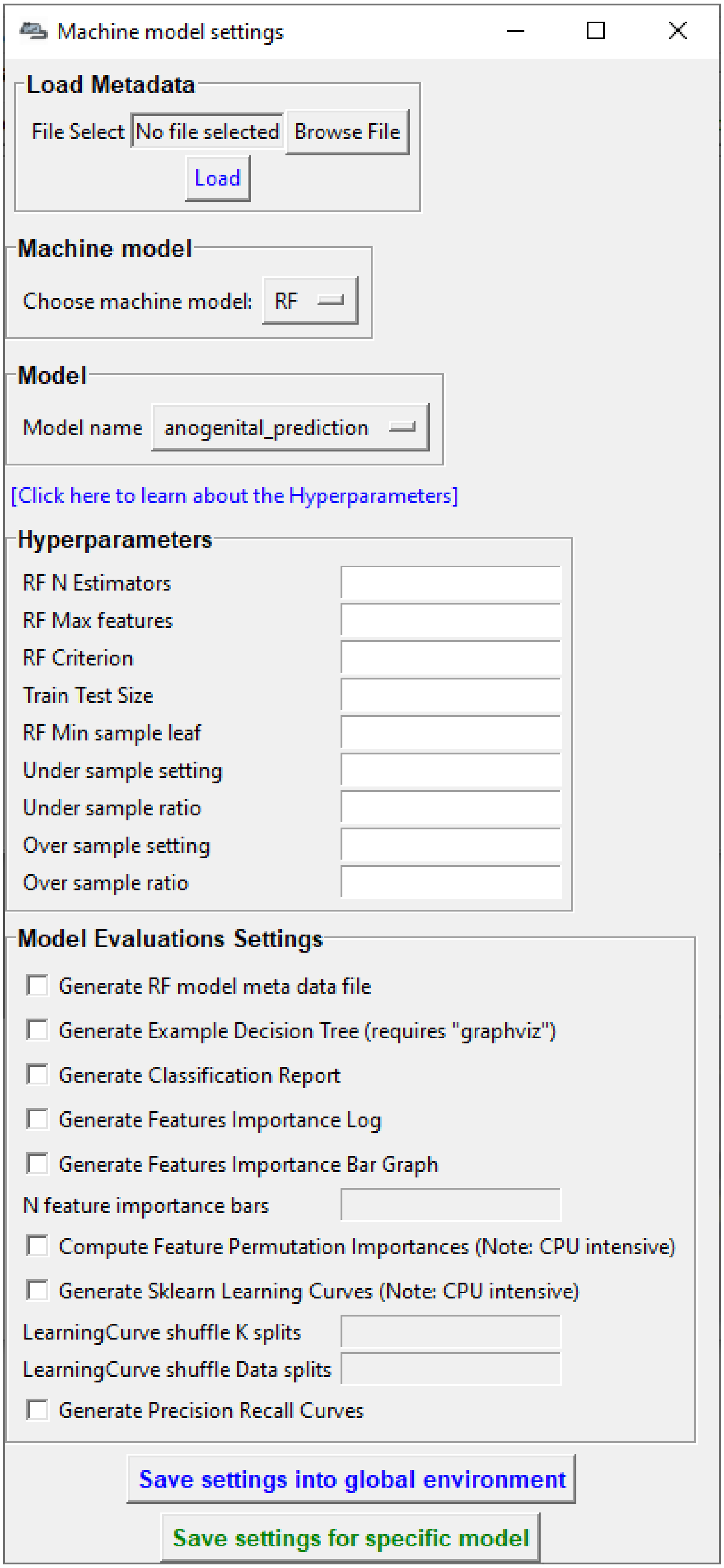
This step is used for training new machine models for behavioral classifications.

Note: If you import existing models, you can skip this step and go straight to **Step 8** to run machine models on new video data.

Train single model

For more details on the training parameters please click here

1. Click on `Settings` and the following window will pop up.



Machine model settings

Load Metadata

File Select: No file selected [Browse File]

[Load]

Machine model

Choose machine model: RF

Model

Model name: anogenital_prediction

[Click here to learn about the Hyperparameters]

Hyperparameters

RF N Estimators	
RF Max features	
RF Criterion	
Train Test Size	
RF Min sample leaf	
Under sample setting	
Under sample ratio	
Over sample setting	
Over sample ratio	

Model Evaluations Settings

☐ Generate RF model meta data file

☐ Generate Example Decision Tree (requires "graphviz")

☐ Generate Classification Report

☐ Generate Features Importance Log

☐ Generate Features Importance Bar Graph

N feature importance bars

☐ Compute Feature Permutation Importances (Note: CPU intensive)

☐ Generate Sklearn Learning Curves (Note: CPU intensive)

LearningCurve shuffle K splits

LearningCurve shuffle Data splits

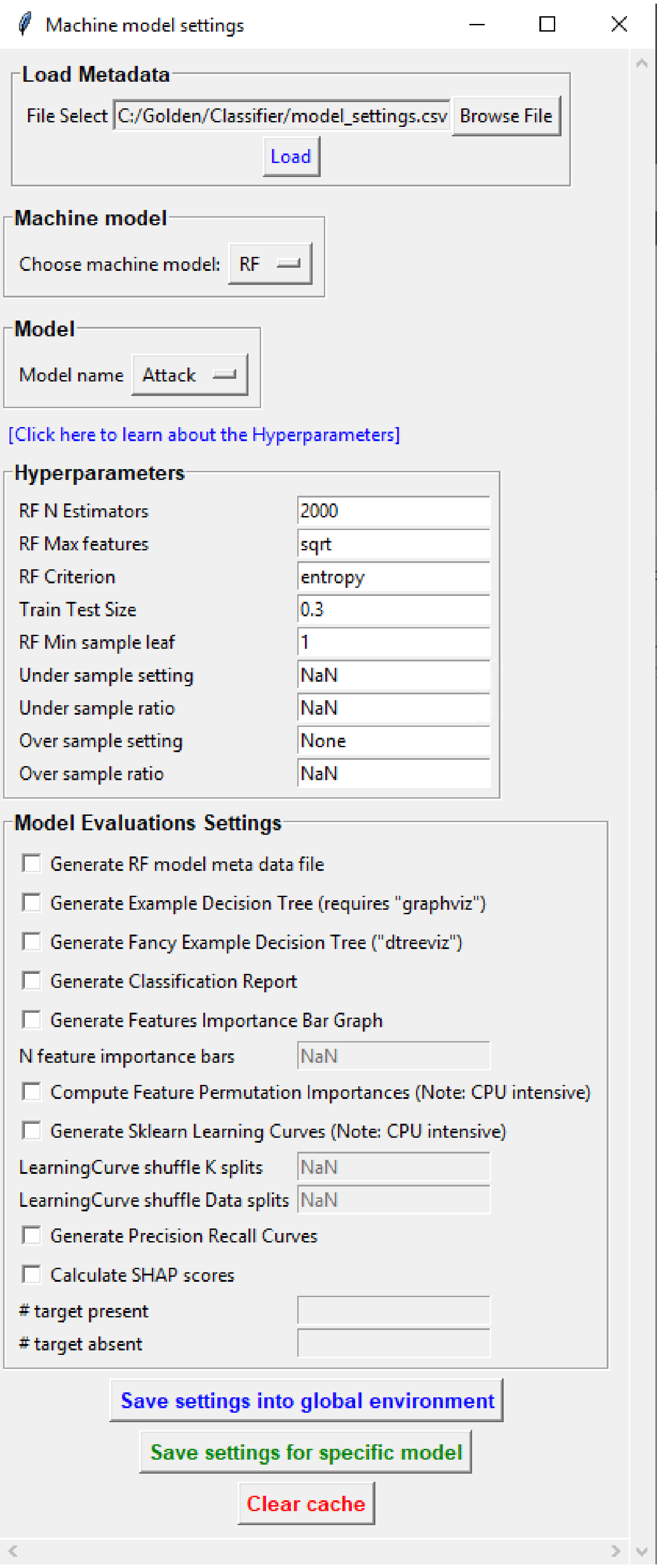
☐ Generate Precision Recall Curves

[Save settings into global environment]

[Save settings for specific model]

Note: If you have a .csv file containing hyper-parameter meta data, you can import this file by clicking on `Browse File` and then click on `Load`. This will autofill all the hyper-parameter entry boxes and model evaluation settings.

2. Under `Load Metadata`, click on `Browse File` and navigate to *C:/Golden/Classifier* and select the *model_settings.csv* and click `Load`. It will fill up the information as shown as the image below.



Machine model settings

Load Metadata

File Select C:/Golden/Classifier/model_settings.csv

Machine model

Choose machine model: RF

Model

Model name Attack

[Click here to learn about the Hyperparameters]

Hyperparameters

RF N Estimators	2000
RF Max features	sqrt
RF Criterion	entropy
Train Test Size	0.3
RF Min sample leaf	1
Under sample setting	NaN
Under sample ratio	NaN
Over sample setting	None
Over sample ratio	NaN

Model Evaluations Settings

☐ Generate RF model meta data file

☐ Generate Example Decision Tree (requires "graphviz")

☐ Generate Fancy Example Decision Tree ("dtreeviz")

☐ Generate Classification Report

☐ Generate Features Importance Bar Graph

N feature importance bars NaN

☐ Compute Feature Permutation Importances (Note: CPU intensive)

☐ Generate Sklearn Learning Curves (Note: CPU intensive)

LearningCurve shuffle K splits NaN

LearningCurve shuffle Data splits NaN

☐ Generate Precision Recall Curves

☐ Calculate SHAP scores

target present

target absent

Browse File
Load
Save settings into global environment
Save settings for specific model
Clear cache

3. Click on the `Save settings into global environment` button to save your settings into the *project_config.ini* file and use the settings to train a single model.

4. Alternatively, click on the `Save settings for specific model` button to save the settings for one model. To generate multiple models - for either multiple different behaviors and/or using multiple different hyper-parameters - re-define the Machine model settings and click on `Save settings for specific model` again. Each time the `Save settings for specific model` is clicked, a new config file is generated in the */project_folder/configs* folder. In the next step (see below), a model for each config file will be created if pressing the **Train multiple models, one for each saved settings** button.

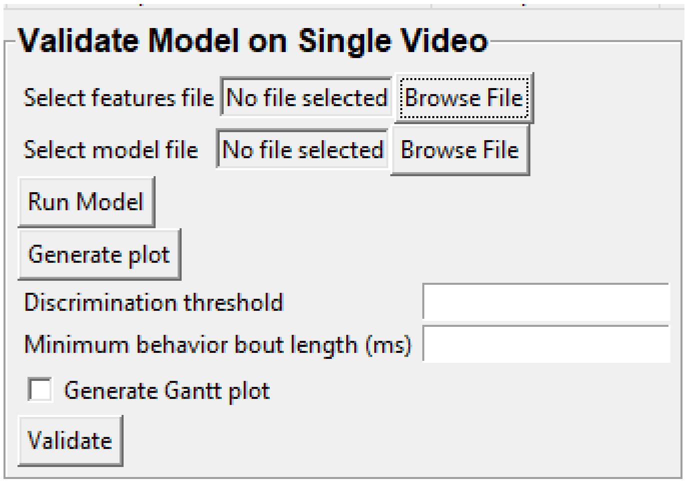
5. Due to time constraint, we will replace the csvs in the **C:/Golden/ibangs_attack/project_folder/csv/targets_inserted** with the Golden Lab annotated csvs in **C:/Golden/targets_inserted**.

6. Then click on `Train single model from global environment`.

Step 7: Model Validation before running machine model on new data

The user can validate each model *(saved in .sav format)* file. In this validation step the user specifies the path to a previously created model in .sav file format, and a .csv file containing the features extracted from a video. This process will (i) run the classifications on the video, and (ii) create a video with the predictions overlaid together with a gantt plot showing predicted behavioral bouts. Click here for an example validation video.

1. Under **[Run machine model]** tab –> **validate Model on Single Video**, select a features file (Box4-20200705T1421-1425.csv). It is located in `project_folder/csv/features_extracted`.

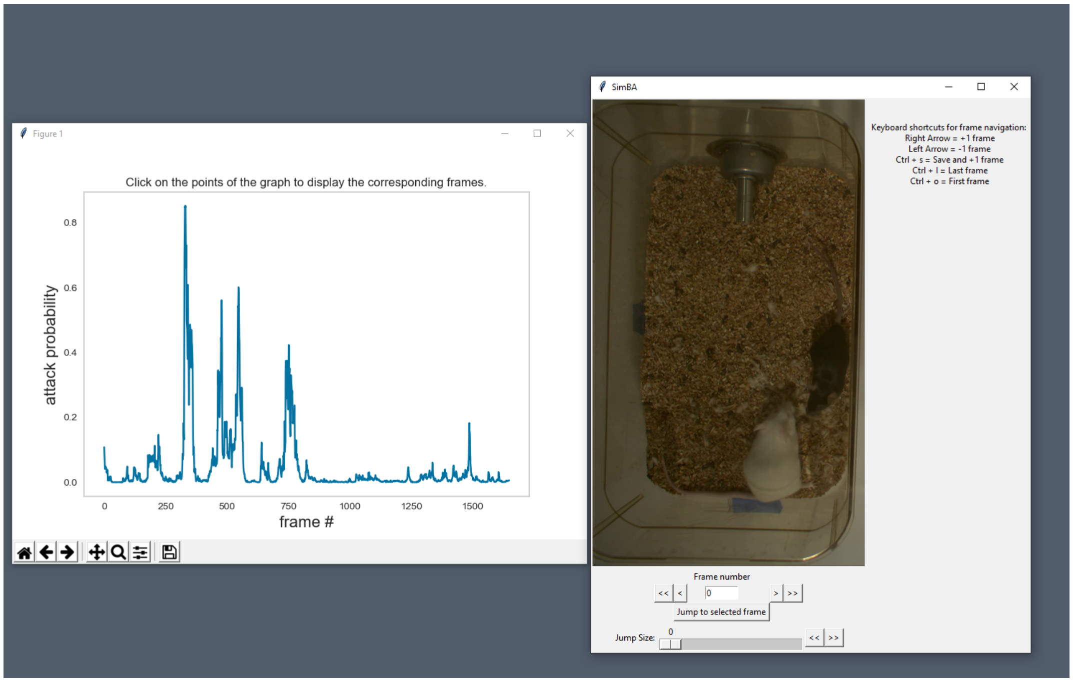


2. Under `Select model file`, click on `Browse File` to select the *Attack.sav file* from **C:/Golden/Classifier**.

3. Click on `Run Model`.

4. Once, it is completed, it should print *"Predictions generated."*, now you can click on `Generate plot`. A graph window and a frame window will pop up.

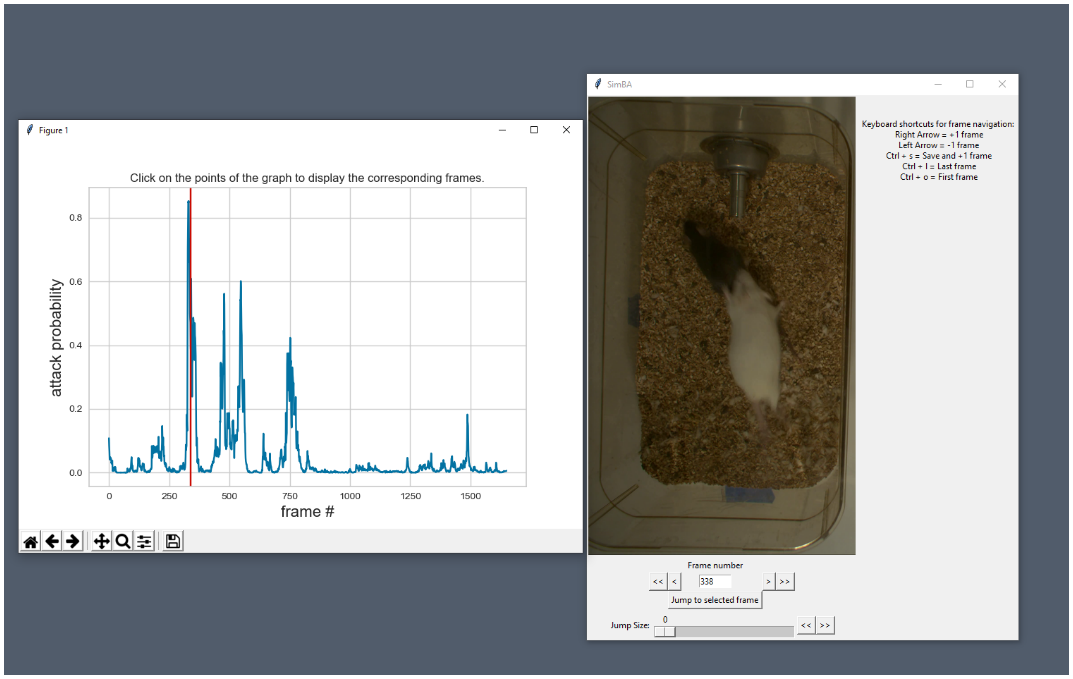
   - `Graph window`: model prediction probability versus frame numbers will be plotted. The graph is interactive - click on the graph and the frame window will display the selected frames.
   - `Frame window`: Frames of the chosen video with controls.



Click on the points of the graph to display the corresponding frames.

Keyboard shortcuts for frame navigation:
Right Arrow = +1 frame
Left Arrow = -1 frame
Ctrl + s = Save and +1 frame
Ctrl + l = Last frame
Ctrl + o = First frame

5. As you **double click** on points on the graph, the selected frame will pop up in the adjacent window. There will be a red line to show the points that you have clicked.



Click on the points of the graph to display the corresponding frames.

Keyboard shortcuts for frame navigation:
Right Arrow = +1 frame
Left Arrow = -1 frame
Ctrl + s = Save and +1 frame
Ctrl + l = Last frame
Ctrl + o = First frame

6. Once it jumps to the desired frame, you can navigate through the frames to determine if the behavior is present. This step is to find the optimal threshold to validate your model.

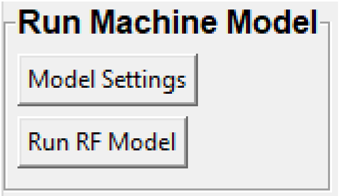
7. Once the threshold is determined, enter the threshold into the `Discrimination threshold` entry box and the desire minimum behavior bouth length into the `Minimum behavior bout lenght(ms)` entrybox. Note: we recommend using a minimum duration of 0ms and following this with Kleinberg Smoothing as described further down.

- `Discrimination threshold`: The level of probability required to define that the frame belongs to the target class. Accepts a float value between 0.0-1.0. For example, if set to 0.50, then all frames with a probability of containing the behavior of 0.5 or above will be classified as containing the behavior. For more information on classification threshold, click here
- `Minimum behavior bout length (ms)`: The minimum length of a classified behavioral bout. **Example**: The random forest makes the following attack predictions for 9 consecutive frames in a 50 fps video: 1,1,1,1,0,1,1,1,1. This would mean, if we don't have a minimum bout length, that the animals fought for 80ms (4 frames), took a break for 20ms (1 frame), then fought again for another 80ms (4 frames). You may want to classify this as a single 180ms attack bout rather than two separate 80ms attack bouts. With this setting you can do this. If the minimum behavior bout length is set to 20, any interruption in the behavior that is 20ms or shorter will be removed and the behavioral sequence above will be re-classified as: 1,1,1,1,1,1,1,1,1 - and instead classified as a single 180ms attack bout.

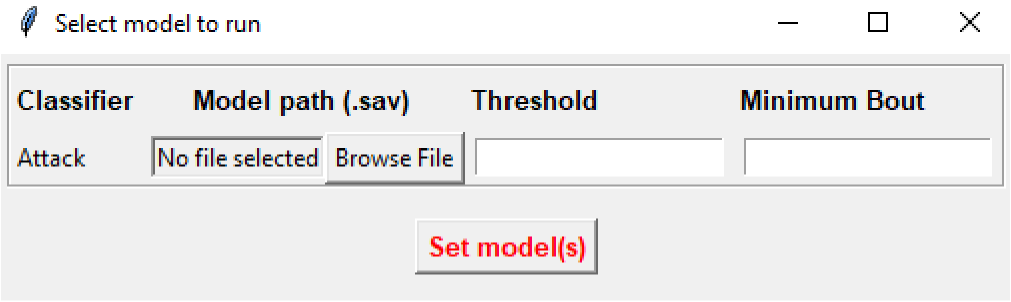
8. Click `Validate` to validate your model. **Note that this step will take a long time as it will generate a lot of frames.**

Step 8: Run Machine Model

This step runs behavioral classifiers on new data.



1. Under the **Run Machine Model** heading, click on `Model Settings`. The following window with the classifier names defined in the *project_config.ini* file will pop up.

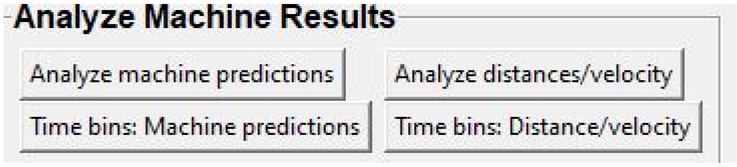


Classifier	Model path (.sav)		Threshold	Minimum Bout
Attack	No file selected	Browse File		

2. Click on `Browse File` and select the *Attack.sav* file from *C:/Golden/Classifier*.
3. Once all the models have been chosen, click on `Set Model` to save the paths.
4. Fill in the `Discrimination threshold`.
   - `Discrimination threshold`: The level of probability required to define that the frame belongs to the target class that you found in step 7.
5. Fill in the `Minimum behavior bout length`.
   - `Minimum behavior bout length (ms)`: Your desired minimum bout length.
6. Click on `Set model(s)`, and exit out of the current window, then click on `Run RF Model` to run the machine model on the new data.

Step 9: Analyze Machine Results

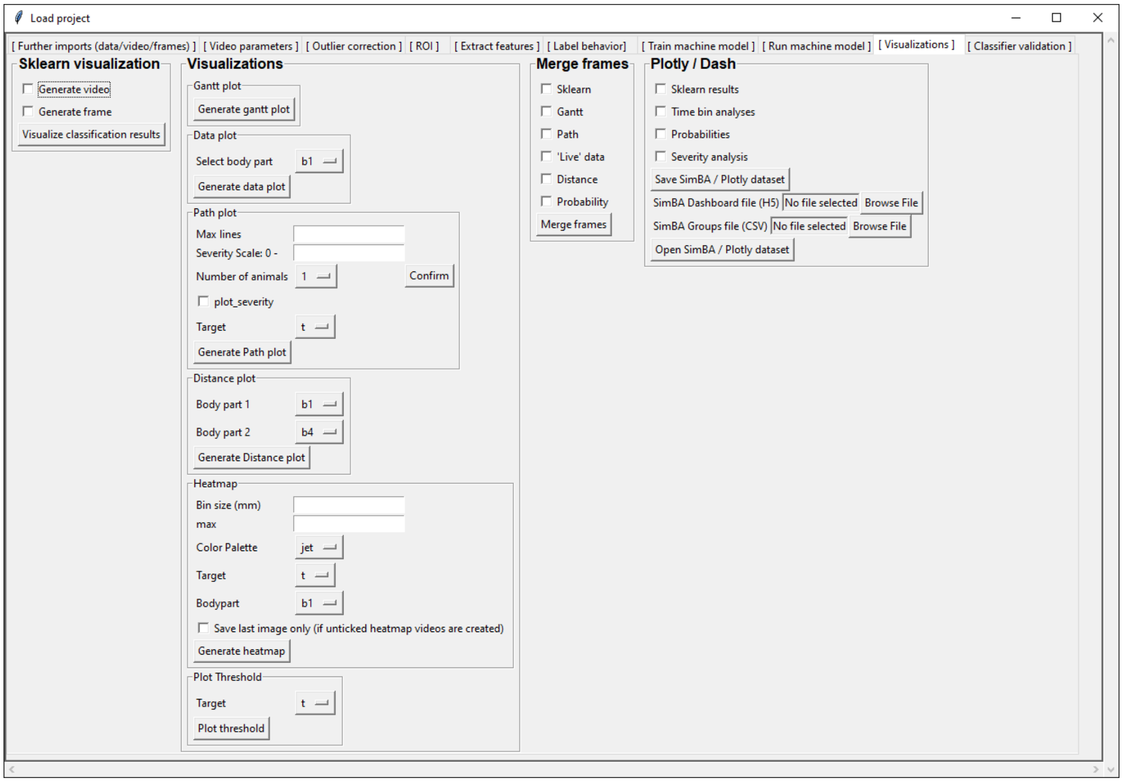
Access this menu through the `Load project` menu and the `Run machine model` tab. This step performs summary analyses and presents descriptive statistics in .csv file format. There are three forms of summary analyses: `Analyze`, `Analyze distance/velocity`, and `Analyze severity`.


Analyze Machine Results
Analyze machine predictions
Analyze distances/velocity
Time bins: Machine predictions
Time bins: Distance/velocity

- `Analyze machine prediction`: This button generates descriptive statistics for each predictive classifier in the project, including the total time, the number of frames, total number of 'bouts', mean and median bout interval, time to first occurrence, and mean and median interval between each bout. A date-time stamped output csv file with the data is saved in the `/project_folder/log` folder.
- `Analyze distance/velocity`: This button generates descriptive statistics for mean and median movements and distances between animals. The date-time stamped output csv file with the data is saved in the `/project_folder/log` folder.

Step 10: Sklearn Visualization

These steps generate visualizations of features and machine learning classification results. This includes images and videos of the animals with prediction overlays, gantt plots, line plots, paths plots and data plots. In this step the different frames can also be merged into video mp4 format.



1. Under the **Sklearn visualization** heading, check on the box and click on `Visualize classification results`.

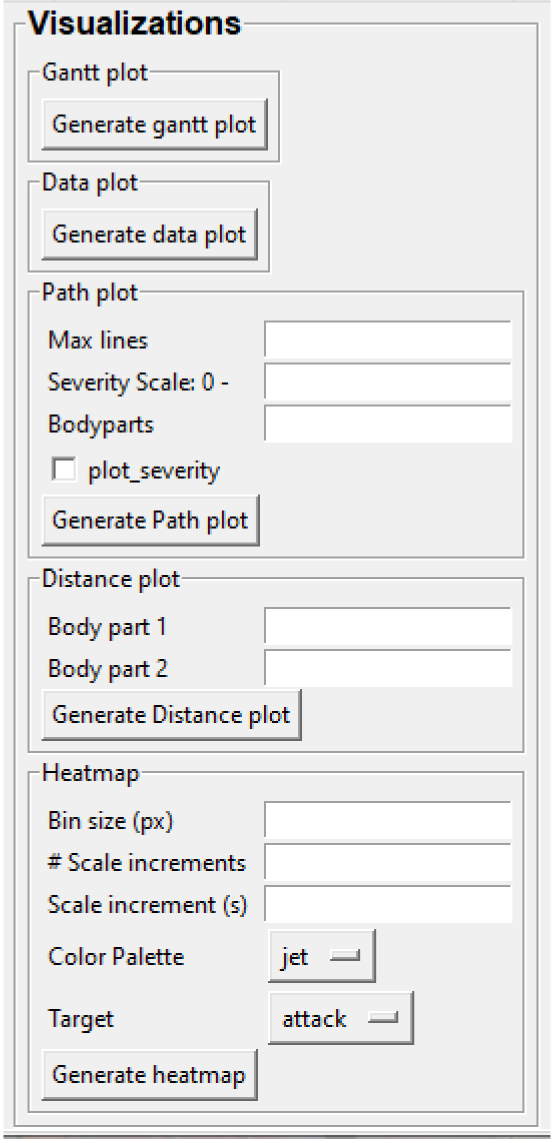
- `Generate video`: This generates a video of the classification result
- `Generate frame`: This generates frames(images) of the classification result

> **Note:** Generating frames is required if you want to merge frames into videos in the future.

This step grabs the frames of the videos in the project, and draws circles at the location of the tracked body parts, the convex hull of the animal, and prints the behavioral predictions on top of the frame. For an example, click here

Step 11: Visualizations

The user can also create a range of plots: **gantt plot**, **Data plot**, **Path plot**, **Distance plot**, and **Heatmap**.



Color Palette: jet

Target: attack

Gantt plot

Gantt plot generates gantt plots that display the length and frequencies of behavioral bouts for all the videos in the project.

1. Under the **Gantt plot** heading, click on `Generate Gantt plot` and gantt plot frames will be generated in the `project_folder/frames/output/gantt_plots` folder.

Data plot

Generates 'live' data plot frames for all of the videos in the project that display current distances and velocities.

1. Under the **Data plot** heading, click on `Generate Data plot` and data plot frames will be generated in the `project_folder/frames/output/live_data_table` folder.

Path plot

Generates path plots displaying the current location of the animal trajectories, and location and severity of attack behavior, for all of the videos in the project.

1. Under the **Path plot** heading, fill in the following user defined values.
   - `Max Lines`: Integer specifying the max number of lines depicting the path of the animals. For example, if 100, the most recent 100 movements of animal 1 and animal 2 will be plotted as lines.
   - `Severity Scale`: Integer specifying the scale on which to classify 'severity'. For example, if set to 10, all frames containing attack behavior will be classified from 1 to 10 (see above).
   - `Bodyparts`: String to specify the bodyparts tracked in the path plot. For example, if Nose_1 and Centroid_2, the nose of animal 1 and the centroid of animal 2 will be represented in the path plot.
   - `plot_severity`: Tick this box to include color-coded circles on the path plot that signify the location and severity of attack interactions.
2. Click on `Generate Path plot`, and path plot frames will be generated in the `project_folder/frames/output/path_plots` folder.

Distance plot

Generates distance line plots between two body parts for all of the videos in the project.

1. Fill in the `Body part 1` and `Body part 2`
   - `Body part 1`: String that specifies the the bodypart of animal 1. Eg., Nose_1
   - `Body part 2`: String that specifies the the bodypart of animal 1. Eg., Nose_2
2. Click on `Generate Distance plot`, and the distance plot frames will be generated in the `project_folder/frames/output/line_plot` folder.

Heatmap

Generates heatmap of behavior that happened in the video.

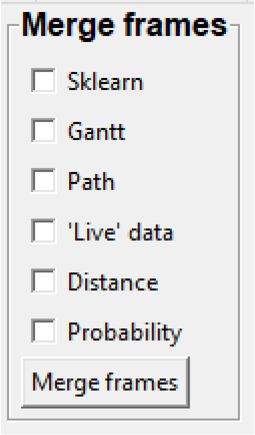
To generate heatmaps, SimBA needs several user-defined variables:

- `Bin size(mm)` : Pose-estimation coupled with supervised machine learning in SimBA gives information on the location of an event at the single pixel resolution, which is too-high of a resolution to be useful in heatmap generation. In this entry box, insert an integer value (e.g., 100) that dictates, in pixels, how big a location is. For example, if the user inserts *100*, and the video is filmed using 1000x1000 pixels, then SimBA will generate a heatmap based on 10x10 locations (each being 100x100 pixels large).
- `max` (integer, or auto): How many color increments on the heatmap that should be generated. For example, if the user inputs *11*, then a 11-point scale will be created (as in the gifs above). If the user inserts auto in this entry box, then SimBA will calculate the ideal number of increments automatically for each video.
- `Color Palette` : Which color pallette to use to plot the heatmap. See the gifs above for different output examples.
- `Target`: Which target behavior to plot in the heatmap. As the number of behavioral target events increment in a specific location, the color representing that region changes.
- `Bodypart`: To determine the location of the event in the video, SimBA uses a single body-part coordinate. Specify which body-part to use here.
- `Save last image only`: Users can either choose to generate a "heatmap video" for every video in your project. These videos contain one frame for every frame in your video. Alternative, users may want to generate a **single image** representing the final heatmap and all of the events in each video - with one png for every video in your project. If you'd like to generate single images, tick this box. If you do not tick this box, then videos will be generated (which is significantly more time-consuming).

2. Click `Generate heatmap` to generate heatmap of the target behavior. For more information on heatmaps based on behavioral events in SimBA - check the tutorial for scenario 2 - visualizing machine predictions

Step 12: Merge Frames

Merge all the generated plots from the previous step into a single frame and generate a **video** as an **output**.



Note: All the frames must be generated in the previous step for this to work. This step combines all the frames(images) that are generated and merge them together and make a video.**

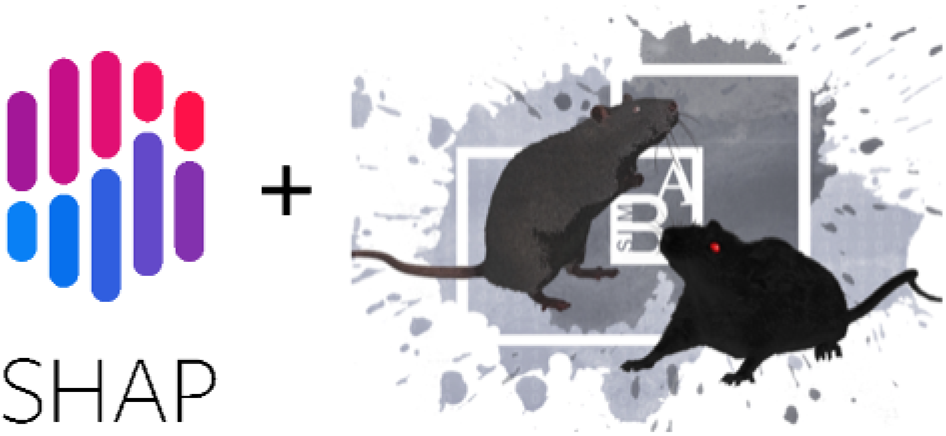
1. Check on the plot that you wish to merge together and output as a single video.
2. Under **Merge Frames**, click `Merge Frames` and frames with all the generated plots will be combined and saved in the `project_folder/frames/output/merged` folder in a video format.

3.1.7 Other cool stuff

- SHAP
- Kleinberg smoothing
- FSTTC
- ROI

3.2 SHAP

Model explainability and calculating feature SHAP values in SimBA



An understanding of how machine learning models reach their decisions is important not only for the scientific method but can also help you gain insight into how your classifier works and what makes it different or similar to other classifiers. Machine learning explainability metrics, such as SHAP, can help you answer questions like:

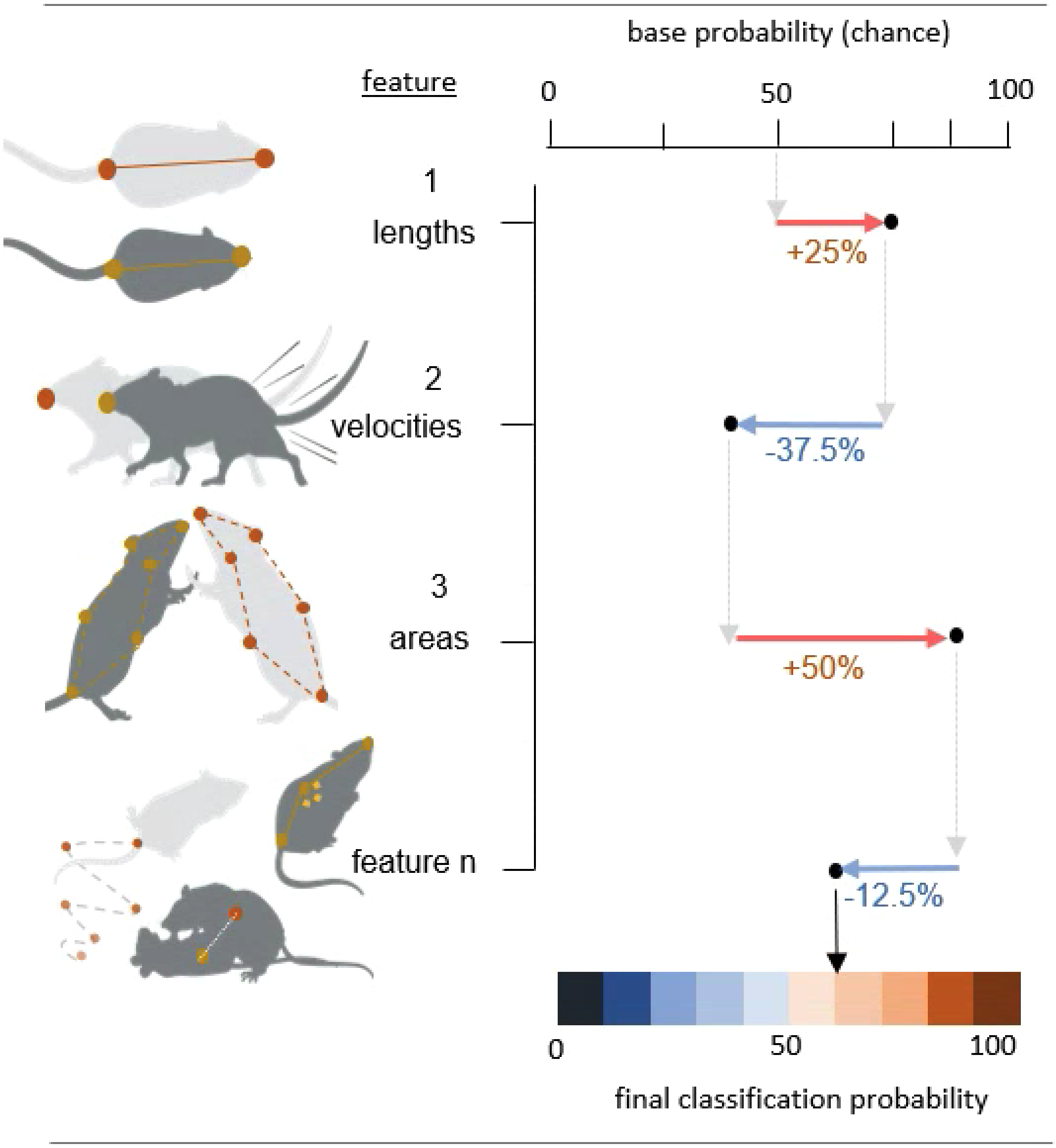
- Why does my classifier think some specific frames contain my behavior of interest, while other frames do not?
- Why does the classifier generated by annotator X classify events so differently (or similarly) to the classifier generated by annotator Y?
- Why does my tracking model think the location of the nose of the animal is located in this particlar part of the image?
- Which, of several classifiers, classify the target events by using the same behavioural features that a human observers would use to classify the same events?
- Are there any potential differences in the features that annotator X and annotator Y look at when annotating videos for the presence or absence of the same target behavior?

Explainability metrics can be very important, as it is possible that the classifiers you are using *appear* to look for the same behavioral target behaviours and features as a human observer would, while the classifier in fact looks as something very different from the human observer. Such weaknesses are typically revealed when analyzing new videos in new recording environments, that were not included in the data used to train the classifier, and explainability metrics can help you avoid those pitfalls.

Here we will look at how we can use SHAP (SHapley Additive exPlanations) within SimBA to calculate how much each feature contributes to the final behavioral classification score for each annotated video frame. Through this method we will get an verbalizable explanation for the classification probability score for each frame, such as:

Frame N in Video X was classified as containing my behavior of interest, mainly because of the distance between animal A and B, but also because the movements of animal A. In frame N, the larger movements of the animals increased the behavior classification probability with 20%, and the distance between the animals increased the classification probability with a further 70%.

In brief, when using SHAP, each feature is evaluated independently, and the final classification probability is distributed among the individual features according to their contribution to it. This value is calculated after exhaustive permutations within the order of feature-introductions into the classification scenario:



The *base probability* in the figure above is the probability of picking a frame that contains your behavior by pure chance (e.g., if half of your video frames contain you behavior of interest , then the base probability will be 50%; more info below!). The values associated with each feature describe the features effect on the classification probability. To read more about SHAP values, also see the SHAP GitHub repository which SimBA wraps, or read the excellent SHAP paper in Nature Machine Learning Intelligence.

3.2.1 Part 1: Generate a dataset

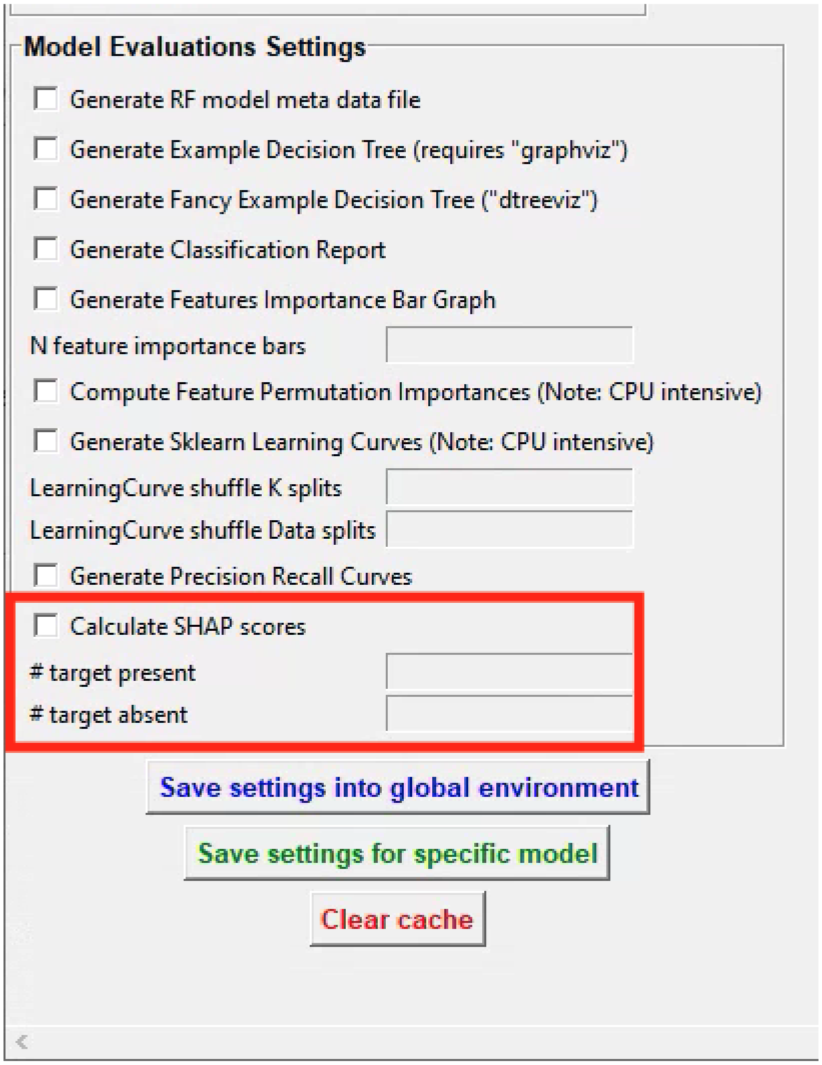
SimBA calculates SHAP values for the classifier at the same time as the model is being trained. Thus, before analysing SHAP scores, we need a dataset that contains behavioral annotations. You will need complete the steps detailed in the Scenario 1 tutorial **Part 1 Step 1** up to **Part 2 Step 6**. That is, you will need to complete everything from Creating a project up to, and including labelling behavioral events.

Note: If you already have annotations generated elsewhere (e.g., downloaded from the SimBA OSF repository, you may not have to go through **Part 1 Step 1** to **Part 2 Step 6** as detailed above. When calculating the SHAP values, SimBA will loook inside your `project_folder/csv/targets_inserted` subdirectory for files containing the annotations (just as SimBA does when generating the classifier). So to calculate SHAP values, SimBA needs this folder to be populated with files containing behavioral annotations.

3.2.2 Part 2: Compute SHAP scores

Step 1: Define SHAP settings

Navigate to the `Train machine model` tab and click on `Settings`. In the pop-up window, fill out your model `hyperparameter` settings as described HERE. At the bottom of the `Settings` pop-up window, you will see these entry boxes. Begin by ticking the `Calculate SHAP values` entry box.



Model Evaluations Settings

□ Generate RF model meta data file

□ Generate Example Decision Tree (requires "graphviz")

□ Generate Fancy Example Decision Tree ("dtreeviz")

□ Generate Classification Report

□ Generate Features Importance Bar Graph

N feature importance bars

□ Compute Feature Permutation Importances (Note: CPU intensive)

□ Generate Sklearn Learning Curves (Note: CPU intensive)

LearningCurve shuffle K splits

LearningCurve shuffle Data splits

□ Generate Precision Recall Curves

□ Calculate SHAP scores

target present

target absent

Save settings into global environment
Save settings for specific model
Clear cache

When this box is ticked, and the entry boxes are filled in, SiMBA will also calculate SHAP values while generating your behavioral classifier. SimBA will use your annotations in the `project_folder/csv/targets_inserted` folder when doing so. SHAP calculations are an computationally expensive process, so you most likely can't use **all** of your annotations to calculate them. The time it takes to calculate SHAP scores for a single frame will depend on how many features you have the the specs of your computer, but in all likelihood it will take **several seconds, and possibly tens of seconds**, for a single frame. We therefore have to select a random sub-set of frames to calculate SHAP scores for.

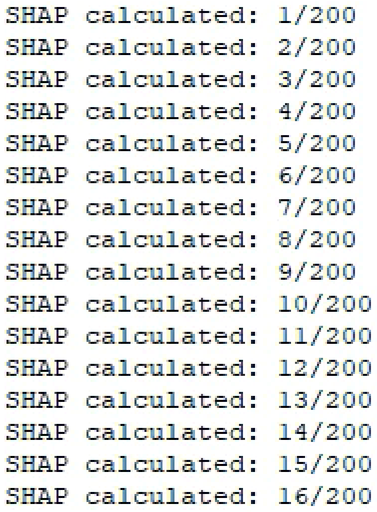
- In the `# target present` entry box, enter the number of frames (integer - e.g., `200`) with the behavioral target **present** to calculate SHAP values for.
- In the `# target absent` entry box, enter the number of frames (integer - e.g., `200`) with the behavioral target **absent** to calculate SHAP values for.

Once you have filled in the SHAP entry boxes, click on either *save settings into global environment* or *save settings for specific model*, depening on wether you are generating one model, or several models at once. For more information on generating one vs several models, click HERE.

Click to close the `Settings` pop-up window.

Step 2: Train the classifier and generate SHAP values.

Start the classifier training by clicking on `Train single model from global environment` or `Train multiple models, one for each saved settings`. You will be able to follow the progress in the Terminal window. A new message will be printed in the main SimBA terminal for every SHAP score computed. If you are calculating the shap scores for 200 frames, you can expect the beginning of the calculations to look something like this in the main SimBA terminal:



```
SHAP calculated: 1/200
SHAP calculated: 2/200
SHAP calculated: 3/200
SHAP calculated: 4/200
SHAP calculated: 5/200
SHAP calculated: 6/200
SHAP calculated: 7/200
SHAP calculated: 8/200
SHAP calculated: 9/200
SHAP calculated: 10/200
SHAP calculated: 11/200
SHAP calculated: 12/200
SHAP calculated: 13/200
SHAP calculated: 14/200
SHAP calculated: 15/200
SHAP calculated: 16/200
```

Note: As noted above, calculating SHAP scores is computationally expensive and depending on the number of frames you entered in the `# target present` and `# target absent`, this could take a while. If you are calculating SHAP scores for a lot of frames, it's best to make it an overnighter.

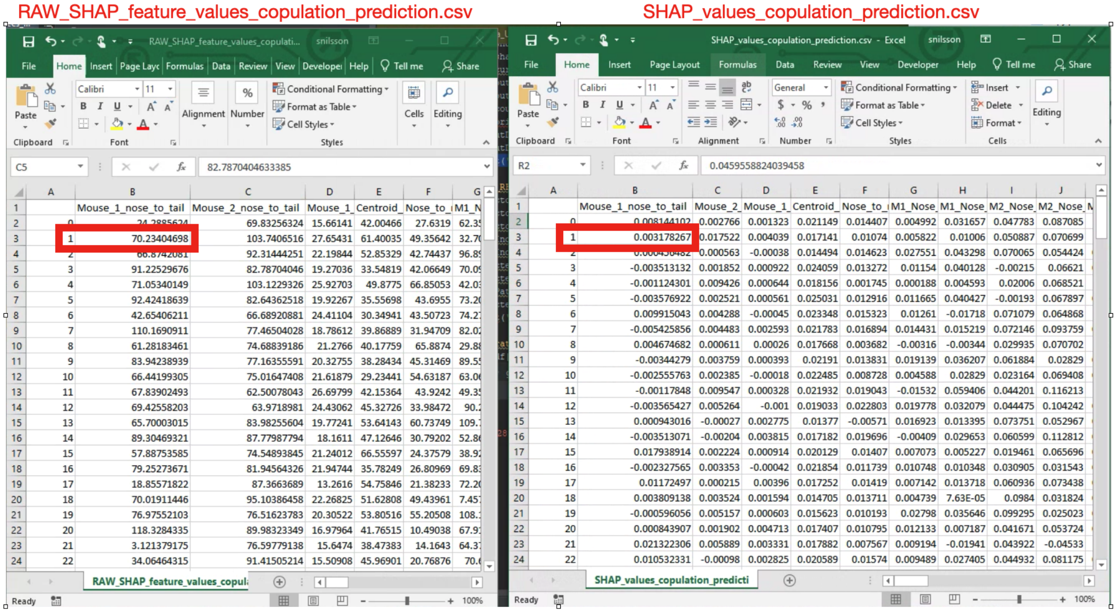
Once complete, you will see the following message: `All SHAP data saved in project_folder/models/evaluations directory`. Navigate to the directory to access your SHAP values. In this folder you will see two seperate CSV files. For example, if you performed SHAP calculations for a classifier called `copulation`, you will find the files `RAW_SHAP_feature_values_copulation_prediction.csv` and `SHAP_values_copulation_prediction.csv`. Below we will go through how the data in these two files can be interpreted.

Step 3: Interpreting the SHAP value ouput generated by SimBA.

The two SHAP value output files have an equal number of rows, where every row represent one of the frames that we calculated SHAp scores for. If you chose to generate SHAP values for 200 frames, each of the two files will contain 200 rows, where row *N* within both files represent the data for the **same frame**. The first file (`SHAP_values_copulation_prediction.csv`) contains the SHAP probability values. The second file (`RAW_SHAP_feature_values_copulation_prediction.csv`) contain the **raw** feature values for the same frames. The reason for generating two files is that it is sometimes necessary to match the SHAP values (represented in the `SHAP_values_copulation_prediction.csv`) with an actual feature values (represented in the `RAW_SHAP_feature_values_copulation_prediction.csv`).

RAW_SHAP_feature_values_copulation_prediction.csv

SHAP_values_copulation_prediction.csv



C5 | 82.7870404633385

	A	B	C	D	E	F	G
1		Mouse_1_nose_to_tail	Mouse_2_nose_to_tail	Mouse_1_	Centroid_	Nose_to_	M1_N
2	0	24.2885624	69.83256324	15.66141	42.00466	27.6319	62.3
3	1	70.23404698	103.7406516	27.65431	61.40035	49.35642	32.7
4	2	66.8742081	92.31444251	22.19844	52.85329	42.74437	96.8
5	3	91.22529676	82.78704046	19.27036	33.54819	42.06649	70.0
6	4	71.05340149	103.1229326	25.92703	49.8775	66.85053	42.0
7	5	92.42418639	82.64362518	19.92267	35.55698	43.6955	73.2
8	6	42.65406211	66.68920881	24.41104	30.34941	43.50723	74.2
9	7	110.1690911	77.46504028	18.78612	39.86889	31.94709	82.0
10	8	61.28183461	74.68839186	21.2766	40.17759	65.8874	29.8
11	9	83.94238939	77.16355591	20.32755	38.28434	45.31469	89.5
12	10	66.44199305	75.01647408	21.61879	29.23441	54.63187	63.0
13	11	67.83902493	62.50078043	26.69799	42.15364	43.9242	49.3
14	12	69.42558203	63.9718981	24.43062	45.32726	33.98472	90.
15	13	65.70003015	83.98255604	19.77241	53.64143	60.73749	109.
16	14	89.30469321	87.77987794	18.1611	47.12646	30.79202	52.8
17	15	57.88753585	74.54893845	21.24012	66.55597	24.37579	38.9
18	16	79.25273671	81.94564326	21.94744	35.78249	26.80969	69.8
19	17	18.85571822	87.3663689	13.2616	54.75846	21.38233	72.2
20	18	70.01911446	95.10386458	22.26825	51.62808	49.43961	7.45
21	19	76.97552103	76.51623783	20.30522	53.80516	55.20508	108.
22	20	118.3284335	89.98323349	16.97964	41.76515	10.49038	67.9
23	21	3.121379175	76.59779138	15.6474	38.47383	14.1643	64.3
24	22	34.06464315	91.41505214	15.50908	45.96901	20.76876	70.

RAW_SHAP_feature_values_copul

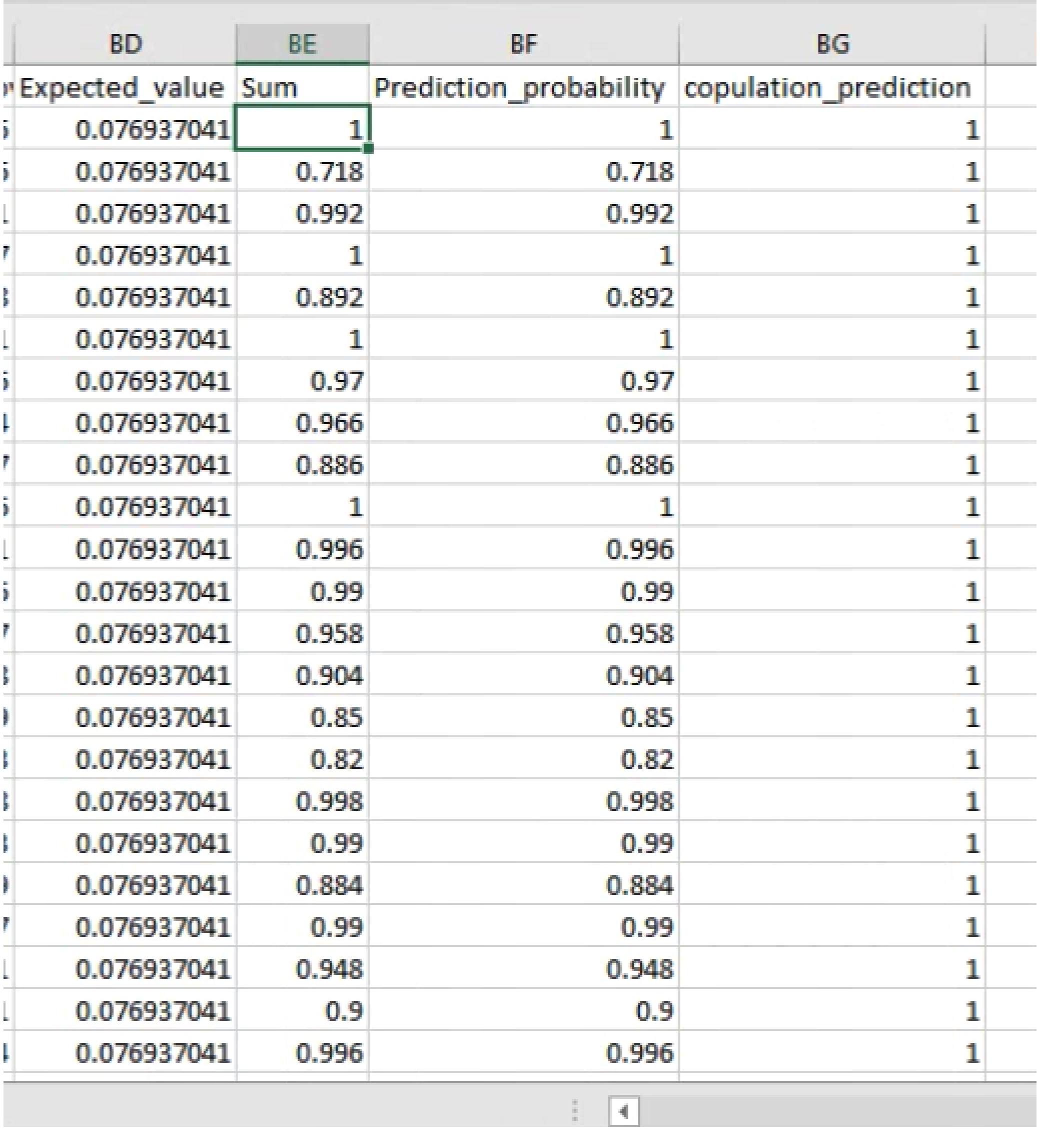
R2 | 0.0459558824039458

	A	B	C	D	E	F	G	H	I	J
1		Mouse_1_nose_to_tail	Mouse_2_	Mouse_1_	Centroid_	Nose_to_	M1_Nose_	M1_Nose_	M2_Nose_	M2_Nose_
2	0	0.008144102	0.002766	0.001323	0.021149	0.014407	0.004992	0.031657	0.047783	0.087085
3	1	0.003178267	0.017522	0.004039	0.017141	0.01074	0.005822	0.01006	0.050887	0.070699
4	2	0.000456482	0.000563	-0.00038	0.014494	0.014623	0.027551	0.043298	0.070065	0.054424
5	3	-0.003513132	0.001852	0.000922	0.024059	0.013272	0.01154	0.040128	-0.00215	0.06621
6	4	-0.001124301	0.009426	0.000644	0.018156	0.001745	0.000188	0.004593	0.02006	0.068521
7	5	-0.003576922	0.002521	0.000561	0.025031	0.012916	0.011665	0.040427	-0.00193	0.067897
8	6	0.009915043	0.004288	-0.00045	0.023348	0.015323	0.01261	-0.01718	0.071079	0.064868
9	7	-0.005425856	0.004483	0.002593	0.021783	0.016894	0.014431	0.015219	0.072146	0.093759
10	8	0.004674682	0.000611	0.00026	0.017668	0.003682	-0.00316	-0.00344	0.029935	0.070702
11	9	-0.00344279	0.003759	0.000393	0.02191	0.013831	0.019139	0.036207	0.061884	0.02829
12	10	-0.002555763	0.002385	-0.00018	0.022485	0.008728	0.004588	0.02829	0.023164	0.069408
13	11	-0.00117848	0.009547	0.000328	0.021932	0.019043	-0.01532	0.059406	0.044201	0.116213
14	12	-0.003565427	0.005264	-0.001	0.019033	0.022803	0.019778	0.032079	0.044475	0.104242
15	13	0.000943016	-0.00027	0.002775	0.01377	-0.00571	0.016923	0.013395	0.073751	0.052967
16	14	-0.003513071	-0.00204	0.003815	0.017182	0.019696	-0.00409	0.029653	0.060599	0.112812
17	15	0.017938914	0.002224	0.000914	0.020129	0.01407	0.007073	0.005227	0.019461	0.065696
18	16	-0.002327565	0.003353	-0.00042	0.021854	0.011739	0.010748	0.010348	0.030905	0.031543
19	17	0.01172497	0.000215	0.00396	0.017252	0.01419	0.007142	0.013718	0.060936	0.073438
20	18	0.003809138	0.003524	0.001594	0.014705	0.013711	0.004739	7.63E-05	0.0984	0.031824
21	19	-0.000596056	0.005157	0.000603	0.015623	0.010193	0.02798	0.035646	0.099295	0.025023
22	20	0.000843907	0.001902	0.004713	0.017407	0.010795	0.012133	0.007187	0.041671	0.053724
23	21	0.021322306	0.005889	0.003331	0.017882	0.007567	0.009194	-0.01941	0.043922	-0.04533
24	22	0.010532331	-0.00098	0.002825	0.020589	0.01574	0.009489	0.027405	0.044932	0.081175

SHAP_values_copulation_predicti

To help understand this, I've placed the two CSV files next to each other in the image above, with the `RAW feature values` file shown on the left, and the `SHAP values` file on the right. The red rectangle in the RAW values, on the left, shows that raw feature distance between the nose and the tail of animal number 1 (the feature name is in the header) was 70.23404 millimeters in frame number 1. Conversely, the SHAP values, shown on the right, shows that the distance between the nose and the tail of animal number 1 **increased** the copulation probability in frame number 1 with 0.317%.

The last four columns of the `SHAP_values_copulation_prediction.csv` file contain some information that might be helpful for interpretating the data, and give a sanity check that the calculations were done as expected:

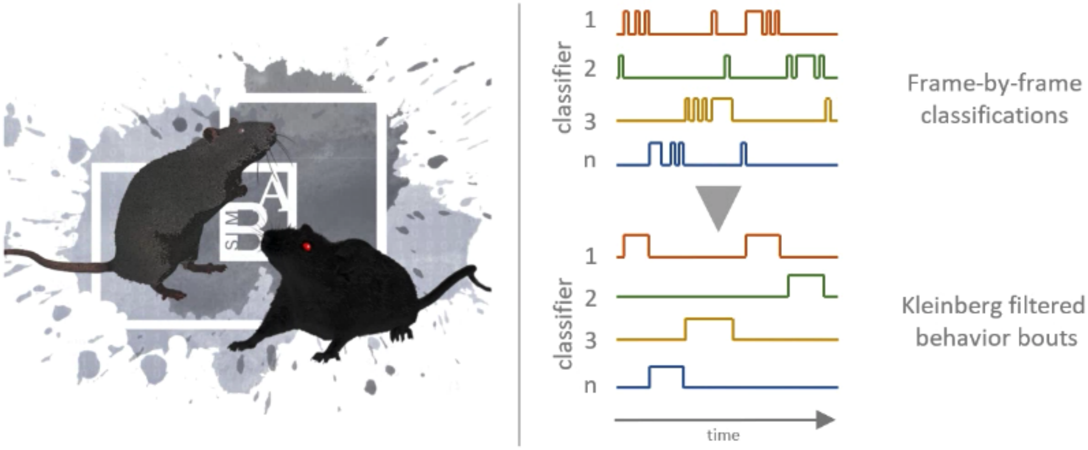


BD	BE	BF	BG
Expected_value	Sum	Prediction_probability	copulation_prediction
0.076937041	1	1	1
0.076937041	0.718	0.718	1
0.076937041	0.992	0.992	1
0.076937041	1	1	1
0.076937041	0.892	0.892	1
0.076937041	1	1	1
0.076937041	0.97	0.97	1
0.076937041	0.966	0.966	1
0.076937041	0.886	0.886	1
0.076937041	1	1	1
0.076937041	0.996	0.996	1
0.076937041	0.99	0.99	1
0.076937041	0.958	0.958	1
0.076937041	0.904	0.904	1
0.076937041	0.85	0.85	1
0.076937041	0.82	0.82	1
0.076937041	0.998	0.998	1
0.076937041	0.99	0.99	1
0.076937041	0.884	0.884	1
0.076937041	0.99	0.99	1
0.076937041	0.948	0.948	1
0.076937041	0.9	0.9	1
0.076937041	0.996	0.996	1

The first of these 4 columns (`Expected_value`), contains the baseline probability value. That is - in this toy example - if you picked a frame at random, there is a 7. 693% chance that the frame contains the target behavior `copulation`.

The second column (`Sum`) contains the sum of all of the SHAP values for each individual frame. The third column (`Prediction_probability`) is the classifiers probability for the presence of the behavior in each individual frame. **These two columns are generated as a sanity check, because the final prediction probability seen in the `Prediction_probability` column should equal the sum of all the SHAP values seen in the `Sum` column**. If the values in these two columns are **not** identical, then something has gone astray. Check in with us on the Gitter chat channel or raise a an issue and we may be able to help. The fourth column (`copulation_prediction`) will read either 0 or 1, and tell you if this particular frame was annotated as containing the behavior of interest (`1`), or **not** containing the behavior of interest (`0`).

3.3 Kleinberg behavior classification smoothing in SimBA



Classifiers generated in SimBA provide a classification for the presence vs. absence of a behavior of interest in every frame of the analysed videos. This mean that, according to the classifier, a behavioural bout can be a short as one frame of the video (e.g., 20ms in a 50 fps video). Similarly, according to the classifier, two behavioral bouts can also be separated by a single frame where the classifier judges the behavioral event to be absent. Often we want to smooth these classified events, so that the (i) behaviours expressed for very short periods are removed, and (ii) behavioral bouts seperated by very short time are concatenated. We can do this by setting heuristic rules in SimBA when we run the classifiers (i.e., a behavioural bout cannot be shorter than N milliseconds).

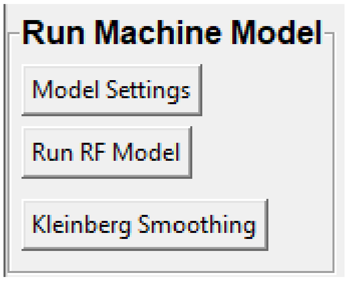
However, a more sophisticated (albeit less interpretable) approach is to use infinite hidden Markov models through the Kleinberg burst detection method to unify perdiods when the behavior is more and less likely. The Kleinberg burst detection method for smoothing behavioral data was championed by the Curley lab in the excellent Lee et al. 2019 paper. SimBA performs Kleinberg burst smoothing by using a modified version of the no-longer maintained pyburst package.

After applying the Kleinberg smoother, you can expect the output to look like below - with the original video on the left and the kleinberg smoothened video on the right. Click HERE to see a longer higher resolution video of expected results.

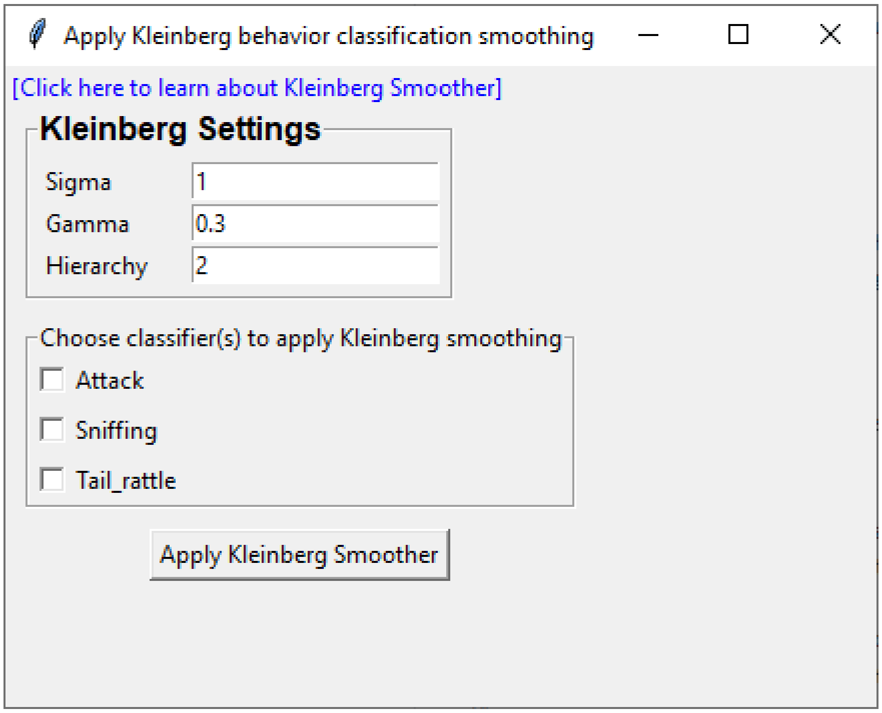
> **Note:** If you are smoothing your data using the Kleinberg smoother, I recommend refraining from setting heuristic rules in SimBA when we run the classifiers (i.e., a behavioural bout cannot be shorter than N milliseconds) and instead let the Kleinberg algorithm take care of **all** of the smoothing. Thus, when you first run the classifier, set the **min bout duration** for the classifiers you want to smooth (as described HERE to `0`.)

3.3.1 How to use Kleinberg Filter in SimBA

1. After running the machine model (i.e., after clicking on `Run RF Model` as documented HERE, go ahead and click on the `Kleinberg Smoothing` button.



2. You should see the below `settings menu` pop up. Check the classifier(s) that you wished to apply the Kleinberg smoothing for and set the "hyperparameters". See the section below for more information about the hyperparameters and the default settings. Click on the `Apply Kleinberg Smoother` button to apply the filter.


Apply Kleinberg behavior classification smoothing
[Click here to learn about Kleinberg Smoother]
Kleinberg Settings

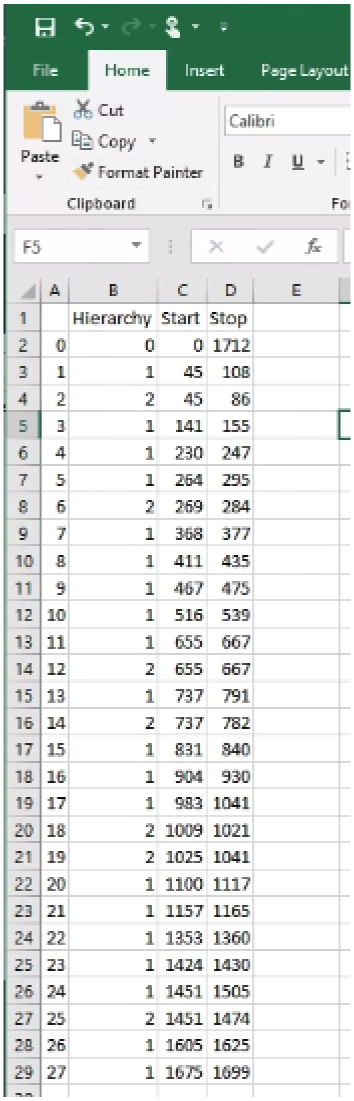
Sigma 1
Gamma 0.3
Hierarchy 2

Choose classifier(s) to apply Kleinberg smoothing

□ Attack
□ Sniffing
□ Tail_rattle

3. The files in the `project_folder/csv/machine_results` will be overwritten with new files, with the same file names as the original files, but now containing the kleinberg smoothened classifications for the selected classifiers. To compare the results, either (i) back-up the original files and open it up alongside the newly generated file and compare the *0* and *1* in the relevant classification columns in the right-most part of the file. Or, alternative, generate new classifier visualizations as documented HERE and compare the classifications pre- and post smoothing.

4. For troubleshooting purposes - for each classifier you smooth - SimBA will save a log file in the `project_folder/csv/logs` directory that is named after the classifier name and the current date and time, for example *Kleinberg_log_Attack_20210204202223.csv*. This file contains information on **all** the classified bouts found by the Kleinberg smoother, their hierarchy, and start and end frame. It may look like this:



	A	B	C	D	E
1		Hierarchy	Start	Stop	
2	0	0	0	1712	
3	1	1	45	108	
4	2	2	45	86	
5	3	1	141	155	
6	4	1	230	247	
7	5	1	264	295	
8	6	2	269	284	
9	7	1	368	377	
10	8	1	411	435	
11	9	1	467	475	
12	10	1	516	539	
13	11	1	655	667	
14	12	2	655	667	
15	13	1	737	791	
16	14	2	737	782	
17	15	1	831	840	
18	16	1	904	930	
19	17	1	983	1041	
20	18	2	1009	1021	
21	19	2	1025	1041	
22	20	1	1100	1117	
23	21	1	1157	1165	
24	22	1	1353	1360	
25	23	1	1424	1430	
26	24	1	1451	1505	
27	25	2	1451	1474	
28	26	1	1605	1625	
29	27	1	1675	1699	

Here, every row is a classified behavioral bout. The *Hierarchy* columns denotes which level in the hierarchy the bout belongs to. The *Start* columns denotes the frame when the behavioral bout started within this part of the hierarchy. The *End* columns denotes the frame when the behavioral bout ended within this part of the hierarchy. If you look carefully you see that the lower hierarchies encompass the same boural events further up the hierarchy.

If you are unhappy with the results, I suggest you tinker with the hierarchy value based on the result in this logfile - have a look with the majority of your classifies bouts are located and set your hierarchy hyperparameter accordingly.

If satisfied with the results, go ahead and compute new descriptive stastistics as documented HERE.

Hyperparameters

Through the SimBA menues the user can access three hyperparameters. For more information of what the hyperparameters mean, check out the R 'bursts' package documentation the Lee et al. 2019 paper, or the original paper.

In short,

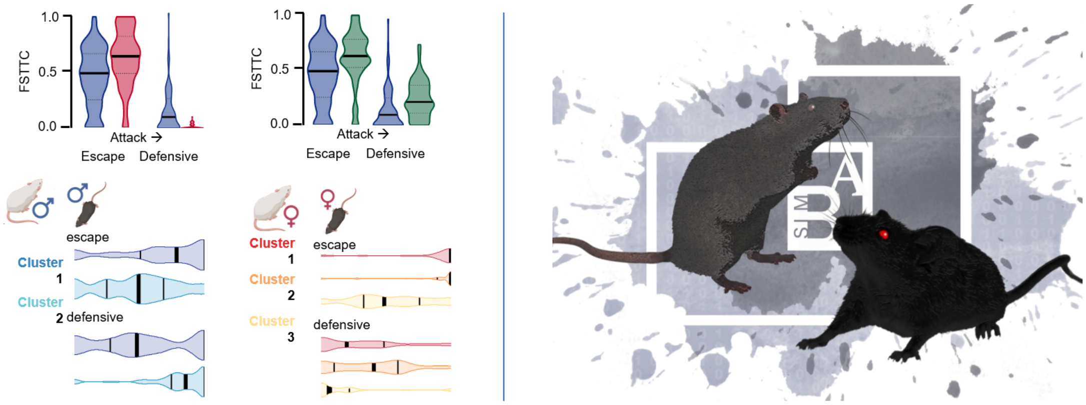
(i) Sigma - Higher values and the discrepancy between no-behavior events and behavior events has to be larger for the event to be recognised. **Higher sigma values and fewer behavioural bursts will be recognised (Default: 2)**

(ii) Gamma - Higher gamma values and behaviors needs to be sustained for a longer time for them to be recognised as burts. **Higher gamma values and fewer behavioural bursts will be recognised (Default: 0.3)**

(ii) Hierarchy - Which order or depth or the markov chain should be considered when evaluating the bursts. **Higher hierarchy values values and fewer behavioural bursts are likely to be recognised (Default: 1)**

3.4 FSTTC

Calculating forward-spike time tiling coefficents in SimBA



When classifying mutiple behaviors in SimBA, we may be interested in causal sequences. For example, does behavior A cause the expression of behavior B, or does behavior B cause the expression of behavior A (… and so one for all the different pairwise behavior combinations of interest).

Several statistical solutions to get to such answers have been presented (e.g., Haccou et al. 1988), and Lee et al. (2019) proposed the Forward Spike Time Tiling Coefficient (FSTTC), an adaptation of the Spike Time Tiling Coefficient (Cutts and Eglen, 2014), to detect how behaviors initiated by one animal can trigger behavioral responses of a second animal during dyadic encounters. We urge users interested in these metrics to check out the excellent Lee et al. (2019).

In short, FSTTC may be helpful to answer questions such as:

- Does resident attack behavior predominantly cause intruder defensive behavior or intruder escape behavior?
- Does attack behavior cause defensive behavior or does defensive behavior cause attack behavior?
- Does attack behavior typically follow defensive behavior or escape behavior?
- Does rearing events typically follow sniffing events or freezing events?

Note: Note that SimBA will calculate the FSTTC for all the behaviors selected by the user, and SimBA does **not** require there to be two, or more, tracked animals.

The FSTCC is calculated as:

$$\text{FSTTC} = \frac{1}{2}\left(\frac{P - T_B}{1 - PT_B} + \frac{P - T_A}{1 - PT_A}\right)$$

Where the t hyper-parameter is a time-window (e.g 2000ms - Lee et al. 2019). P is the proportion of behavior B onsets that fall inside t following the onset of behavior A. T_A is the proportion of the total session time that falls inside t following behavior A onset. T_B is the proportion of the total session time that falls inside t following behavior B onset.

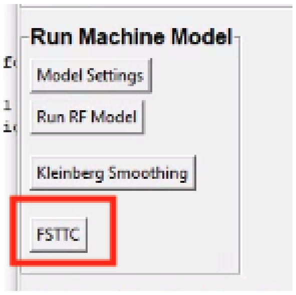
Note: Although transitional relationships of behavioral events often are evaluated and visualized using Markov decision processes, such techniques may require mutually exclusive states and this introduces statistical challenges for multi-individual environments. We recognise that the more sophisticated approach to explore casuse and effect in multi-individual environments would be multi-agent reinforcement learning techniques but.. yeah.. this works and RL have to wait.. :)

3.4.1 Step 1: Generate a dataset

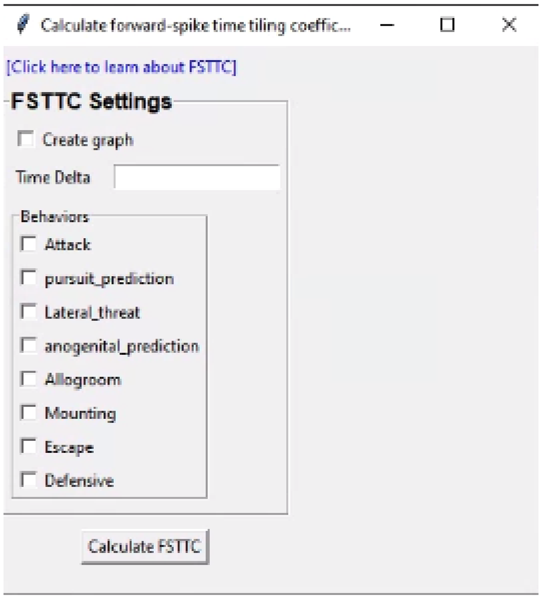
First, SimBA needs classified data to calculate the FSTTC. SimBA will look in the `project_folder/csv/machine_results` directory for files, and calculate the FSTTC scores for all the files in this folder. Thus, before calculating the FSTTC, make sure that you have run your classifiers on your data as documented HERE. In other words, make sure you have processed your data as documented in the Scenario 1 Tutorial, up to and including Step 8.

3.4.2 Step 2: Perform FSTTC analysis.

Load you SimBA project, and click on the `[Run machine model]` tab. In the *Run Machine Model* submenu, you should see a button named `FSTTC`.



Go ahead and click on it, and you should see the following menu pop up:



Calculate forward-spike time tiling coeffic...

[Click here to learn about FSTTC]

FSTTC Settings

□ Create graph

Time Delta

Behaviors

□ Attack

□ pursuit_prediction

□ Lateral_threat

□ anogenital_prediction

□ Allogroom

□ Mounting

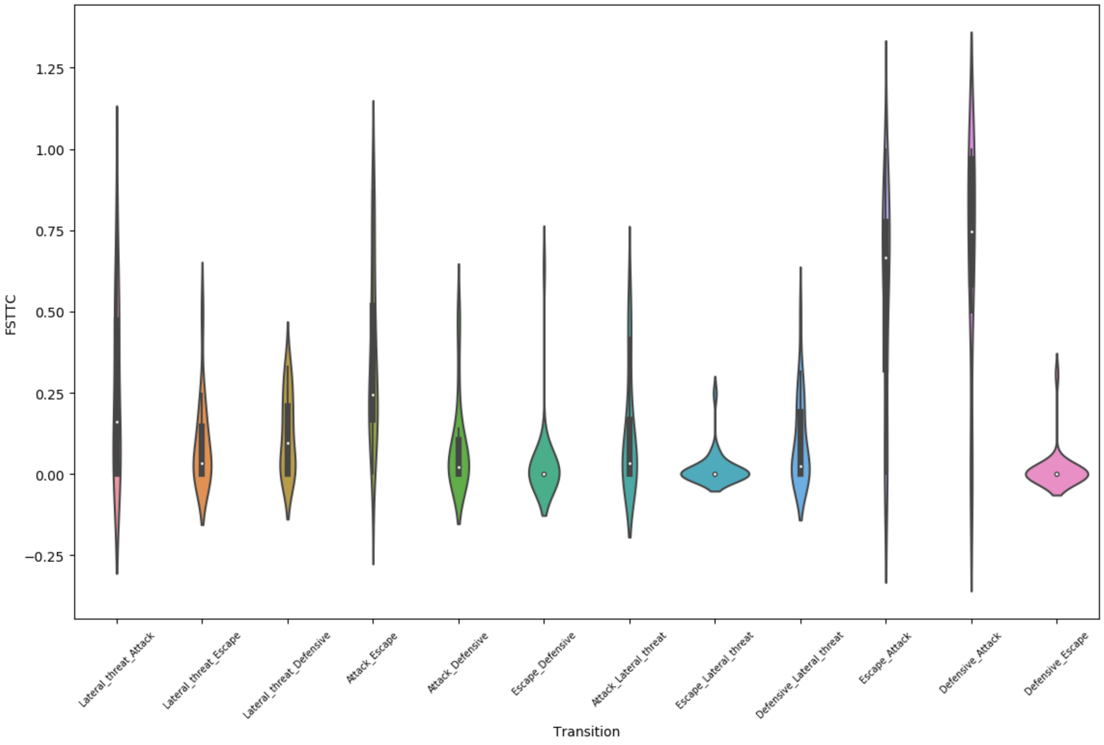
□ Escape

□ Defensive

Calculate FSTTC

In this menu, we need to insert some settings telling SimBA how to perform the FSTTC:

- `Create graph`: If you tick this box, SimBA will create and save a violin plot of the FSTTC representating all the videos in you project. This can be helpful to get a quick overview of the data. The graph will be saved as a time-stamped `.png` file in the `project_folder/logs` directory, and be named something like `FSTTC_20210307071243.png`. It will look something like this image below (click on the image to enlarge), with the FSTTC is on the y-axis, and the dyadic pairwise behaviors on the x-axis. On the x-axis, the first behavior name is behavior A, and the second behavior name is behavior B. Thus, in the image below, the first violin represents the FSTTC for **Lateral threat** behavior followed by **Attack** behavior.

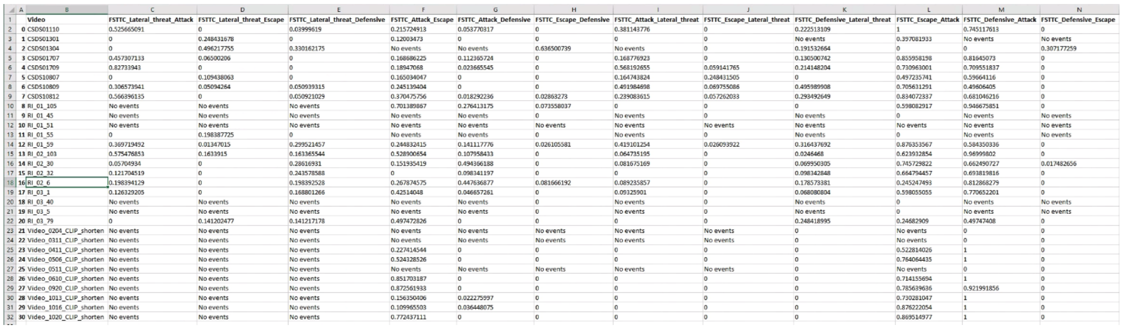


- `Time Delta`: The FSTTC needs a single hyperparameter - time delta - which should be expressed in milliseconds. This represents the length of the time-window following behavior A onset. If the expression of behavior B happens within time-delta, then we will consider behavior B as produced by behavior A. If unsure, start with a time-delta of 2000ms Lee et al, 2019.
- `Behaviors`: Go ahead and tick the behaviors in your project that you want to calculate FSTTC's for. SimBA will calculate FSTTC's for all possible pairwise combinations of the behaviors you tick.

Once done, go ahead and click on `Caluclate FSTTC`. You can follow the progress in the main SimBA terminal window.

3.4.3 Step 3: Interpreting the FSTTC results output.

Once done, inside your `project_folder/logs` directory, you will have a time-stamped CSV file - which is named something like `FSTTC_20210307071243.csv` - and contains the results for each of the videos in your project. If you open it up, it will look like this (click on the image to enlarge):



	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1		Video	FSTTC_Lateral_threat_Attack	FSTTC_Lateral_threat_Escape	FSTTC_Lateral_threat_Defensive	FSTTC_Attack_Escape	FSTTC_Attack_Defensive	FSTTC_Escape_Defensive	FSTTC_Attack_Lateral_threat	FSTTC_Escape_Lateral_threat	FSTTC_Defensive_Lateral_threat	FSTTC_Escape_Attack	FSTTC_Defensive_Attack	FSTTC_Defensive_Escape
2	0	CSDS01110	0.525665091	0	0.03999619	0.215724913	0.053770317	0	0.381143776	0	0.222513109	1	0.745117613	0
3	1	CSDS01301	0	0.248431678	0	0.12003473	0	0	0	0	No events	0.397081933	No events	No events
4	2	CSDS01304	0	0.496217755	0.330162175	No events	No events	0.636500739	No events	0	0.191532664	0	0	0.307177259
5	3	CSDS01707	0.457307133	0.06500206	0	0.168686225	0.112365724	0	0.168776923	0	0.130500742	0.855958198	0.81645073	0
6	4	CSDS01709	0.82733943	0	0	0.18947068	0.023665545	0	0.568192655	0.059141765	0.214148204	0.730963001	0.709551837	0
7	5	CSDS10807	0	0.109438063	0	0.165034047	0	0	0.164743824	0.248431505	0	0.497235741	0.59664116	0
8	6	CSDS10809	0.306573941	0.05094264	0.050939315	0.245139404	0	0	0.491984698	0.069755086	0.495989908	0.705631291	0.49606405	0
9	7	CSDS10812	0.566396135	0	0.050921029	0.370475756	0.018292236	0.02863273	0.239083615	0.057262033	0.293492649	0.834072337	0.681046216	0
10	8	RI_01_105	No events	No events	No events	0.701389867	0.276413175	0.073558037	0	0	0	0.598082917	0.946675851	0
11	9	RI_01_45	No events	No events	No events	No events	No events	0	No events	0	No events	0	No events	No events
12	10	RI_01_51	No events	No events	No events	No events	No events	No events	No events	No events	No events	No events	No events	No events
13	11	RI_01_55	0	0.198387725	0	No events	No events	0	No events	0	No events	0	No events	No events
14	12	RI_01_59	0.369719492	0.01347015	0.299521457	0.244832415	0.141117776	0.026105581	0.419101254	0.026093922	0.316437692	0.876353567	0.584350336	0
15	13	RI_02_103	0.575476853	0.1633915	0.163365544	0.528900654	0.107958433	0	0.064735195	0	0.0246468	0.623932854	0.96999802	0
16	14	RI_02_30	0.05704934	0	0.28616931	0.151935419	0.494366188	0	0.081675169	0	0.069950305	0.745729822	0.662490727	0.017482656
17	15	RI_02_32	0.121704519	0	0.243578588	0	0.098341197	0	0	0	0.098342848	0.664794457	0.693819816	0
18	16	RI_02_6	0.198394129	0	0.198392528	0.267874575	0.447636877	0.081666192	0.089235857	0	0.178573381	0.245247493	0.812868279	0
19	17	RI_03_1	0.126329205	0	0.168801266	0.42514048	0.046657261	0	0.09325901	0	0.068080804	0.598055055	0.770652201	0
20	18	RI_03_40	No events	No events	No events	No events	No events	0	No events	0	No events	0	No events	No events
21	19	RI_03_5	No events	No events	No events	No events	No events	0	No events	0	No events	0	No events	No events
22	20	RI_03_79	0	0.141202477	0.141217178	0.497472826	0	0	0	0	0.248418995	0.24682909	0.49747408	0
23	21	Video_0204_CLIP_shorten	No events	No events	No events	No events	No events	No events	No events	No events	0	No events	0	0
24	22	Video_0311_CLIP_shorten	No events	No events	No events	No events	No events	No events	No events	No events	0	No events	0	0
25	23	Video_0411_CLIP_shorten	No events	No events	No events	0.227414544	0	0	0	0	0	0.522814026	1	0
26	24	Video_0506_CLIP_shorten	No events	No events	No events	0.524328526	0	0	0	0	0	0.764064435	1	0
27	25	Video_0511_CLIP_shorten	No events	No events	No events	No events	No events	No events	No events	No events	0	No events	0	0
28	26	Video_0610_CLIP_shorten	No events	No events	No events	0.851703187	0	0	0	0	0	0.714155694	1	0
29	27	Video_0920_CLIP_shorten	No events	No events	No events	0.872561933	0	0	0	0	0	0.785639636	0.921991856	0
30	28	Video_1013_CLIP_shorten	No events	No events	No events	0.156350406	0.022275997	0	0	0	0	0.730281047	1	0
31	29	Video_1016_CLIP_shorten	No events	No events	No events	0.109965503	0.036448075	0	0	0	0	0.876222054	1	0
32	30	Video_1020_CLIP_shorten	No events	No events	No events	0.772437111	0	0	0	0	0	0.869514977	1	0

This file contains one row for every video in your project, and one column for every pairwise behavior combination of the behaviors you selected in **Step 2** above. So, for example, if you want the FSTTC for video *CSDS01110* and *Lateral threat* -> *Attack*, look in cell C2.

- If the video contained no expression of behavior B, but expression of behavior A, then the FSTTC cell value will read `0`.
- If the video contained no expression of behavior A, and no expression of behavior B, then the FSTTC cell value will read `No events`.

Note: If you ticked the box for *Create graph* in **Step 2** above, you will also have a violin plot in *project_folder/logs* directory. In this violin plot, videos that has neither expression of behavior A nor expression of behavior B have been omitted from the graph calculation.

If you bump into any issues using the FSTTC, let on know by opening an issue or reach out to us on Gitter

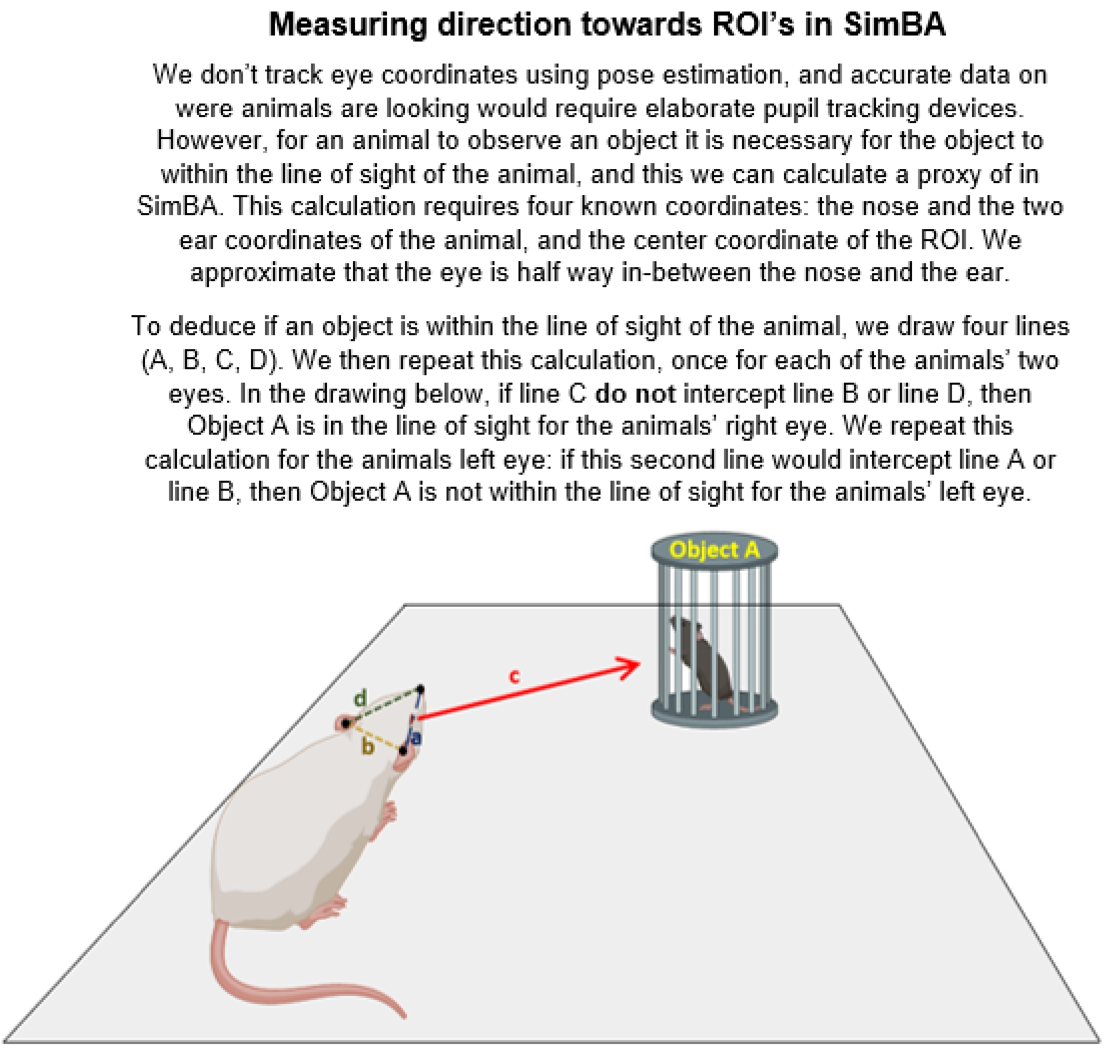
3.5 Who is looking at who: calculate "directionality" between animals in SimBA

In videos containing multiple animals, it may be valuable to know when, and how much time, each animal spend directing towards each other or towards stationary objects. We can calculate this with appropriate pose-estimation data in SimBA. For a better understanding of this kind of data, and if it is relevant for your specific application, see the videos and gifs below and the rendered visualization examples at the end of this tutorial. Here we use SimBA to get measures, in seconds, of much time *Animal 1* spend directing towards *Animal 2*, and how much time *Animal 2* spend directing towards *Animal 1* (. . . and so on for all the relationships for all the animals tracked in each video). We can also use SimBA to generate visualizations of these metrics, so that we can be comfortable with that the "directionality" data that is summarized in the created CSV files and dataframes are accurate.

Note:

1. Firstly, and importantly, SimBA does not calculate actual *gaze* (this is *not* possible using **only** pose-estimation data). Instead, SimBA use a proxy calculation for estimating if the animal is directing toward another animal, or a user-defined region of interest, using the nose coordinate and the two ear-coordinates (or the equivalent coordinate on any other animal species). For more information on how SimBA estimates the location of the animals *eyes*, and how SimBA estimates if an object, or another animal, is within the *line of sight* of the specific animal, check out the [SimBA ROI tutorial - Part 3].
2. The pose-estimation tracking of the two *ears*, and the *nose*, of each animal, or the equivalent in other species, is required for this to work. See more information on this pre-requisite below. Furthermore, see the below image and its legend for more information on how the code in SimBA estimates the relationships between animal body-parts and/or user-defined ROIs.

[Click here for better-quality rendered MP4]



Measuring direction towards ROI's in SimBA

We don't track eye coordinates using pose estimation, and accurate data on were animals are looking would require elaborate pupil tracking devices. However, for an animal to observe an object it is necessary for the object to within the line of sight of the animal, and this we can calculate a proxy of in SimBA. This calculation requires four known coordinates: the nose and the two ear coordinates of the animal, and the center coordinate of the ROI. We approximate that the eye is half way in-between the nose and the ear.

To deduce if an object is within the line of sight of the animal, we draw four lines (A, B, C, D). We then repeat this calculation, once for each of the animals' two eyes. In the drawing below, if line C **do not** intercept line B or line D, then Object A is in the line of sight for the animals' right eye. We repeat this calculation for the animals left eye: if this second line would intercept line A or line B, then Object A is not within the line of sight for the animals' left eye.

3.5.1 Before analyzing directionality in SimBA

To analyze directionality data in SimBA (for descriptive statistics, machine learning features, or both descriptive statistics and machine learning features) the tracking data **first** has to be processed the up-to and including the *Outlier correction* step described in Part 2 - Step 4 - Correcting outliers . Thus, before proceeding to calculate directionality-based measurements between animals in SimBA, you should have one CSV file for each of the videos in your project located within the `project_folder\csv\outlier_corrected_movement_location` sub-directory of your SimBA project.

Specifically, for working with directionality between animal in SimBA, begin by

(i) Importing your videos to your project

(ii) Import the tracking data and relevant videos to your project

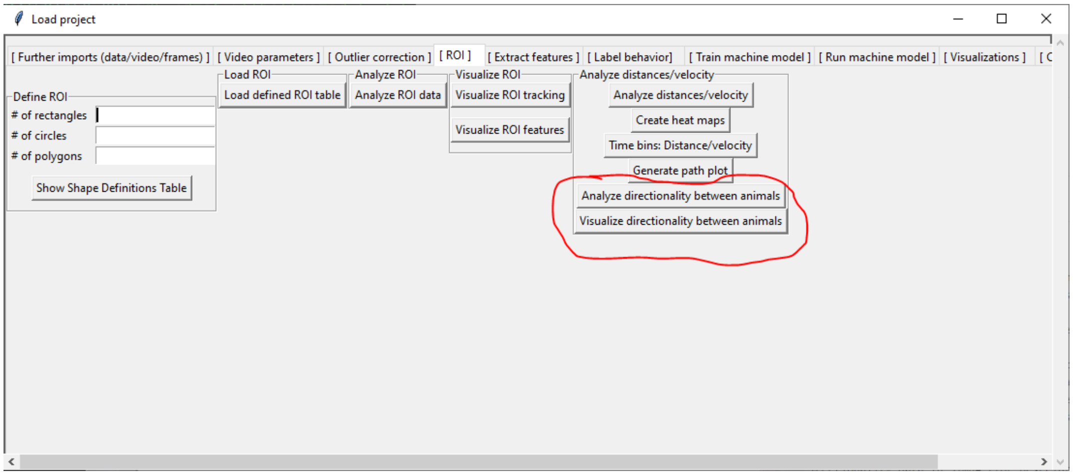
(iii) Set the video parameters

(iv) Correct outliers

3.5.2 Calculate what other body-parts and what other animals, are in the line of sight for specific animal.

1. In the main SimBA console window, begin loading your project by clicking on `File` and `Load project`. In the **[Load Project]** tab, click on `Browse File` and select the `project_config.ini` that belongs to your project.

2. Navigate to the ROI tab. On the right hand side, within the `Analyze distances/velocity` sub-menu, there are two buttons (circled in red in the image below) - (i) *Analyze directionality between animals* and (ii) *Visualize directionality between animals*.

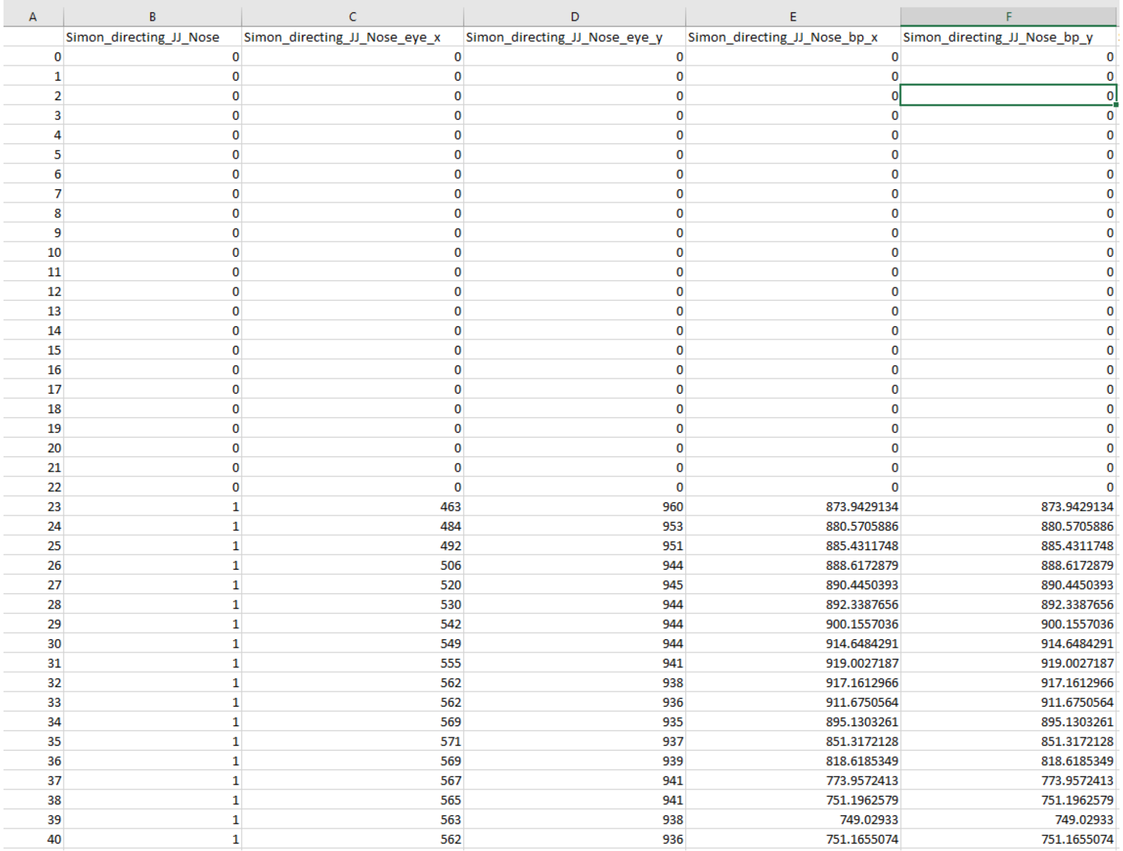


3. We will begin clicking on the `Analyze directionality between animals` button. Clicking on this button will calculate how much time each animal spend directing towards each other within all of your videos represented as CSV files within in your `project_folder/csv/outlier_corrected_movement_location` directory. Again, for more information on how SimBA calculates directionality, check out this schematic. This schematic depicts a user-defined ROI, but exactly the same methods is applied when using body-parts rather than a user-defined ROI regions.

Once you have clicked the `Analyze directionality between animals` button, you can follow the calculation progress in the main SimBA terminal window.

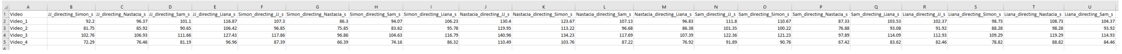
Once the calculations are complete, several new files will be generated in your SimBA project directory. First, a new folder has been created in your project directory. If you look within the `project_folder/csv/directionality_dataframes` directory, you should see one new file for each of the videos in your project, each file should have the name of the video that was analyzed. If you open up one of these files, then you should see one row for each frame in your video, together with a bunch of columns representing the different animal relationships. For each animal (and by *animal* I mean the animal "looking" at something) and target (and by *target* I mean the body-part being "looked" at or directed towards), there will be a total of 5 columns. If we look at the **first five** column in example image just below, then the **first of these five columns** has the header *Simon_directing_JJ_Nose* and contains boolean 0 or 1 values. If the specific animal (in this case Simon) directed towards the specific body-part (in this case JJs's nose) in that frame (i.e., the animal called Simon had the animal JJs nose within the line of sight) then the column will read *1*. Conversely, if the animal (Simon) did **not** direct towards that body-part (JJ's nose) in that frame (i.e., JJ's nose was **not** within Simon's line of sight), then the column reads a *0*. Thus, in this example image below, animal JJs nose **was** within the line of sight of the animal Simon in frame 23-40, but was **not** within the line of sight of Simon in frame 0-22.

Next, the preceeding four columns are primarily saved for visualization purposes, should the user want to verify the data by genering rendered videos of the different "line of sight" paths. The **second** and **third** column (*Simon_directing_JJ_Nose_eye_x* and *Simon_directing_JJ_Nose_eye_y*) stores the coordinates of animal *Simon's* directing eye when the specific body-part was within the line of sight. If the specific body-part was **not** within the line of sight, then the specific row for these columns will read *0*. Lastly, the **fourth** and **fifth** column (*Simon_directing_JJ_Nose_bp_x* and *Simon_directing_JJ_Nose_bp_y*) stores the coordinates of the body-part being observed by animal Simon. If the specific body-part was **not** within the line of sight, then the specific row for these columns will also read *0*.



A	B	C	D	E	F
	Simon_directing_JJ_Nose	Simon_directing_JJ_Nose_eye_x	Simon_directing_JJ_Nose_eye_y	Simon_directing_JJ_Nose_bp_x	Simon_directing_JJ_Nose_bp_y
0	0	0	0	0	0
1	0	0	0	0	0
2	0	0	0	0	0
3	0	0	0	0	0
4	0	0	0	0	0
5	0	0	0	0	0
6	0	0	0	0	0
7	0	0	0	0	0
8	0	0	0	0	0
9	0	0	0	0	0
10	0	0	0	0	0
11	0	0	0	0	0
12	0	0	0	0	0
13	0	0	0	0	0
14	0	0	0	0	0
15	0	0	0	0	0
16	0	0	0	0	0
17	0	0	0	0	0
18	0	0	0	0	0
19	0	0	0	0	0
20	0	0	0	0	0
21	0	0	0	0	0
22	0	0	0	0	0
23	1	463	960	873.9429134	873.9429134
24	1	484	953	880.5705886	880.5705886
25	1	492	951	885.4311748	885.4311748
26	1	506	944	888.6172879	888.6172879
27	1	520	945	890.4450393	890.4450393
28	1	530	944	892.3387656	892.3387656
29	1	542	944	900.1557036	900.1557036
30	1	549	944	914.6484291	914.6484291
31	1	555	941	919.0027187	919.0027187
32	1	562	938	917.1612966	917.1612966
33	1	562	936	911.6750564	911.6750564
34	1	569	935	895.1303261	895.1303261
35	1	571	937	851.3172128	851.3172128
36	1	569	939	818.6185349	818.6185349
37	1	567	941	773.9572413	773.9572413
38	1	565	941	751.1962579	751.1962579
39	1	563	938	749.02933	749.02933
40	1	562	936	751.1655074	751.1655074

Perhaps more importantly, SimBA generates a CSV log file that contains summary statistics. This includes how much time each animal spent directing towards all other animals present in the video. This file can be found in the `project_folder/logs` directory. This filename of this file is time-stamped, and have a name that may look something like this: `Direction_data_20200822151424.csv`. The content of this file should look like the image just below (if you are tracking 5 animals. if you are tracking fewer animals, then the file will contain fewer columns):



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
1	Video	JJ_directing_Simon_s	JJ_directing_Nastacia_s	JJ_directing_Sam_s	JJ_directing_Liana_s	Simon_directing_JJ_s	Simon_directing_Nastacia_s	Simon_directing_Sam_s	Simon_directing_Liana_s	Nastacia_directing_JJ_s	Nastacia_directing_Simon_s	Nastacia_directing_Sam_s	Nastacia_directing_Liana_s	Sam_directing_JJ_s	Sam_directing_Simon_s	Sam_directing_Nastacia_s	Sam_directing_Liana_s	Liana_directing_JJ_s	Liana_directing_Simon_s	Liana_directing_Nastacia_s	Liana_directing_Sam_s
2	Video_1	92.2	96.37	101.1	116.87	107.3	86.3	94.07	106.23	130.4	123.67	107.13	96.83	111.8	110.67	87.33	103.53	102.37	98.73	108.73	104.37
3	Video_2	81.75	85.92	90.65	106.42	96.85	75.85	83.62	95.78	119.95	113.22	96.68	86.38	101.35	100.22	76.88	93.08	91.92	88.28	98.28	93.92
4	Video_3	102.76	106.93	111.66	127.43	117.86	96.86	104.63	116.79	140.96	134.23	117.69	107.39	122.36	121.23	97.89	114.09	112.93	109.29	119.29	114.93
5	Video_4	72.29	76.46	81.19	96.96	87.39	66.39	74.16	86.32	110.49	103.76	87.22	76.92	91.89	90.76	67.42	83.62	82.46	78.82	88.82	84.46
6																					

Thi log file will contain one column for each animal relationship, and one row for each analysed video. In the example screenshot above, the first column header reads *JJ_directing_Simon_s* and contains the time, in seconds, that the animal with the ID *JJ* spent directing towards the animal with the ID *Simon* (92.2s in the video named Video1). In column F (or column 5), with the header *Simon_directing_JJ_s*, we can read the time, in seconds, that the animal with the ID *Simon* spent directing towards the animal with the ID *JJ* (107.3s in Video1). Indeed, studying the directionality results in all of the four videos in the example screenshot above, the animal *Simon* does seems a little more interested in directing towards the animal named *JJ*, than the animal *JJ* is interested in directing towards the animal named *Simon*.

Note: These summary statistics are calculated based on all body-parts for each animal. If an animal is directing **any** body-part belonging to a specific other animal, then the animal is directing towards that other specific animal. For example, even if the animal with the ID *Simon* only has the tail-base of the animal with the ID JJ in the *line of sight* in one frame, the animal with the ID *JJ* is still counted as being in the *line of sight* in of the animal with the ID *Simon*.

4. Next, we may to visualize these directionality data - for peace of mind - that the metrics seen in the output desriptive statistics and CSV files outlined above are accurate and plausable. To do this, ga ahead and click on the second button described above - *Visualize directionality between animals*.

Note: Rendering videos are time-consuming and computaionally expensive. If you have many videos, many animals, and high fps/resolution, it might take some time. You can follow the progress in the main SimBA terminal window.

Once complete, you can find your final rendered videos in your `project_folder\frames\output\ROI_directionality_visualize` folder. For higher-quality examples of the expected final output videos for experimental scenarios containing five animals, or two animals, see the SimBA YouTube playlist.

Example 1 - 5 mice on YouTube

Example 2 - 2 mice on YouTube

3.6 SimBA behavioral annotator GUI

The behavioural annotator GUI is used to label (annotate) frames extracted from experimental videos that contain behaviors of interest. SimBA appends the behavioral annotations directly to the pose-estimation tracking dataframe to build supervised machine learning predictive classifiers of behaviors of interest. Specifically, this GUI integrates two additional consoles with the main SimBA console: (1) a video player and (2) a frame viewer. The video player and frame viewer are synced, such that pausing or starting the video will advance the frame viewer to the identical frame that the video was paused or started at. The frame viewer is then used to annotate when behaviors of interest are present or absent within the given frame. Such annotations can be flexibily annotated from single to numerous frames using the annotation interface. Please see "**Step 3. Labelling**" below for detailed instructions on these functions.

Note that SimBA performs similar functions such as the open-source JWatcher or commercial Noldus Observer systems, with the exception that SimBA automates the backend integration of behavioral annotation with creating predictive classifiers. If you already have such annotations stored in alterantive file formats, like JWatcher or Noldus Observer, they can be appended directly to the tracking data and no behavioral annotations needs to be done in SimBA. For example, the Crim13 dataset was annotated using Piotr's Matlab Toolbox and we appended the annotations to the tracking data using a version of this script.

If you already have annotation videos created with these alternative tools, or any other behavioral annotator, and would like to use them to create predictive classifiers, please let us know as we would like to write scripts that could process these data for SimBA. If you have created such scripts yourself, please consider contributing them to the community!

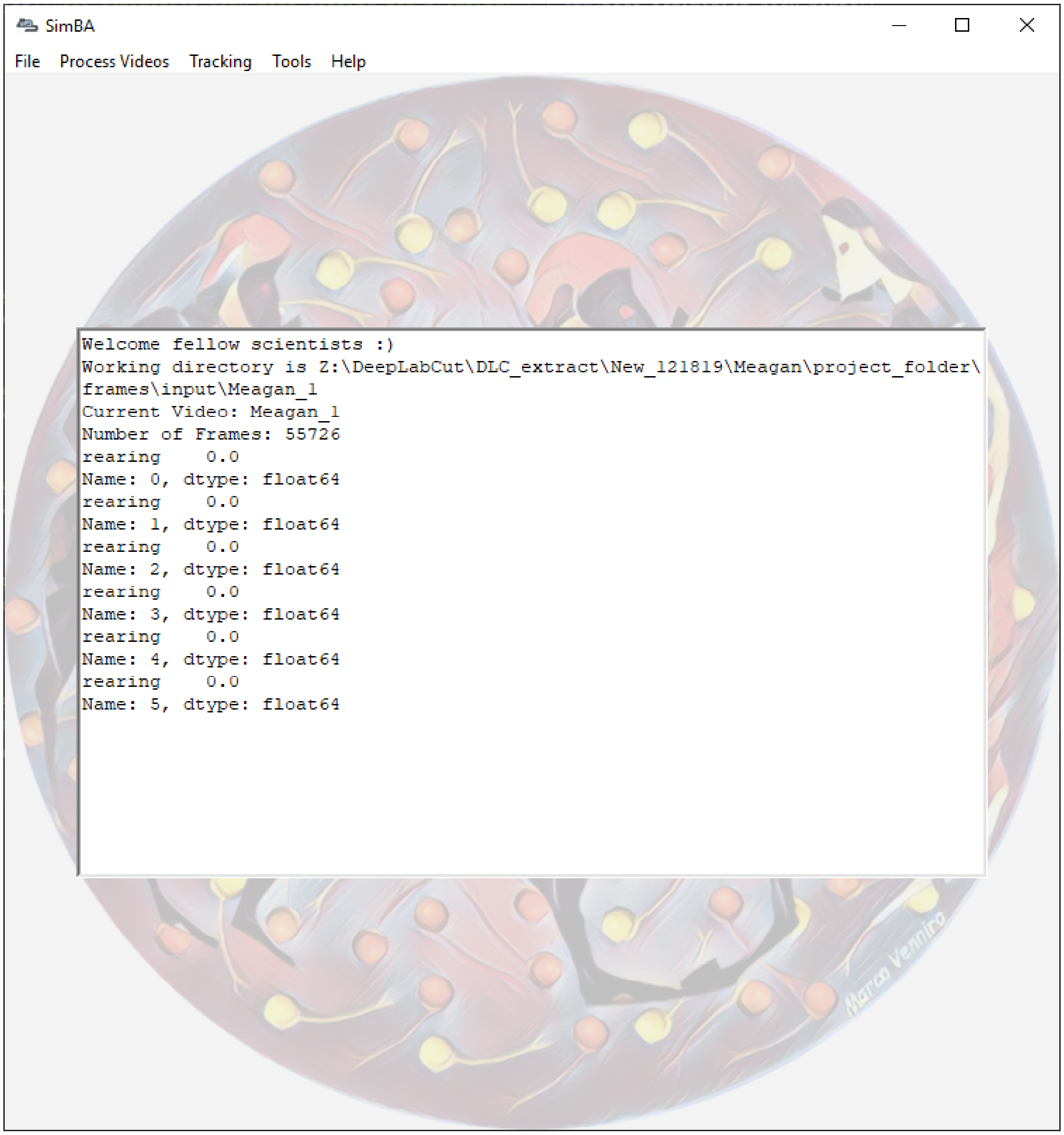
We will provide support in either scenario.

3.6.1 Step 1. Loading project_config file

In the main SimBA menu, click on `File > Load Project > Load Project.ini > Browse File` and select the config file (project_config.ini) for the current project. This step **must** be done before proceeding to the next step.

3.6.2 Step 2. Opening labelling behavior window

In the 'Load project' window, under `Label Behavior` click on `Select video (create new video annotation)`. This will prompt you to select a **vidoe file**. If you have previously annotated some frames and would like to continue your annotation, click on `Select video (continue existing video annotation)`

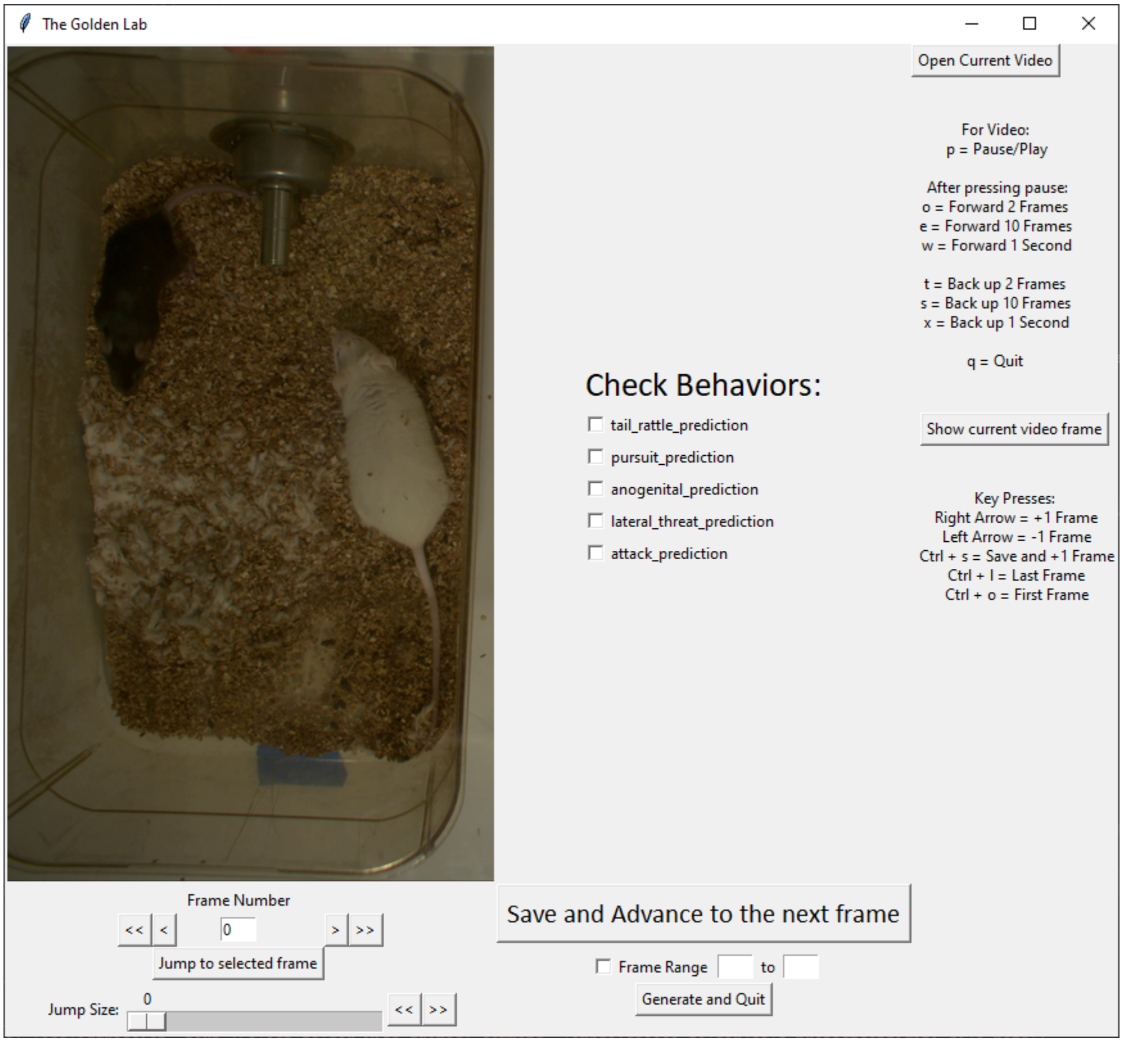


```
Welcome fellow scientists :)
Working directory is Z:\DeepLabCut\DLC_extract\New_121819\Meagan\project_folder\
frames\input\Meagan_1
Current Video: Meagan_1
Number of Frames: 55726
rearing    0.0
Name: 0, dtype: float64
rearing    0.0
Name: 1, dtype: float64
rearing    0.0
Name: 2, dtype: float64
rearing    0.0
Name: 3, dtype: float64
rearing    0.0
Name: 4, dtype: float64
rearing    0.0
Name: 5, dtype: float64
```

Note: At any time, the user can refer to the main screen window to see the values of the frame. In the picture above, **Name** is the current frame number.

3.6.3 Step 3. Labelling

Under the **Check Behaviors** heading is a list of checkboxes, one for each classifier that were specified when the project was created. Users can check (or uncheck) the boxes if the frame displays the behaviors. When the user press *Save and Advance to the next frame* the behavioral data for that particular frame is saved.



For Video:
p = Pause/Play

After pressing pause:
o = Forward 2 Frames
e = Forward 10 Frames
w = Forward 1 Second

t = Back up 2 Frames
s = Back up 10 Frames
x = Back up 1 Second

q = Quit

Check Behaviors:

- □ tail_rattle_prediction
- □ pursuit_prediction
- □ anogenital_prediction
- □ lateral_threat_prediction
- □ attack_prediction

Key Presses:
Right Arrow = +1 Frame
Left Arrow = -1 Frame
Ctrl + s = Save and +1 Frame
Ctrl + l = Last Frame
Ctrl + o = First Frame

Frame Number: 0

Jump Size: 0

□ Frame Range to

Navigation

Underneath the displayed image, there are options for navigating through all the frames:

Using Mouse

<< jump to the first image frame in the folder

>> jump to the last image frame in the folder

< go back to the previous frame

> advance to the next frame

`Jump Size` You can select a range of frames to advance or jump backwards. When you then press the << or >> buttons to the right of the scale, the frames advance or go back the Jump Size frame count indicated.

`Jump to selected frame` You have the option to manually enter a frame number into this entry box, which will then display the corresponding image.

`Frame Range` By selecting this box and entering a range of numbers in the adjacent entry boxes, saving and advancing to the next frame will save all frames in the Frame Range, inclusive, as containing the marked behaviors.

`Generate and Quit` This will compile and export the data to a .csv file located in the `project folder\csv\targets_inserted\` that contain the behavioral annotations.

Keyboard Shortcuts

Keyboard shortcut information is displayed on the right side of the window for ease of use.

`For Video` shortcuts are only applicable for navigating the video when opened, and have no effect on the displayed frame.

`Key Presses` allows users to jump from frame to frame as well as saving frame information by using the keyboard.

Tip: If using Ctrl + S to save, do NOT hold down keys to save multiple frames. Please just hit multiple times.

Playing Video

`Open Current Video` Pressing this button will open the video that corresponds to the frame folder that is being analyzed. Refer to keyboard shortcuts to pause/play and move forward or backwards in the video by a certain amount of frames.

`Show current video frame` This will display the current frame of the paused video on the labelling screen.

Note: Video name must be the same as the folder name. The video must also be paused by pressing `p` before any other keyboard command (such as moving forward or backwards a set number of frames).

3.7 Regions of Interest (ROIs) in SimBA

The SimBA region of interest (ROI) interface allows users to define and draw ROIs on videos. ROI data can be used to calculate basic descriptive statistics based on animals movements and locations (such as how much time the animals have spent in different ROIs, or how many times the animals have entered different ROIs). Furthermore, the ROI data can be used to build potentially valuable, additional, features for random forest predictive classifiers. Such features can be used to generate a machine model that classify behaviors that depend on the spatial location of body parts in relation to the ROIs. **CAUTION**: If spatial locations are irrelevant for the behaviour being classified, then such features should *not* be included in the machine model generation as they just only introduce noise.

3.7.1 Before analyzing ROIs in SimBA

To analyze ROI data in SimBA (for descriptive statistics, machine learning features, or both descriptive statistics and machine learning features) the tracking data **first** has to be processed the up-to and including the *Outlier correction* step described in [Part 2 - Step 4 - Correcting outliers]. Thus, before proceeding to calculate ROI based measures, you should have one CSV file for each of the videos in your project located within the `project_folder\csv\outlier_corrected_movement_location` sub-directory of your SimBA project.

Specifically, for working with ROIs in SimBA, begin by

(i) Importing your videos to your project

(ii) Import the tracking data and relevant videos to your project

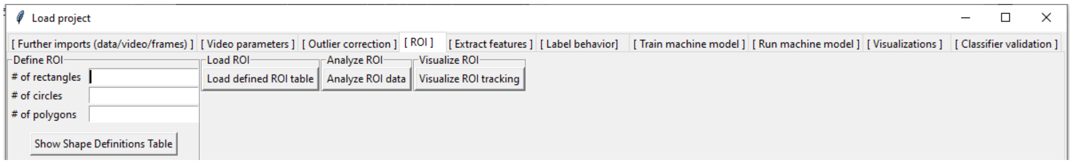
(iii) Set the video parameters

(iv) Correct outliers

A short explanation on what is meant by "using ROI data as features for random forest classifiers" When ROIs have been drawn in SimBA, then we can calculate several metrics that goes beyond the coordinates of the different body-parts from pose-estimation. For example - for each frame of the video - we can calculate the distance between the different body-parts and the different ROIs, or if the animal is inside or outside the ROIs. In some instances (depending on which body parts are tracked), we can also calculate if the animal is `directing towards the ROIs <https://github.com/sgoldenlab/simba/blob/master/docs/ROI_tutorial.md#part-3-generating-features-from-roi-data>`_(see below for more information). These measures can be used to calculate several further features such as the percentage of the session, up to and including the any frame, the animal has spent within a specific ROI. These and other ROI-based features could be useful additions for classifying behaviors in certain scenarios.

3.7.2 Part 1. Defining ROIs in SimBA

1. In the main SimBA console window, begin loading your project by clicking on `File` and `Load project`. In the **[Load Project]** tab, click on `Browse File` and select the `project_config.ini` that belongs to your project.
2. Navigate to the **ROI** tab, which should look like this:

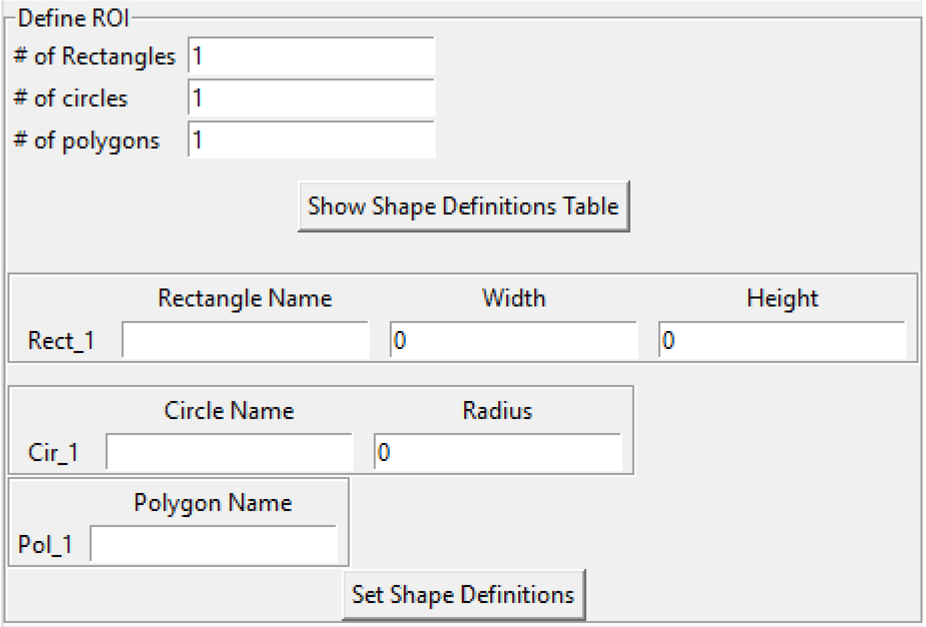


3. Begin at the left hand side of the **ROI** tab: using the sub-menu **Define ROI**. This menu lets you define **the number of rectangles, circles, and polygons** that you which to draw on the videos in the project.

Note: All videos in the project needs to contain the same number of shapes. For example, if you wish to analyze four ROIs in the first video of your project, but only three ROIs in the second video of your project, then you still need to draw four ROIs for both the first and second video in your project.

In the entry boxes associated with each shape, enter the number of each shape that you wish to draw. If you do *not* want to draw a specific shape, then leave the entry box empty.

4. Once you've entered the number of shapes that you wish to draw, then click on the `Show Shape Definitions Table` button and the following table will pop up.



Define ROI

# of Rectangles 1

# of circles 1

# of polygons 1

	Rectangle Name	Width	Height
Rect_1		0	0

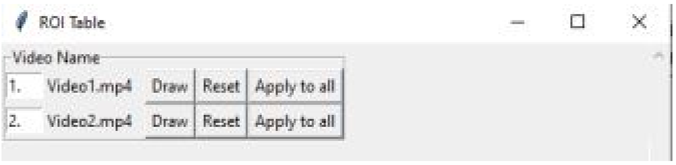
	Circle Name	Radius
Cir_1		0

	Polygon Name
Pol_1	

In this table, enter a name for each of your shapes in the Rectangle Name, Circle Name, and Polygon Name entry boxes. Make sure that the names of each of the shapes are different from each other - e.g., there cannot be teo shapes that are both named `Area_1`. When all the shape names have been entered, click on the `Set Shape Definitions` button which is located at the bottom of the `Shape Definitions Table`.

Note: Each *Circle* and *Rectangle* row has further size-related columns associated with them: Width, Height, or Radius. These columns are set to 0 and should, for now, **remain set to 0**. These settings will - **in future version of SimBA** - be used to draw shapes that are of precise metric sizes (e.g., a 40x40mm rectangle).

5. After clicking on the `Set Shape Definitions` button, the following table will pop open.



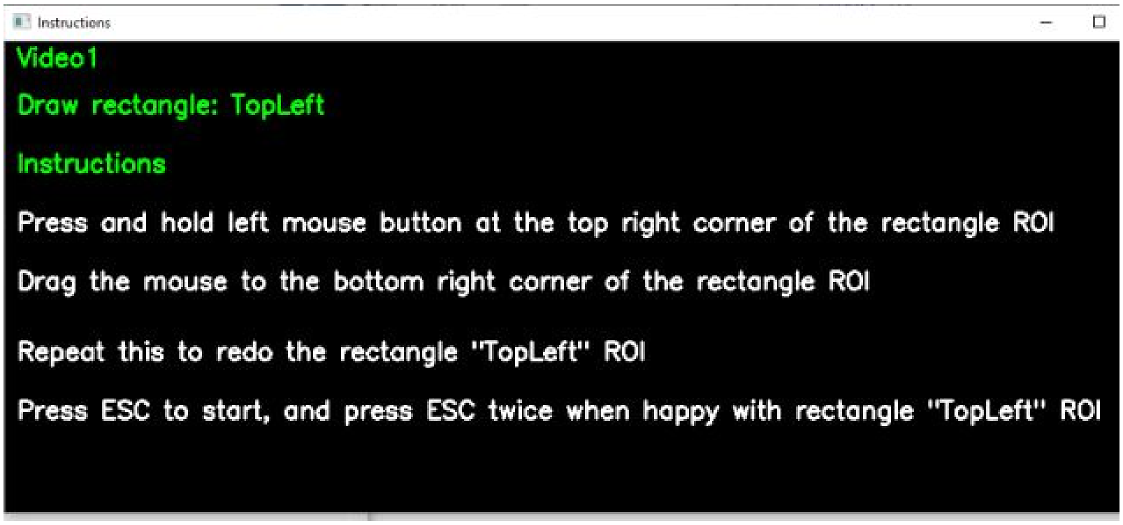
Video Name

1.	Video1.mp4	Draw	Reset	Apply to all
2.	Video2.mp4	Draw	Reset	Apply to all

This table contain one row for each of the videos in the project. Each video in the project has three buttons associated with it: **Draw, Reset,** and **Apply to all**. The functions associated with each button is described in detail below. However, in brief, the **Draw** button allows you to start to draw the defined shapes on the specific video. If drawings already exist for the specific video, then the **Draw** buttons opens an interface where you can move the shapes that has previously been drawn on the video. The **Reset** button deletes any drawing made on the specific video and allows you to restart the drawings from scratch by next clicking on the **Draw** button. The **Apply to all** buttons copies the drawing made on the specific video and replicates it on all other videos in the project. If a drawing has been applied to all videos videos in the project by clicking on the **Apply to all** button, then the shapes for any specific video can be moved by clicking on the **Draw** button.

6. To begin to draw your shapes, click on the **Draw** button for the first video in the table. Once clicked, a window will first pop up that contains instructions for how to draw your shapes. The instructions for drawing rectangles, circles, and polygons are slightly different and are described below.

Any user-defined regles will be processed first. The instructions for drawing rectangles will look like the image below on the left, and the process of drawing the rectangle should look like the gif image below:



Video1

Draw rectangle: TopLeft

Instructions

Press and hold left mouse button at the top right corner of the rectangle ROI

Drag the mouse to the bottom right corner of the rectangle ROI

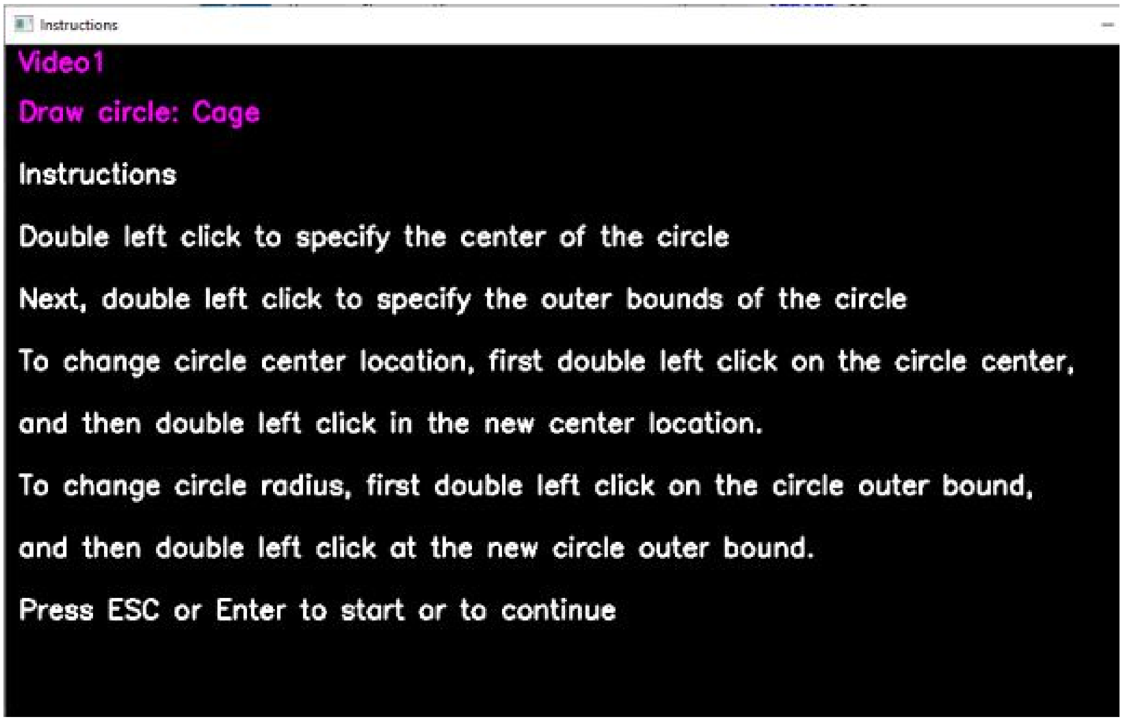
Repeat this to redo the rectangle "TopLeft" ROI

Press ESC to start, and press ESC twice when happy with rectangle "TopLeft" ROI

In this example, to begin to draw the rectangle named “TopLeft” on Video1, press `Esc`. When `Esc` is pressed, the first frame of Video1 is displayed. Click and hold the left mouse button at the top left corner of rectangle “TopLeft” and drag the mouse to the bottom right corner of the rectangle. If you’re unhappy with your rectangle, start to draw the rectangle again by holding the left mouse button at the top left corner of your, new, revised, rectangle. The previous rectangle will be automatically discarded. When you are happy with your rectangle, press `Esc` twice to save your rectangle and move to your next shape.

Note: If the image of the first frame of the video looks distorted, and the aspect ratio appears to be off, then grab the side of the window that contains your frame and correct the aspect ratio. If the first frame of your video contains the experimenter hand, or no animal(s) are visible (because the camera was turned on before the animal(s) was placed in the arena), then consider going back and pre-process your videos appropriately

Once all user-defined rectangles have been drawn, any user-defined circles will be processed. The instructions for drawing circles will look like the image below on the left, and the process of drawing the a circle should look like the gif image below:



Video1

Draw circle: Cage

Instructions

Double left click to specify the center of the circle

Next, double left click to specify the outer bounds of the circle

To change circle center location, first double left click on the circle center,

and then double left click in the new center location.

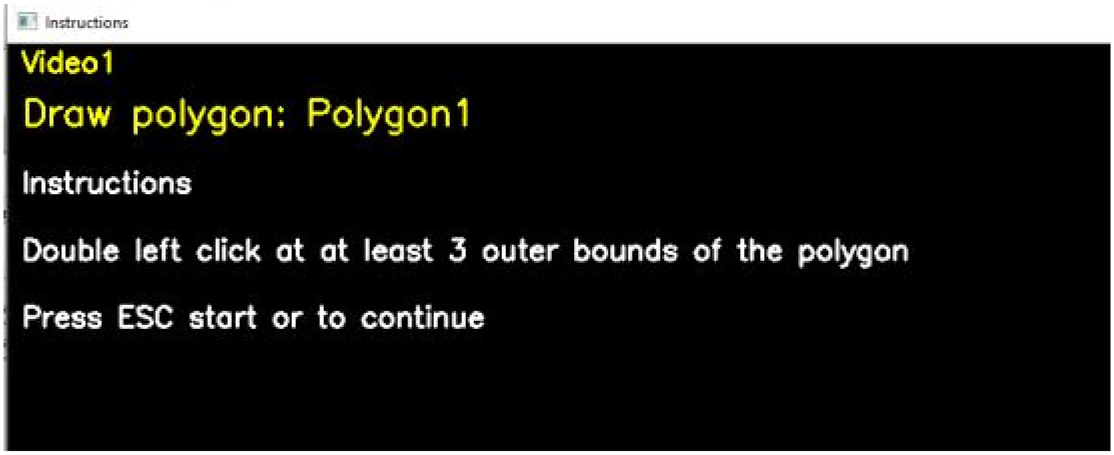
To change circle radius, first double left click on the circle outer bound,

and then double left click at the new circle outer bound.

Press ESC or Enter to start or to continue

In this example, to begin to draw the circle named "Cage" on Video1, press on Esc. When Esc is pressed, the first frame of Video1 is displayed. If you have drawn any rectangles previously, then these rectangles will also be displayed on the frame. Begin by double left mouse clicking on the image where you would like the center of the circle to be. Next, double left click on the image where you would like the outer bound of the circle to be. If you are unhappy with the center of the circle, then double left mouse click on the center of the circle, and double left mouse click again at the new circle center location. Likewise, if you are unhappy with the outer bounds and the diameter of the circle, then double left mouse click on outer bounds of the circle, and then double left click again at the new circle outer bounds location. When you are happy with your circle, press Esc to save your circle and move to the next user-defined shape.

Once all the circles have been drawn, any user-defined polygons will be processed. The instructions for drawing polygons will look like the image below on the left, and the process of drawing the polygon should look like the gif image below:



Video1

Draw polygon: Polygon1

Instructions

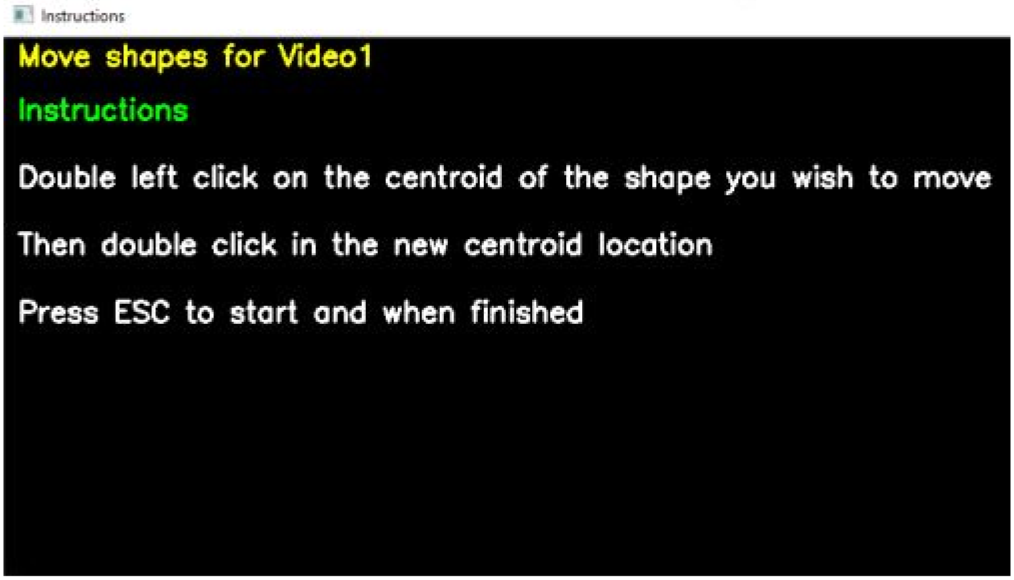
Double left click at at least 3 outer bounds of the polygon

Press ESC start or to continue

In this example, to begin to draw the polygon named "Polygon1" on Video1, press Esc. When Esc is pressed, the first frame of Video1 is displayed. If you have previously drawn any rectangles or circles, then these rectangles and/or circles will also be displayed on the frame. Click on at least three different locations in the image that defines the outer bounds of "Polygon1". SimBA does currently not allow re-definitions of the bound locations of the polygon once they have been specified. If you are unhappy with your drawn polygon, you can chose to either (i) move the entire polygon in a later step (see Step 6, just below), or **Reset** the drawings and start again (again see Step 6, just below). When you have finished your polygon, press `Esc` to save your polygon and move to the next user-defined shape (or to finish, if all user-defined shapes have been drawn).

6. If you have finished the drawings for Video1, and you are not entirely happy with the drawings, you can either: (i) click on the **Reset** button next to Video1: this will delete the drawings you have made on Video1, and you can now again click on **Draw** button next to Video1 to restart your drawing as described in Step 5 above. Alternatively, you can click on **Draw** again **without** first clicking on the **Reset** button. This allows you to move the shapes in Video1 that you have previously drawn.

When you click on **Draw** again, **without** first clicking on the **Reset** button, then the following instruction window will pop open (below left), and moving the shapes should look like this below:



Move shapes for Video1

Instructions

Double left click on the centroid of the shape you wish to move

Then double click in the new centroid location

Press ESC to start and when finished

To start moving the shapes that has previously been drawn, first click on the centroid belonging to the shape that you wish to move. This will initially cause the shape to disappear. Next, double left click on the new centroid location of the shape you are moving. This will display the shape in its new location. When you are happy with the locations of all of your shapes, then press on `Esc` to save the new shape locations.

7. Once all the shapes have been drawn on a single video there are two possible routes to getting the shapes drawn on all of the videos in the project. The first alternative, which is likely the more time-consuming option, is to click on each of the **Draw** buttons associated with each video, and replicate **Step 5** for all of the videos in your project. The second route, and likely the faster and more standardized option, is to replicate the shapes drawn on Video1 in all of the videos of the project, and then nudge them to the correct location in each of the videos (if the camera/arena moved *slightly* across recordings).

Note: If the camera moved **significantly** across the different recordings in the project, the first alternative route described above is the only option to generate accurate all the ROI shapes in each video.

To replicate the ROI shapes drawn in Video1 to all of the videos in the project, click on the **Apply to all** button next to Video1. Once done, click on **Draw** for Video2. When you click on **Draw** for Video2, a window will pop open with Instructions for how to move shapes - press `Esc` to proceed. When pressing `Esc` the first frame of Video2 will be displayed. Now move the shapes so that they are in the correct locations (as described in Step 6 above), and press *Esc* to save the locations of the shapes.

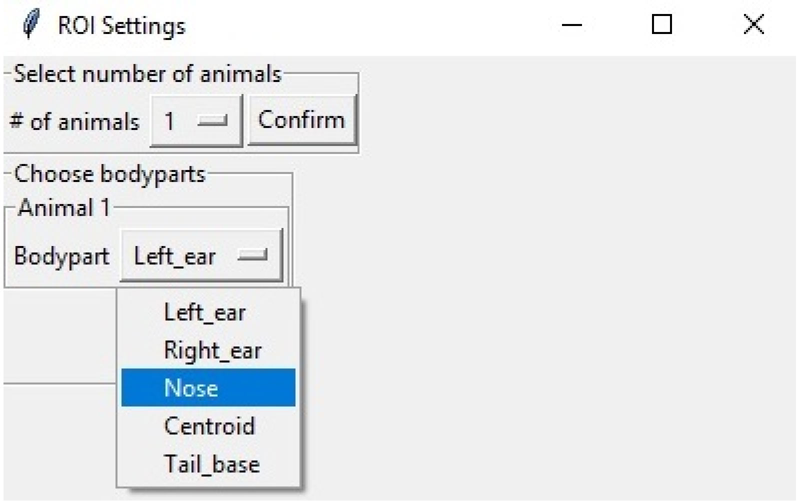
Repeat this step for all of the videos in the project. Once complete, close the **ROI Table** window, and navigate back to the **ROI** tab in the **Load project** window.

Note: If you at any point want to look at, or modify, the defined ROI shapes for each of the videos, click on the `Load defined ROI table` button in the **ROI** tab under the `Load project` menu.

Likewise, if you'd like to delete all the shapes you have drawn for all of the videos and start fresh with new shapes, go back and repeat the above [Part 1 - Step 3]

3.7.3 Part 2. Analyzing ROI data

1. You can now generate CSV files containing descriptive statistics for the the number entries into each ROI shape, and the time spent within each ROI. SimBA allows you to generate these measures for up-to 2 body-parts (or two animals) in any one video for any one analysis. If you would like to get these measures for more than 2 body-parts, then we suggest that you simply run the analysis multiple times.
2. To analyze your ROI data, first click on `Analyze ROI data` in the `Analyze ROI` sub-menu. The following menu will pop open:



# of animals 1

Bodypart Left_ear

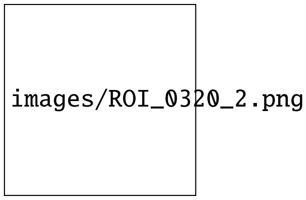
Here, begin by selecting the number of animals (or body-parts) you wish to produce the ROI descriptive statistics for, and then click confirm. A second sub-menu will appear below named `Choose bodyparts`. This menu will contain as many dropdown menus as the number of animals (or body-parts) selected in the `Select number of animals` menu. Each of these drop-down menus will contain the body-parts used to track the animals in pose-estimation. Select the body-parts you wish to use when calculating the ROI entry and time data.

At the bottom of this window, you'll see an entry box named *Bp probability threshold* (the first arrow in the image below) and a tick box option named `calculate distance moved within ROI` (the second arrow in the image below).

images/ROI_0320_1.png

`Bp probability threshold`: If you can, we **strongly** recommend to pre-process all videos, and remove any segments of the videos where animals are not present in the video as documented [HERE] prior to perfroming pose-estimation and importing videos into SimBA. However, if this is not possible, we can filter out out frames where probability for the pose-estimation accuracy is low, and not use these frames when we calculate the location of the animal in relation to the ROI (i.e., useful if you have an animal leaving the frame and the pose-estimation predictions are jumping all over the place). To filter out these frames, increase the probability threshold (DEFAULT = 0.0. I.e., all frames will be treated as the animal is present).

`calculate distance moved within ROI`: This tick-box option allows you to calculate the distance (in millimeters) that each animal has moved within each of your ROIs. If you run the analysis with this box ticked, SimBA will generate a time-stamped CSV file inside your project log folder named something akin to this: `Movement_within_ROIs_20210320143608.csv`. This file contains one row per analyzed video, and one column for each ROI and analyzed body-part. If you open this CSV up after analysis, you can expect it to look something like this (click here to enlarge):



This CSV file tells me - for example - that in `Video10`, the animal named `Animal_1` moved 1104 millimeters within the ROI shape named `rec_1` (looking in cell B2).

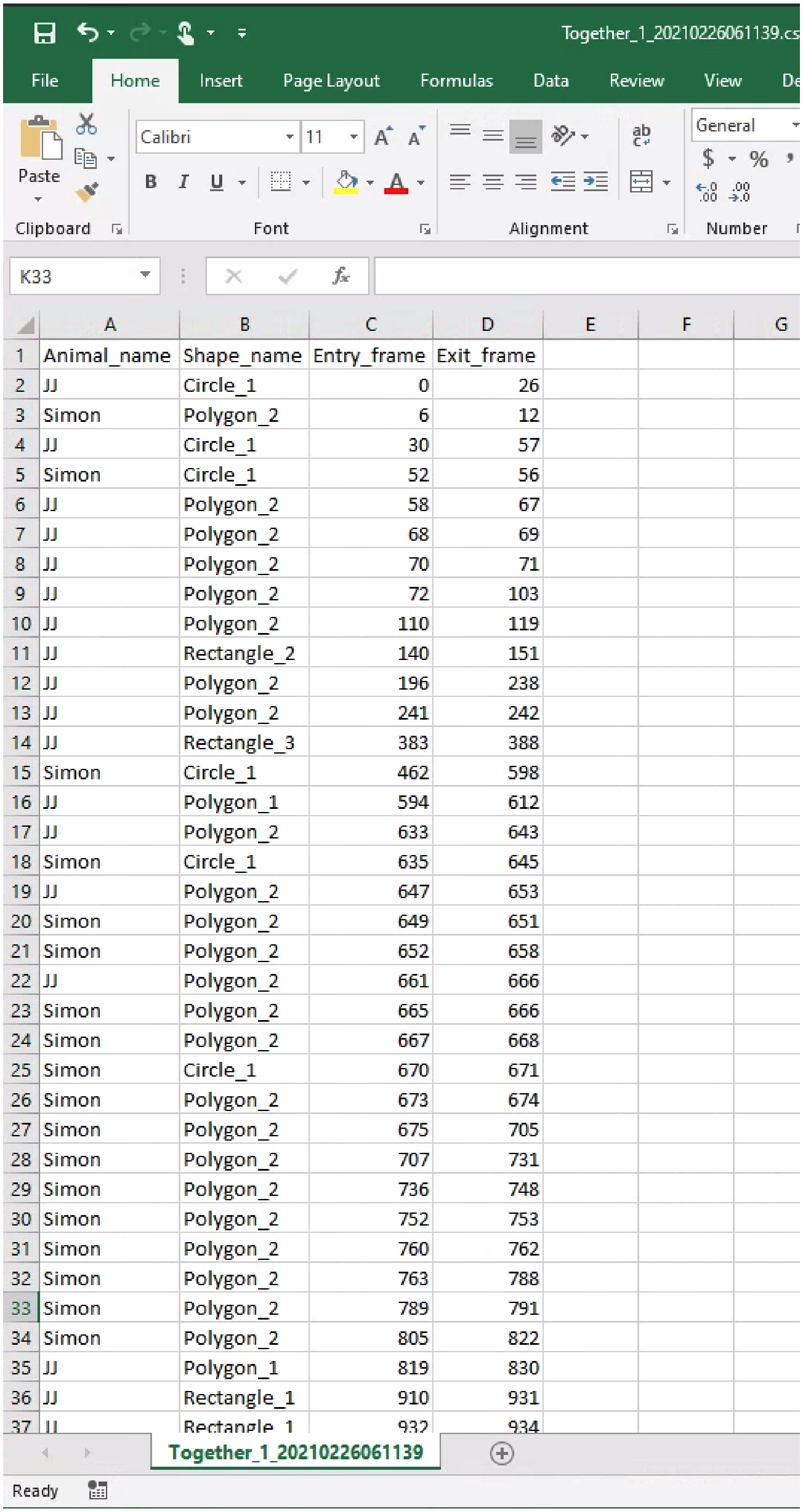
> **Note:** A short note on how simba calculates movements within ROIs: Pose-estimation tracking data can contain significant jitter. That means that even if the animal is standing still, the tracked body-part location can "jump around" a few pixels between sequential frames. Thus, if we calculate the movements of the body-parts between each sequential frame and add it up, it appears that the animal is moving a lot more than it actually is. To solve this, we bin the data into seconds, and calculate the movement of the body-part at the beginning of the second till the end of that second, and sum those values to get the movement distance of the animal.

Click `Run` to perform the ROI analysis.

Once complete, a statement will be printed in the main SimBA terminal window noting that the process is complete. The ROI descriptive statistics can be found in the `Project_folder/logs` directory in two time-stamped files.

- One file will be named `ROI_entry_data` and contain the number of entries into the different ROIs. It also contains the percent of entries into each ROI over all of the ROI entries. For an expected output ROI entry CSV file, click [HERE]
- A second file, named `ROI_time_data`, contain the time spent in in seconds the different ROIs. It also contains the percent of the session time spent in each of the ROIs. For an expected output ROI time CSV file, click [HERE]

Note: As of Feb-26 2021, the SimBA ROI analysis also generates a sub-folder within the *Project_folder/logs* directory named **`Detailed_ROI_analysis`**. This folder contains additional information, with exact time-stamps of when animals enter and exits user-drawn ROIs. For each analysed video, SimBA will generate one date-time stamped CSV file located in the `project_folder/logs/Detailed_ROI_analysis` directory. These CSV files files will contain 4 columns: `Animal_name`, `Shape_name`, `Entry_frame`, and `Exit_frame`, and look something like this:



	A	B	C	D	E	F	G
1	Animal_name	Shape_name	Entry_frame	Exit_frame			
2	JJ	Circle_1	0	26			
3	Simon	Polygon_2	6	12			
4	JJ	Circle_1	30	57			
5	Simon	Circle_1	52	56			
6	JJ	Polygon_2	58	67			
7	JJ	Polygon_2	68	69			
8	JJ	Polygon_2	70	71			
9	JJ	Polygon_2	72	103			
10	JJ	Polygon_2	110	119			
11	JJ	Rectangle_2	140	151			
12	JJ	Polygon_2	196	238			
13	JJ	Polygon_2	241	242			
14	JJ	Rectangle_3	383	388			
15	Simon	Circle_1	462	598			
16	JJ	Polygon_1	594	612			
17	JJ	Polygon_2	633	643			
18	Simon	Circle_1	635	645			
19	JJ	Polygon_2	647	653			
20	Simon	Polygon_2	649	651			
21	Simon	Polygon_2	652	658			
22	JJ	Polygon_2	661	666			
23	Simon	Polygon_2	665	666			
24	Simon	Polygon_2	667	668			
25	Simon	Circle_1	670	671			
26	Simon	Polygon_2	673	674			
27	Simon	Polygon_2	675	705			
28	Simon	Polygon_2	707	731			
29	Simon	Polygon_2	736	748			
30	Simon	Polygon_2	752	753			
31	Simon	Polygon_2	760	762			
32	Simon	Polygon_2	763	788			
33	Simon	Polygon_2	789	791			
34	Simon	Polygon_2	805	822			
35	JJ	Polygon_1	819	830			
36	JJ	Rectangle_1	910	931			
37	JJ	Rectangle_1	932	934			

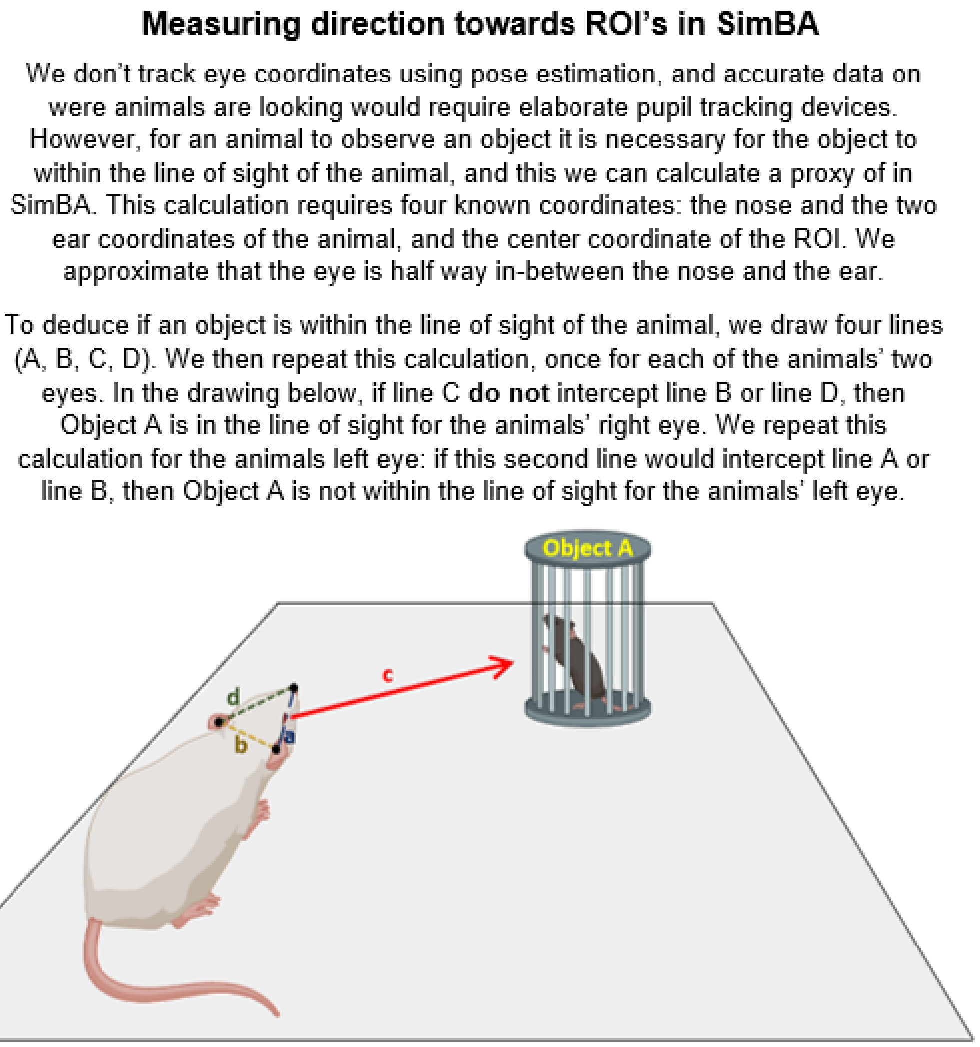
Consult these files if you want to know the exact frame numbers when animals entered and exited your drawn ROIs. If you want to convert the frame numbers to time-stamps, then divide these numbers with the fps of the video.

3.7.4 Part 3. Generating features from ROI data

1. With the ROI information, we can generate several further features that might be useful for predicting behaviors, or be handy within other third-party applications. For each frame of the video, the following features can be added to a previously calculated battery of features:

   - Boolean (TRUE or FALSE) value noting if the user-selected bodypart is located within or outside each of the ROIs
   - The millimeter distance between the animal body-part and the center of each ROI
   - The cumulative time spent in each of the ROIs
   - The cumulative percentage of the total session time spent in each of the ROIs

If the users are using the the recommended body parts (including nose, left ear, and right ear), SimBA will also calculate if the animal is directing towards each of the ROIs:



Measuring direction towards ROI's in SimBA

We don't track eye coordinates using pose estimation, and accurate data on were animals are looking would require elaborate pupil tracking devices. However, for an animal to observe an object it is necessary for the object to within the line of sight of the animal, and this we can calculate a proxy of in SimBA. This calculation requires four known coordinates: the nose and the two ear coordinates of the animal, and the center coordinate of the ROI. We approximate that the eye is half way in-between the nose and the ear.

To deduce if an object is within the line of sight of the animal, we draw four lines (A, B, C, D). We then repeat this calculation, once for each of the animals' two eyes. In the drawing below, if line C **do not** intercept line B or line D, then Object A is in the line of sight for the animals' right eye. We repeat this calculation for the animals left eye: if this second line would intercept line A or line B, then Object A is not within the line of sight for the animals' left eye.

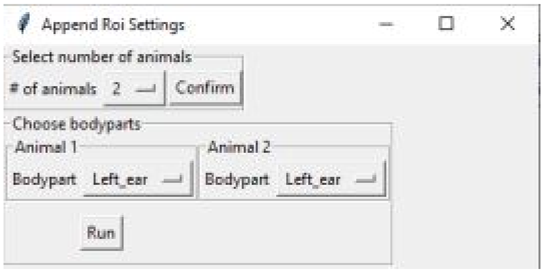
Using the directionality measure described in the image above, we can calculate further potentially informative features from each frame for subsequently generated machine learning models:

- Boolean (TRUE or FALSE) value noting if the animal is directing towards the center each of the ROIs or not.
- The cumulative time spent directing towards each of the ROIs.
- The cumulative percentage of the total session time spent directing towards each of the ROIs.

2. To generate these features and add them to the battery of features already calculated, begin by navigating to the *Extract features* tab.

Note: To calculate ROI based features it is first necessary to extract the non-ROI based features. If you haven't done so already, go ahead and first click on `Extract features` under the **Extract features** tab. A message will be printed in the main SimBA terminal window when the process is complete.

3. Next, click on `Append ROI data to features` under the **Extract features** tab. The following menu will pop open:



This menu is very similar to the Analyzing ROI data submenu described in [Part 2 - Step 2] above. First, begin by selecting the number of animals (or body-parts) you wish to use to calculate the ROI features for, and then click `Confirm`. A second sub-menu will appear below, named `Choose bodyparts`. This menu will contain as many dropdown menus as the number of animals (or body-parts) selected in the `Select number of animals` menu. Each of these drop-down menus will contain the body-parts used to track the animals. The body-part(s) choosen in this menu will determine which body-parts is used to calculate ROI-based features. For example, if you select the `Nose` body-part, the cumulative time spent in each of the ROIs, as well as all other features, will be calculated based on the `Nose` body-part. Once you have chosen your body-parts, then click on `Run`.

3. Once complete, a statement will be printed in the main SimBA terminal window noting that the process is complete. New CSV files will be generated for each of the videos in the project, located in the `project_folder/csv/features_extracted` folder. If you open these files using Microsoft Excel or OpenOffice Calc (watch out, they might be fairly large and therefore difficult to navigate manually) you will see several new columns at the end of the CSV: these columns contain the ROI-based features.

To proceed and generate machine learning classifiers that use these features, continue reading the Label behavior step, followed by the Train machine models parts of the SimBA tutorials.

3.7.5 Part 4. Visualizing ROI data

You can now generate visualizations of the ROI data for each of the videos in your project. To generate the visualizations, click on `Visualize ROI tracking` in the `Visualize ROI` submenu of the **ROI** tab. This will generate visualizations of the ROI and tracking data, like in these gifs:

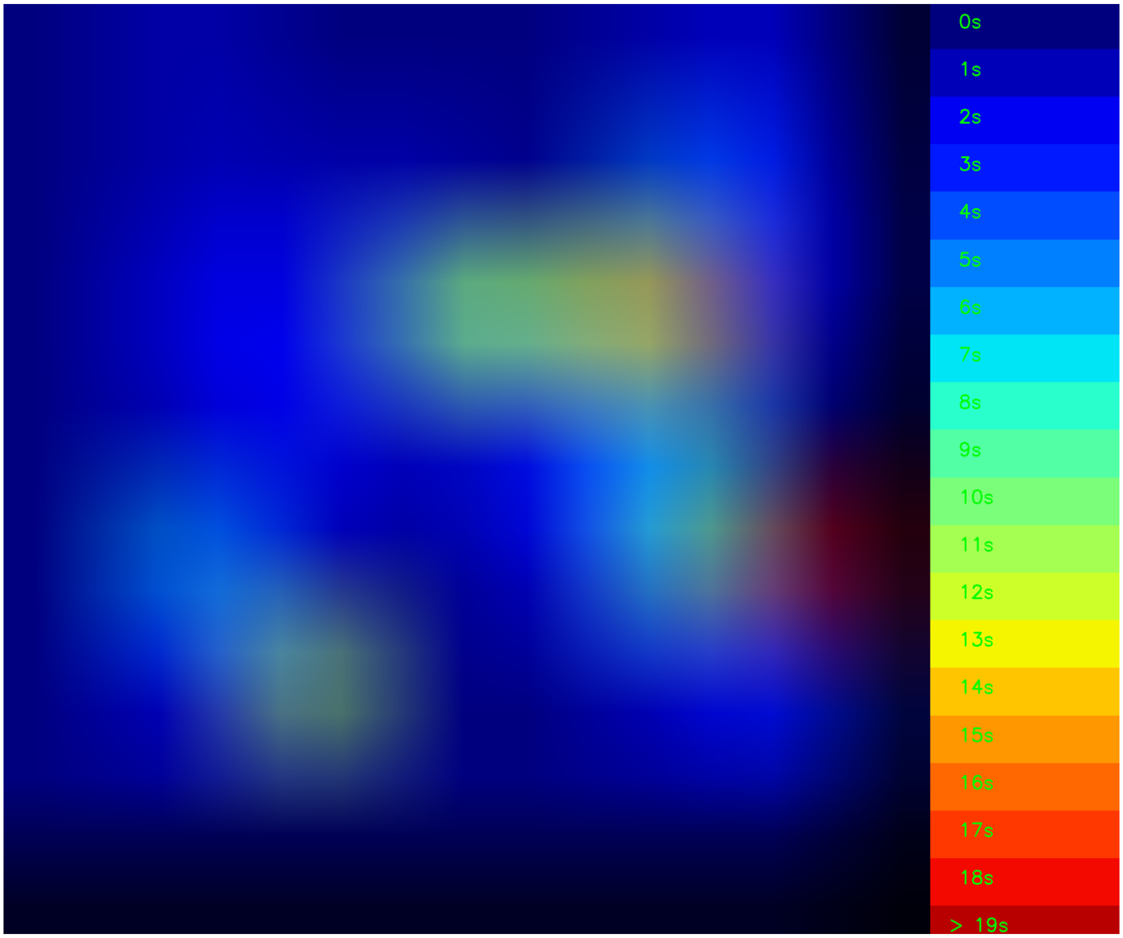
Clicking on `Visualize ROI tracking` will generate visualizations for all of the files located in the `project_folder/csv/features_extracted` folder. The visualization process will use the body-parts specified in `Analyze ROI data` step described in [Part 2 above].

The visualization process will generate **one** output video for every input video. For example, if you are generating visualizations for Video1, then the output mp4 video will be located in the following path: `project_folder/frames/output/ROI_analysis/Video1.mp4`.

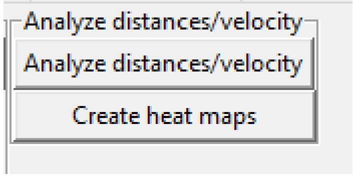
3.7.6 Part 5. Miscellaneous ROI tools

Heatmaps

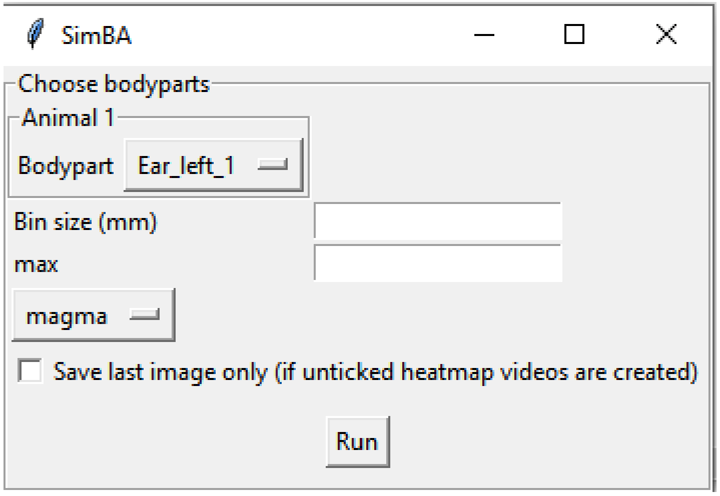
Users can generate heatmaps - in video or image formats - that represents the time spend in different regions of the arena. Note, this tool does **not** depend on *user-drawn* ROIs: instead this tool is for visualizing time spent in all regions of the arena according to user defined region sizes and user-defined color-scales. As an example, this tool is used for generating images, or videos, resembling this image below. For an example of the heatmap tool output in video format, check the [SimBA YouTube playlist].



These heatmap images, or videos, are colour-coded according to the time spent in different parts of the arena. To generate such heatmaps, begin by clicking on the `Create heatmaps` button in the ROI tab:



When clicking on `Create heatmaps`, the following menu pops open which accepts several required user-defined parameters:



Bodypart Ear_left_1

Bin size (mm)

max

magma

☐ Save last image only (if unticked heatmap videos are created)

1. Bodypart: SimBA can, currently, only generate heatmaps representing the time spent in different regions of the image based on a single body-part. Use the drop-down menu to select the body-part you wish to use to generate the heatmaps.

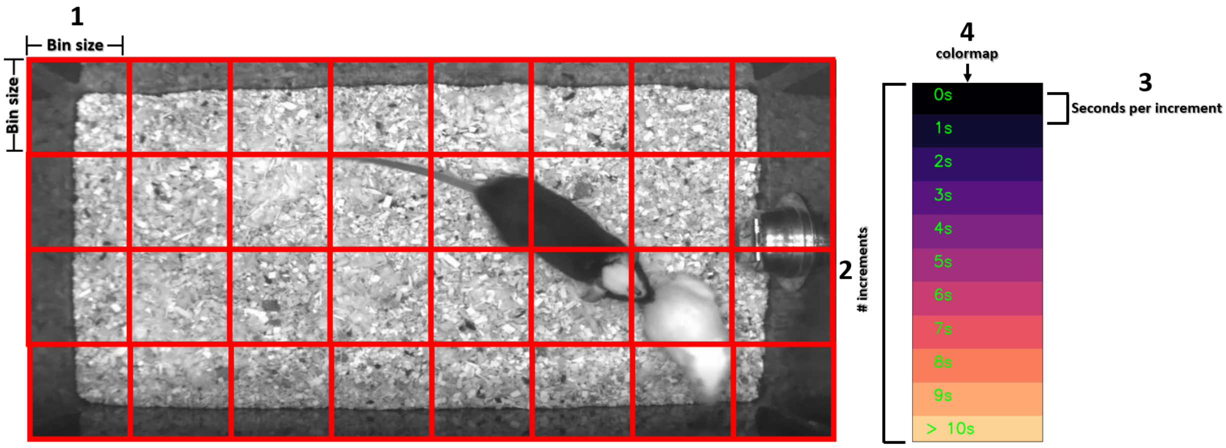
2. Bin size (mm): To generate heatmaps, SimBA needs to divide the image into different square regions. Use this entry-box to define the size of each square region. To get a better sense of what a `Bin size (mm)` is, and how SimBA goes about generating the heatmaps, see the image below (1). For example, enter *50* in this entry box to get regions that are 5x5cm in size.

3. # max ('auto' or integer): The created heatmaps have adjoining colorbars. In this entry box, define the number of seconds that should represent the strongest color and max value in your heatmap. Users can also insert the string `auto` in this entry-box. If `auto`, the SimBa will calculate the max value in the video (max time spent in a single zone) and use this value as max (I'm not entirely sure how these heatmaps are created in commercial tools but I was inspired by the [MATLAB Pathfinder tool].

4. Palette. Use the drop-down to specify the palette of the heatmaps. For reference of how the palette looks like, see the the [SimBA visualization tutorials - Step 9]

7. Save last image only. Users may want to (i) generate videos, where each frame of the video represents the cumulative time spent in each region up until the current point in time, or (ii) generate single images that represents the cumulative time spent in each region across the entire videos. Tick the `Save last image only` box to generate a single PNG images for each video in the project. Untick the `Save last image only` box to generate one heatmap video for each video in the project.

Once this information is entered, click `Run`. Generated heatmap videos/frames are saved in the `project_folder/frames/output/heatmap_location` directory.



'Directionality' between animals

To analyze and visualize how animals are directing towards each other, rather than user-defined ROIs, head to THIS TUTORIAL

3.8 'Pseudo-labelling' behaviors in SimBA

Annotating frames can be the most time-consuming aspect of supervised machine learning, and computational tools are developed at pace to address this time-sink. For example, pose-estimation tools such as DeepLabCut and SLEAP, promote workflows that significantly decrease the required body-part labelling time; DeepLabCut asks users to extract/refine/merge body-part outlier, and SLEAP asks users to take a computer-assisted labelling approach and run non-optimized models and correct their prediction results until they are perfected. JAABA (Janelia Automatic Animal Behavior Annotator), another excellent and popular tool for machine classification of animal behavior, is also based on a workflow composed of iterative re-training of classifiers based on machine-generated predictions.

These workflows are built upon feeding computer-generated predictions, initially with human oversight, back into the model as training examples and thus improving its predictions through more data, all while spending significantly less time annotating images. This is approach is **not** restricted to animal tracking / pose-estimation, but can also use for improving the performance of behavior classifiers in SimBA.

Note: If you want to label behaviors, including the use of pseudo-labelling methods in SimBA, you **must** extract the frames for the videos you wish to label. The frame extraction ensures that the frame number, the behavioral label, and the displayed frame during the annotation-work are all align. For more information on how to extract frames in SimBA, see the Scenario 1 Part 2 - Import more tracking data into project, or the generic Tools menu tutorials.

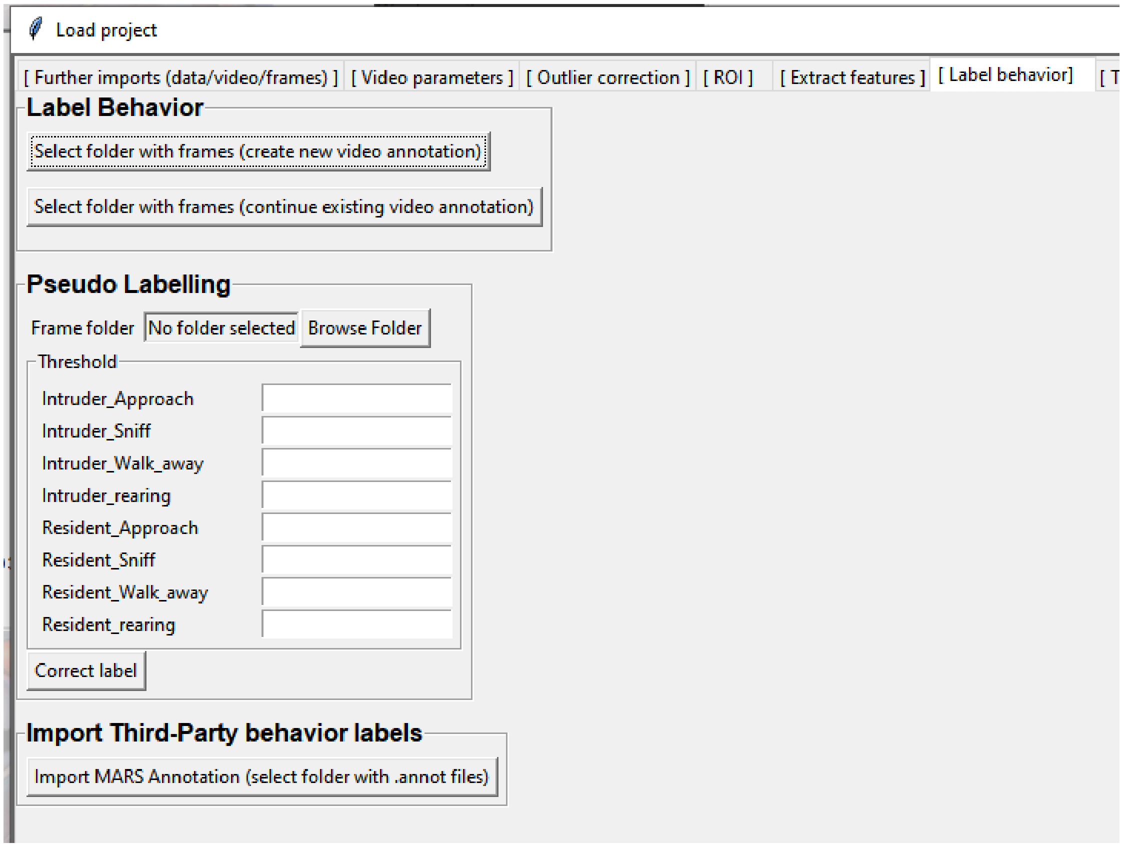
3.8.1 Step 1: Generating initial machine predictions in SimBA

Before using the ‘pseudo-labelling’ tool in SimBA, we need to generate some machine learning classification predictions for videos that has not been used to train the model. Detailed information on how to generate such machine classifications can be found in the Scenario 1 - building classifiers from scratch tutorial. More documentation on how to generate machine predictions can also be found in the generic SimBA documentation. We recommend reading the of these tutorials up to and including Step 8 - Run Machine Model.

When you’ve successfully generated your machine predictions, you will have CSV files containing your machine classifications, with one CSV file for each video in your project, in your `project_folder/csv/machine_results` directory. We will now use the `Pseudo-labelling` tool in SimBA to look at these machine classifications and correct them when necessary.

3.8.2 Step 2: Using Pseudo-labelling in SimBA

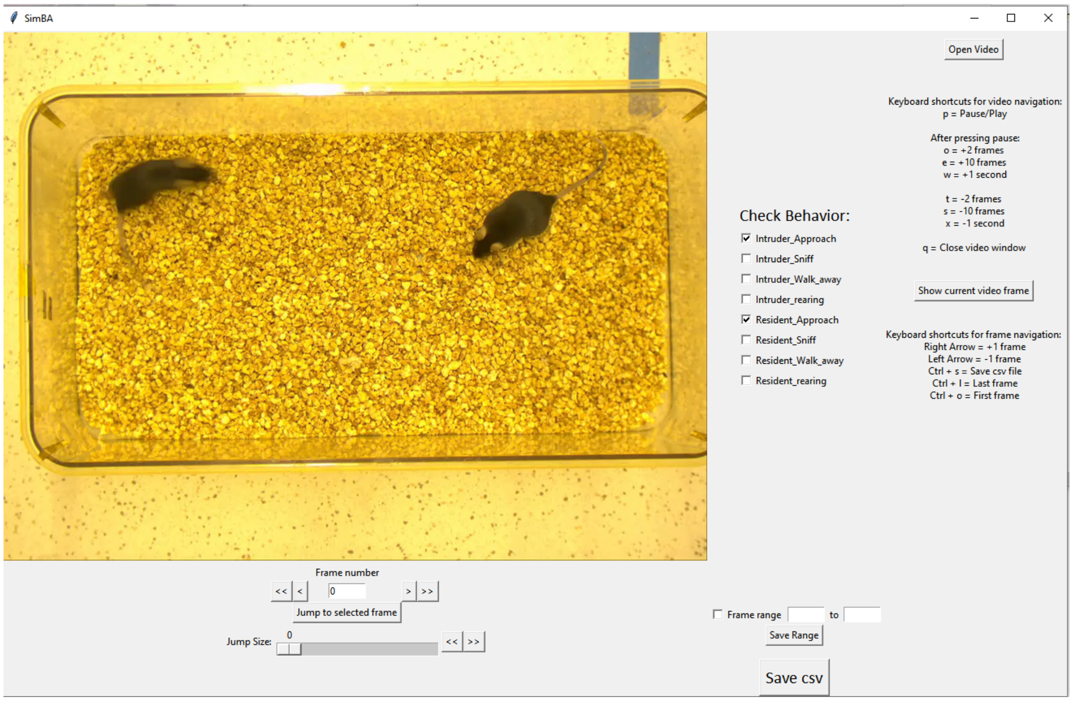
After loading your project in SimBA, navigate to the `Label behavior` tab and uou should see the following menu:



1. In the sub-menu titled `Pseudo Labelling`, there is a entry-box called `Frame folder`. Click on `Browse` and select to the folder containing the frames for the video you wish to correct the machine predictions for.
2. Beneath the `Frame folder` entry-box, there is a sub-menu titled `Threshold`. Within this sub-menu there are entry-boxes, with one entry-box for each classifier in the project. in each of these entry-boxes, insert a value between `0.00` and `1.00`. These numbers represents the classification threshold for each classifier. If you are unfamiliar with `classification thresholds`, then you can read more about them in the tutorial for Scenario 1 - Step 8, or Scenario 2 - Part3. In brief, the `classification threshold` represents how sure the computer has to be before it classifies a frame as containing the behavior. A `classification threshold` value of 0.80, for example, would mean that the computer has to be 80% sure that the frame contains the behavior to classify

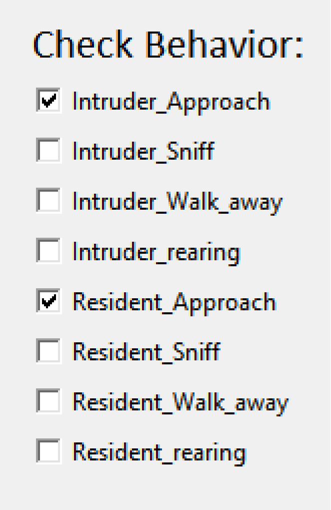
it as containing the behaviour. When the `classification threshold(s)` have all been entered, click the `Correct label` button.

3. After clicking the `Correct label` button, a window will pop open that looks similar to this:



This interface is near-identical to the labelling interface described in the SimBA behavioral annotator GUI tutorial and users should head over to this tutoral for detailed instructions on how to use this interface to label behaviors.

This interface, however, contains one key difference: the `Check Behavior` menu has been pre-filled with the models classification results according to the threshold the user set in `Threshold` menu described in above:



Check Behavior:

☑ Intruder_Approach

☐ Intruder_Sniff

☐ Intruder_Walk_away

☐ Intruder_rearing

☑ Resident_Approach

☐ Resident_Sniff

☐ Resident_Walk_away

☐ Resident_rearing

Now navigate the frames and the machine model predictions, by using the arrow keys, and view the video by clicking the `Open video` button. Proceed to add any ticks in tick-boxes that may be missing ticks in the `Check behavior` menu, or remove ticks that are incorrectly present. Again, please see the SimBA behavioral annotator GUI tutorial for more information on how to use this annotator interface. Once you have finished correcting the machine-generated predictions, click on `Save csv`. A new CSV, named after the video, with the user-corrected labels, is then saved in the project_folder/csv/targets_inserted folder and can be used, together with other annotated CSV files, to generate new machine classification models for the behaviors of interest.

Note: If the representations of the machine predictions (e.g., the tick-boxes) are too sensitive (e.g., too many boxes are ticked when they shouldn't be ticked) or too weak (e.g., too many boxes are un-ticked when they should be ticked), consider closing the labelling interface window, and revise the thresholds in the Pseudo Labelling - thresholds sub-menu. If too many boxes are ticked for a specific behavior classifier, then decrease the threshold for this classifier. If too few boxes are ticked, then increase the threshold for the specific classifier. When you have titrated the thresholds, re-start the labelling interface by clicking the Correct labels button.

If you want to correct further videos with machine-generated predictions, then close the labelling interface, and navigate back to the sub-menu titled Pseudo Labelling and the Frame folder entry-box and browse and select a new folder containing frames from a different video.

Note: What happens during this process, is that SimBA grabs the CSV file associated with the name of the frame folder from the project_folder/csv/machine_results directory when the user clicks the Correct labels button. Moreover, when the user clicks the Save csv button, a new CSV file is generated in the project_folder/csv/targets_inserted folder, and this new file is identical to the file in the project_folder/csv/machine_results folder, except that the relevant prediction columns have been amended to reflect the users-introduced changes indicated through the Check behavior tick-box menu.

Congrats! You have now successfully generated more machine model training data using Pseudo-labelling in SimBA.

For details tutorial on how to generate machine classifiers based on the pseudo-labelled data, and all other CSV data files in your project_folder/csv/targets_inserted directory, see the Train Machine Model tutorial.

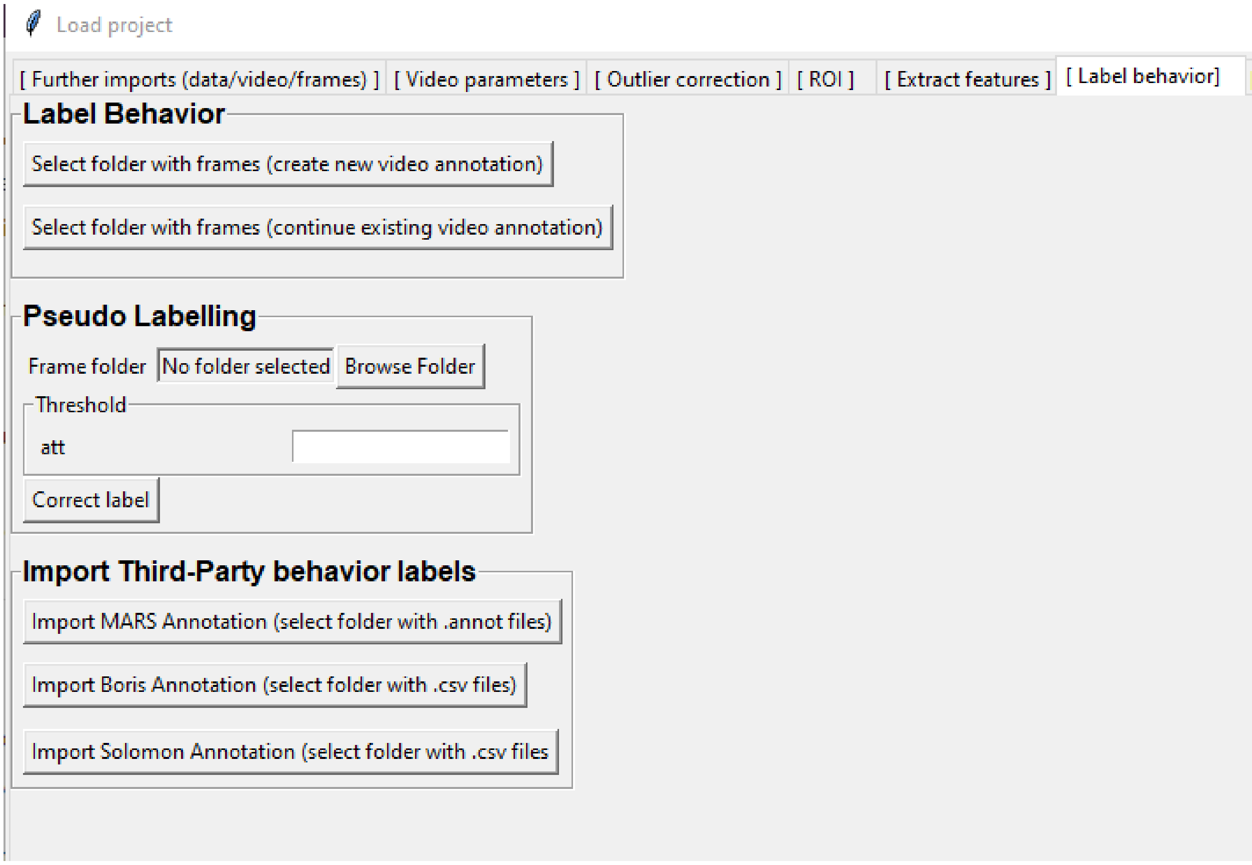
3.9 Appending third-party annotations in SimBA

SimBA has an in-built behavior interface that allow users to append experimenter-made labels to the features extracted from the pose-estimation data. Accurate human labelling of images can be the most time-consuming part of creating reliable supervised machine learning models. Sometimes the experimenter already have accurate labels for the videos a set of videos, but the labels have been generated in a third-party annotation tool such as BORIS, Jwatcher, Solomon coder, or MARS/BENTO. SimBA allows the user to append labels stored in these formats, saving you from having to repeat the annotation process.

For more detailed information on how to export BORIS annotations in a format compatible with SimBA - check out THIS SHORT TUTORIAL.

3.9.1 How to import annotations into SimBA project

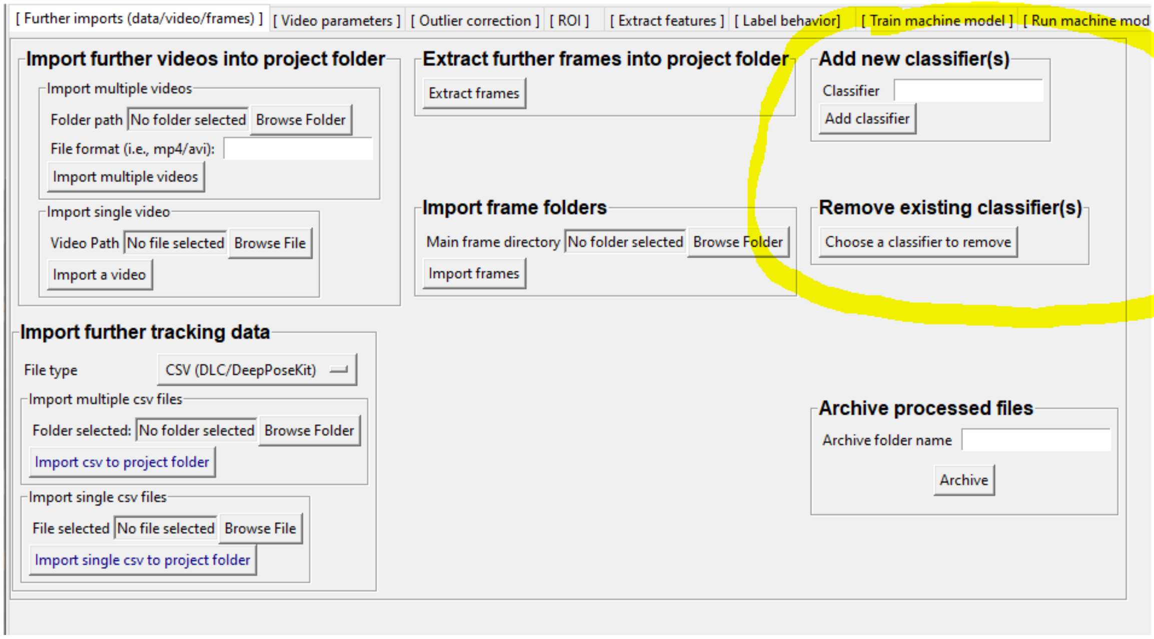
1. Once you have created a project, click on `File` –> `Load project` to load your project. For more information on creating a project, click HERE.
2. Make sure you have extracted the features from your tracking data and there are files, one file representing each video in your project, located inside the `project_folder/csv/features_extracted` directory of your project. For more information on extracting features, click HERE.
3. After loading your project, click on the `Label behavior` tab, and you should see the following window:


Load project
[Further imports (data/video/frames)]
[Video parameters]
[Outlier correction]
[ROI]
[Extract features]
[Label behavior]
Label Behavior
Select folder with frames (create new video annotation)
Select folder with frames (continue existing video annotation)
Pseudo Labelling
Frame folder
No folder selected
Browse Folder
Threshold
att
Correct label
Import Third-Party behavior labels
Import MARS Annotation (select folder with .annot files)
Import Boris Annotation (select folder with .csv files)
Import Solomon Annotation (select folder with .csv files

4. In the **Import Third-Party behavior labels** sub-menu, click on the button that represents the third-party application that the annotations were generated with. A menu will appear that asks you yo select the folder that contains the annotation files in CSV file format. New files will then be generated containing your annotations and features and they will be stored in the `project_folder/csv/targets_inserted` directory of your project.

Tip: For SimBA to know which third-party annotation data should be appended to which video data, the two files need to have the same name. Thus, if you are processing two videos in SimbA named `Video_1` and `Video_2`, then the third-party annotation files should be named `Video_1` and `Video_2`.

Note: Keep in mind that the behaviors/classifiers has to be defined (with the same names as they appear in the third-party annotation files) in the SimBA project folder. You can add or remove classifier in `Further imports(data/video/frames)` –> `Add classifier` or `Remove existing classifier`



Warning: If the third-party annotation files contain annotations for behaviors that you have **not** defined in your SimBA project, then those annotations will be discarded and not appended.

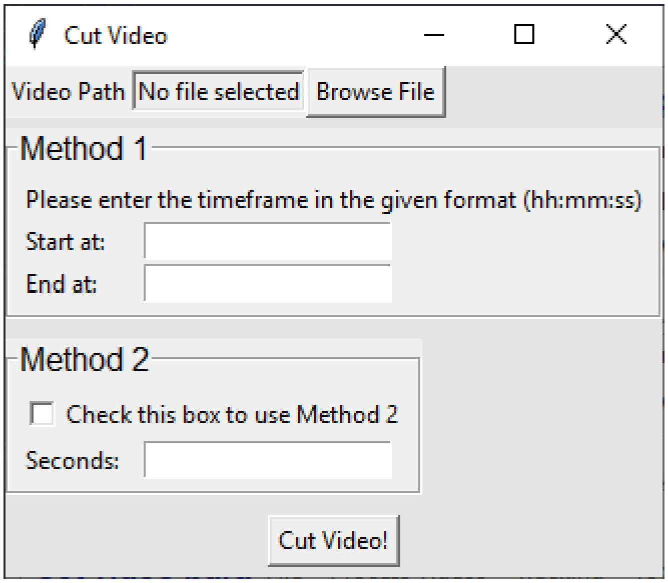
3.10 How to use tools to process videos in SimBA

We have developed video and image processing tools and incorporated them into the overall SimBA pipeline. However, many of the tools are useful on their own or in “one-off” situations. To make these easily accessible, the tools have been incorporated into their own space within the GUI, and are described below.

3.10.1 Shorten Videos

This is a tool used to trim video lengths. The tool contains two different methods:

Method 1 and **Method 2**.

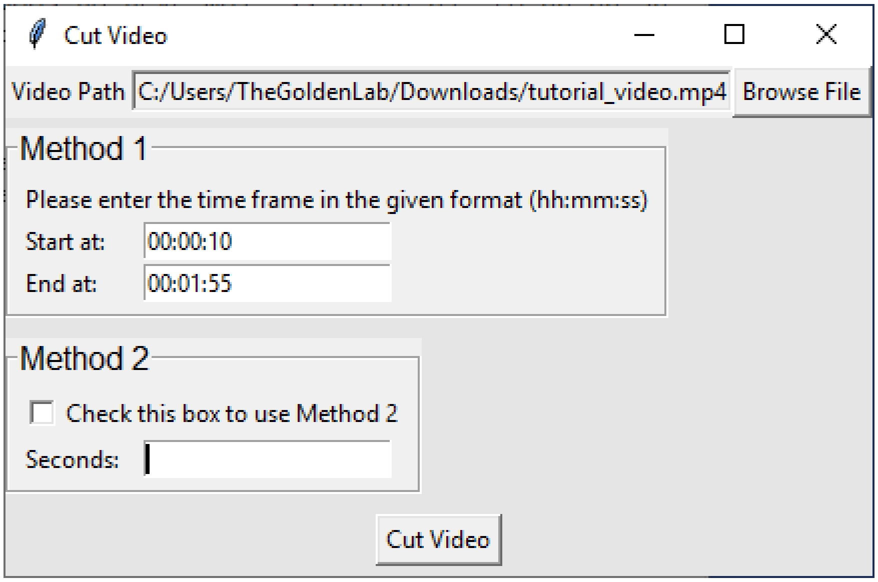


Method 1

Use Method 1 to trim both the beginning and the end of the video.

Let's say we have a 2 minute long video and we want to get rid of the first 10 seconds, and the last 5 seconds. We would start our video on at `00:00:10`, and end the video at `00:01:55`.

1. First, click on `Browse File` to select the video to trim.
2. Enter the start time in the `Start at:` entry box, in this case it will be *00:00:10*
3. Enter the end time in `End at:` entry box, in this case it will be *00:01:55*. The settings should look like the image below.

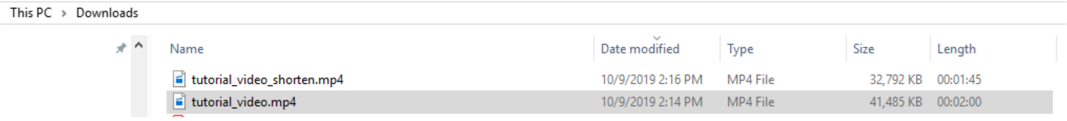

Cut Video
Browse File
Method 1
Please enter the time frame in the given format (hh:mm:ss)
Method 2
□ Check this box to use Method 2
Seconds:
Cut Video

Video Path C:/Users/TheGoldenLab/Downloads/tutorial_video.mp4

Start at: 00:00:10

End at: 00:01:55

4. Click `Cut Video` to trim the video and a new shorter video will be generated. The new shorter video will have a name of *Name of original video* + *_shorten* and will be located in the same folder as the original video.



This PC > Downloads

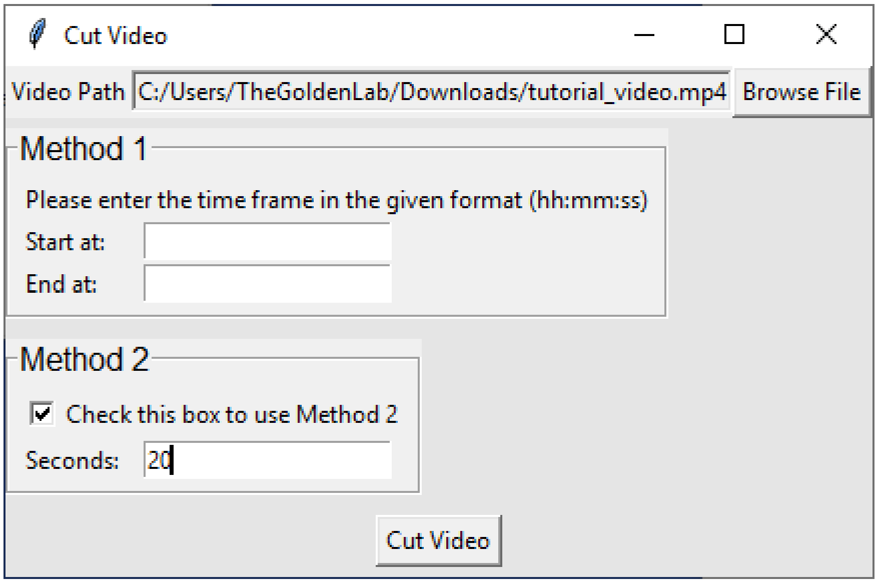
Name	Date modified	Type	Size	Length
tutorial_video_shorten.mp4	10/9/2019 2:16 PM	MP4 File	32,792 KB	00:01:45
tutorial_video.mp4	10/9/2019 2:14 PM	MP4 File	41,485 KB	00:02:00

Method 2

Method 2 cuts of the beginning of the video.

Let's say we have a 2 minute long video and we want to get rid of the first 20 seconds from the start of the video.

1. Enter the amount of time that needs to be trimmed from the start of the video in `Seconds:`, in this case it will be *20*



Cut Video

Video Path: C:/Users/TheGoldenLab/Downloads/tutorial_video.mp4

Method 1

Please enter the time frame in the given format (hh:mm:ss)

Start at:

End at:

Method 2

☑ Check this box to use Method 2

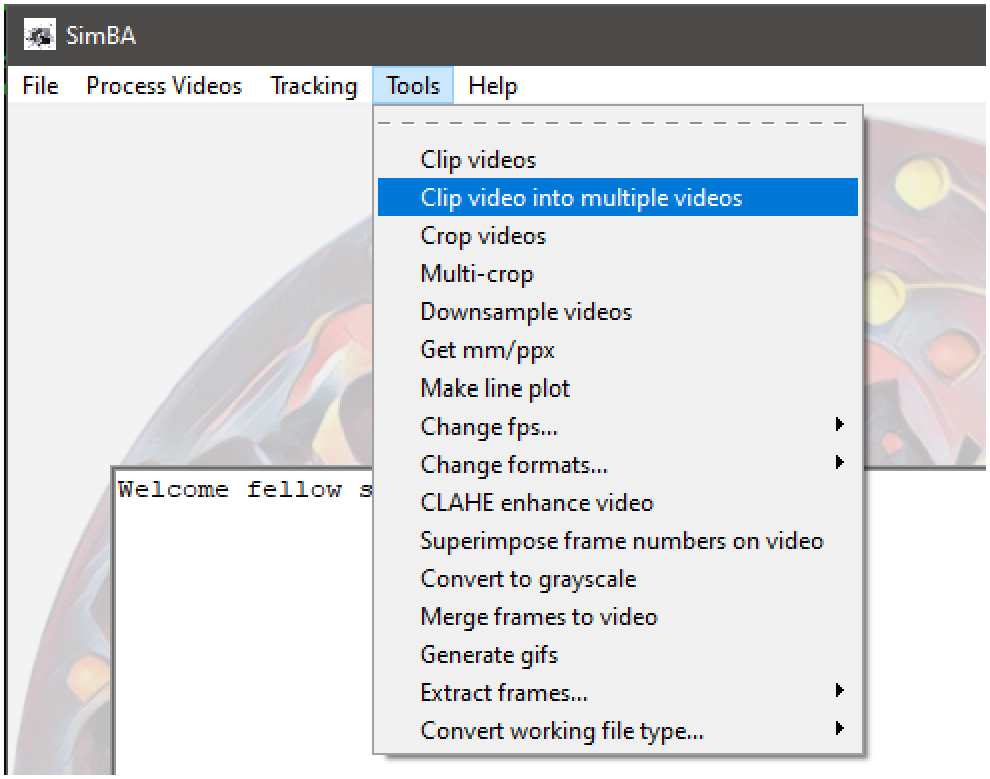
Seconds: 20

2. Click `Cut Video` to trim video and a new shorten video will be generated, the new video will have a name of *Name of original video* + *_shorten* and will be located in the same folder as the original video.

3.10.2 Clip video into multiple videos

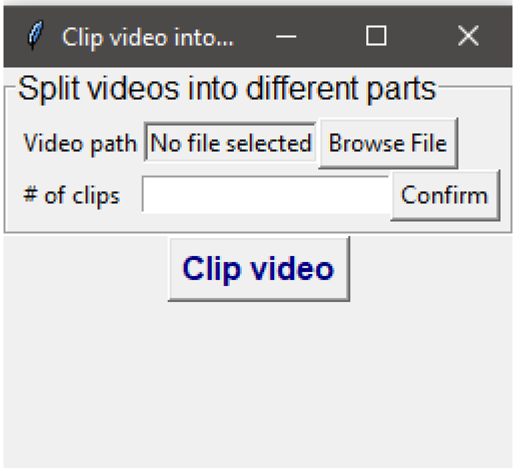
This tool can help users to cut the videos into multiple clips/section.

1. Click on the `Clip video into multiple videos` from the `Tools` section.



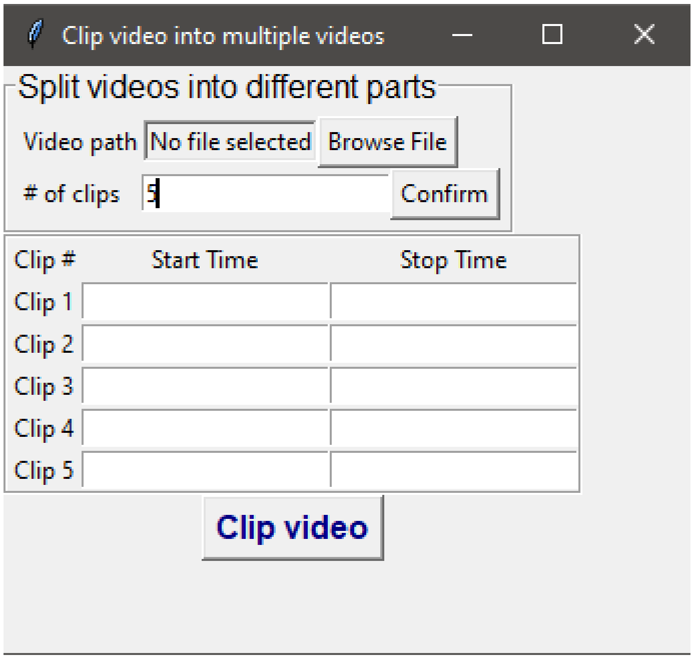
Welcome fellow s

2. Select the video that you want to split/clip by clicking `Browse File`



Video path No file selected

3. Put in the number of output clips/sections that you want in `# of clips`, and click `Confirm`. Please note that if you put in the wrong number the first time, you can re-enter the number and click `Confirm` to change the table.


Clip video into multiple videos
Split videos into different parts
Browse File
Confirm
Clip video

Video path: No file selected

# of clips: 5

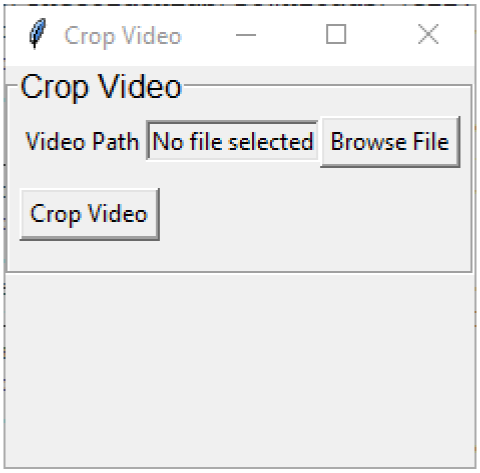
Clip #	Start Time	Stop Time
Clip 1		
Clip 2		
Clip 3		
Clip 4		
Clip 5		

4. Then enter the `Start Time` and `Stop Time` in the following format `HH:MM:SS`. For example, for a minute and 20 seconds it will be `00:01:20`.

5. Once the table has been filled, click `Clip video` and the video will be output on the same folder path/ directory of your original video.

3.10.3 Crop Video

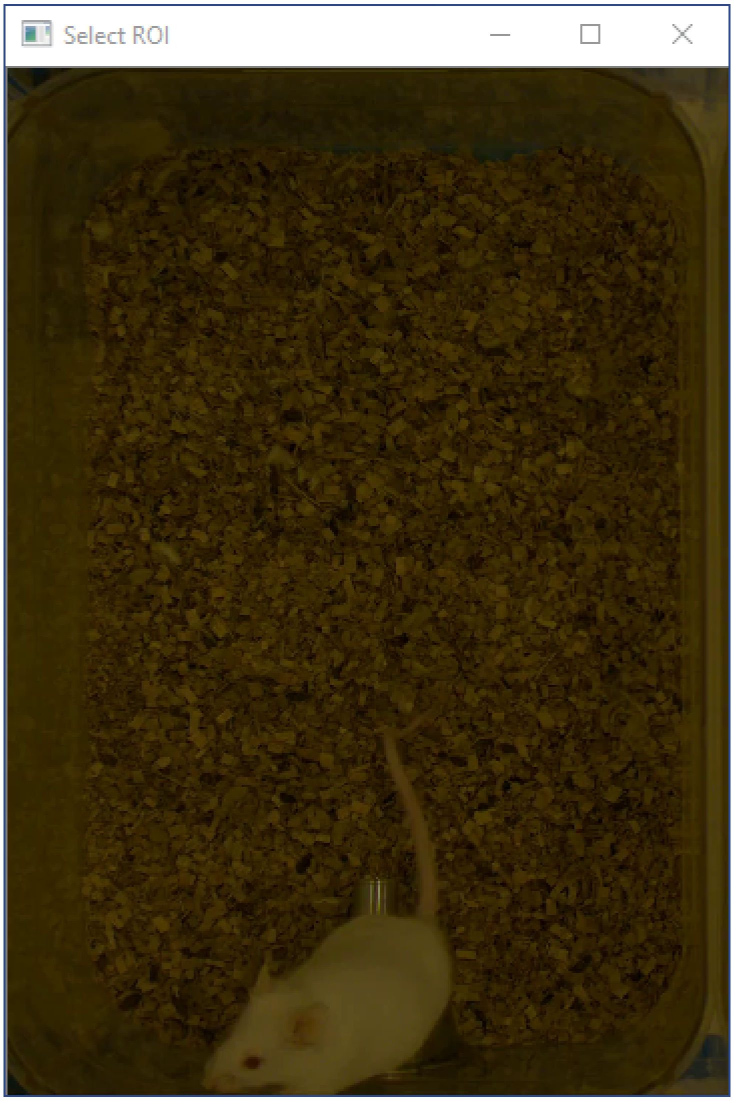
This is a tool to crop videos.



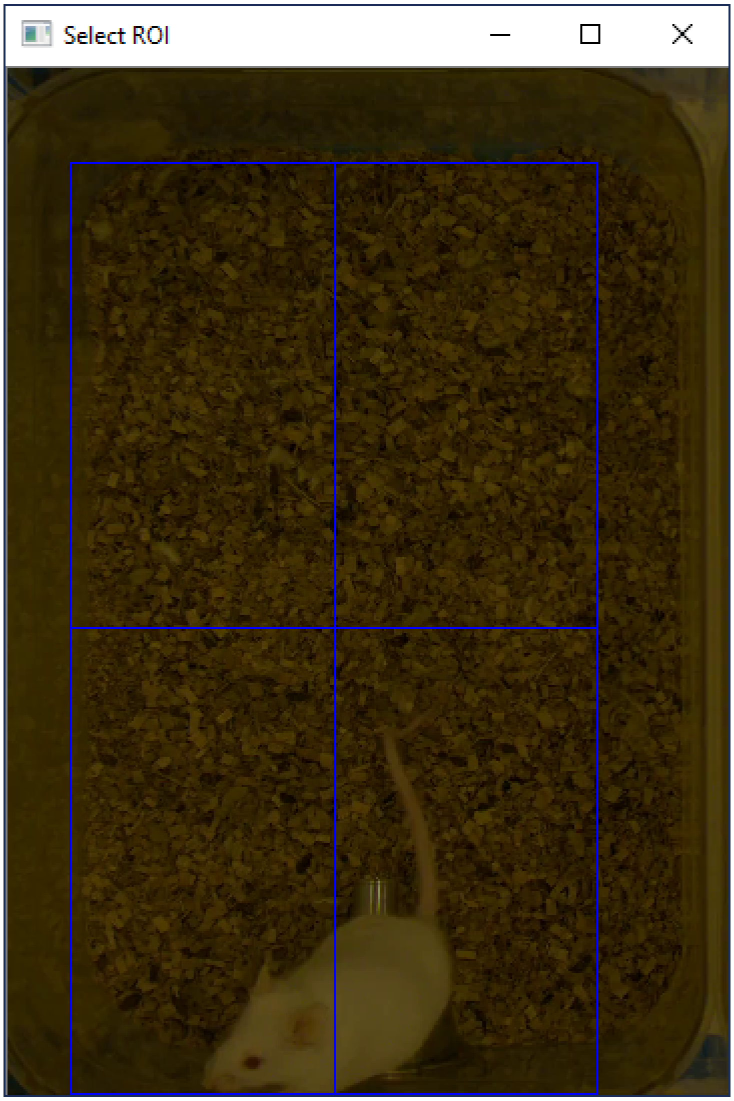
Video Path: No file selected

1. First, click on `Browse File` to select a video to crop.

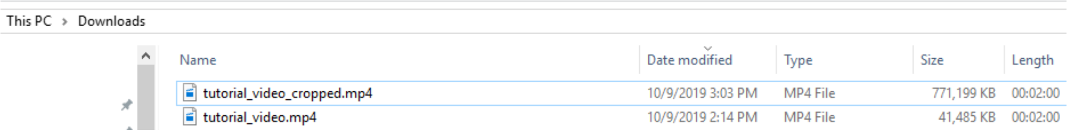
2. Click on `Crop Video` and the following window `Select ROI` will pop up.



3. Use your mouse to *Left Click* on the video and drag the rectangle bounding box to contain the area of the video you wish to keep. *You can always left click and drag on the video to recrop your video*



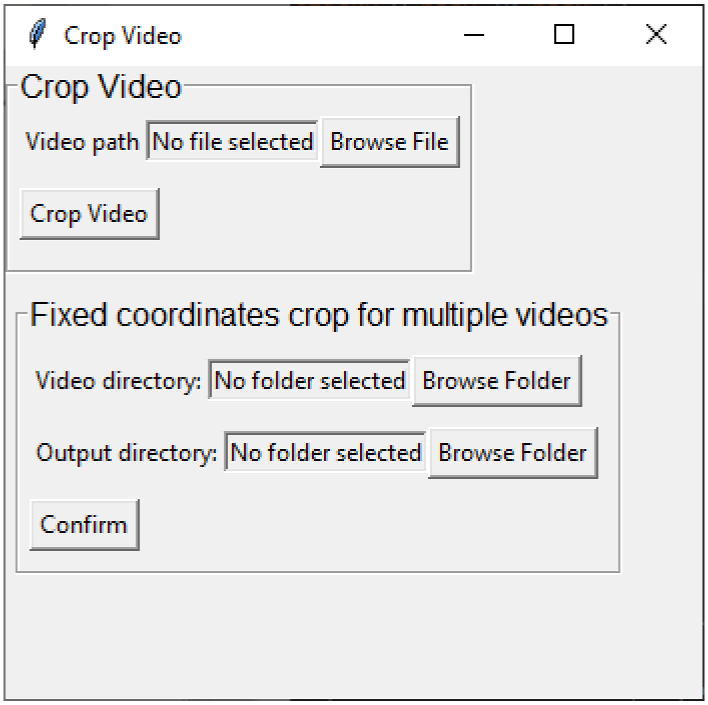
4. Hit `Enter` **twice** and SimBA will crop the video. The new video will have a name of *Name of original video* + *_cropped* and will be located in the same folder as the original video.


This PC › Downloads

Name	Date modified	Type	Size	Length
tutorial_video_cropped.mp4	10/9/2019 3:03 PM	MP4 File	771,199 KB	00:02:00
tutorial_video.mp4	10/9/2019 2:14 PM	MP4 File	41,485 KB	00:02:00

3.10.4 Fixed Crop Videos

This tool allows the user to crop once and apply the same dimension to the rest of the videos in the folder.

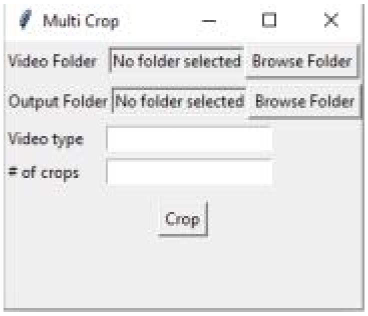


1. Under `Video directory`, select the input video folder by clicking `Browse Folder`.
2. Then, select an output folder.
3. Click on `Confirm` and an image will pop up, use your mouse to *Left Click* on the video and drag the rectangle bounding box to contain the area of the video you wish to keep. *You can always left click and drag on the video to recrop your video*
4. Hit `Enter` **twice** and SimBA will crop all the videos in the folder with the same coordinate. The new videos will have a name of *Name of original video* + *_cropped* and will be located in the output folder.

3.10.5 Multi-crop videos

This is a tool to used to multi-crop videos. For example, if you recorded four different environments with a single camera, you can use this tool to split single recordings into 4 different videos. This tool operates on all videos in a folder that is defined by the user. The user is required to draw the number of defined rectangles on **each** of the videos in the specified folder.

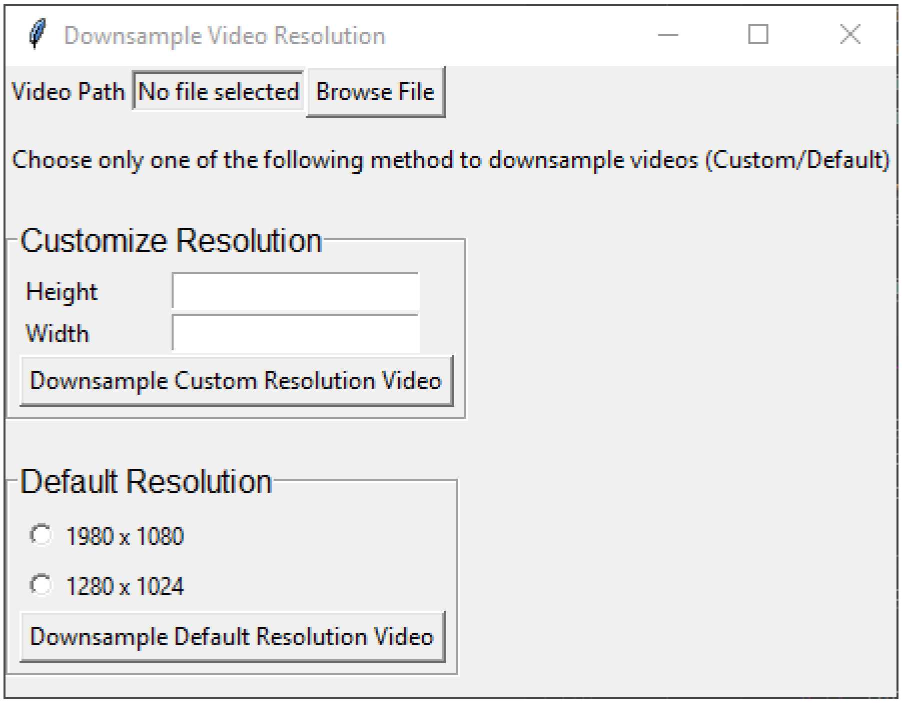
1. First, click on Multi-crop and the following menu will pop-open:



2. Next to *Video Folder*, click on `Browse Folder` and select a folder containing the videos you which to multi-crop.
3. Next to *Output folder*, click on `Browse Folder` and select a folder that should house the cropped output videos.
4. Next to *Video type*, type the file format of your input videos (e.g., mp4, avi etc).
5. Next to *# of crop*, type in the number of cropped videos you which to generate from each single input video (e.g., 4). Click on **Crop** to proceed. When you click on **Crop**, the first frame of the first video in the specified folder will be displayed, and the name of the video and rectangle number is printed overlaid:
6. Left click the mouse and drag from the top left corner to the bottom right corner of the first video you wish to generate. When finished with the first video, press `Enter`. Repeat this step for the next videos you wish to generate from Video 1. Once Video1 is complete, repeat these steps for all the videos in the user-specified *Video Folder*.
7. The cropped output videos will be located in the user-defined *Output folder* as defined in Step 3.

3.10.6 Downsample video

This is a tool to downsample a video into smaller size and reduce the resolution of the video.



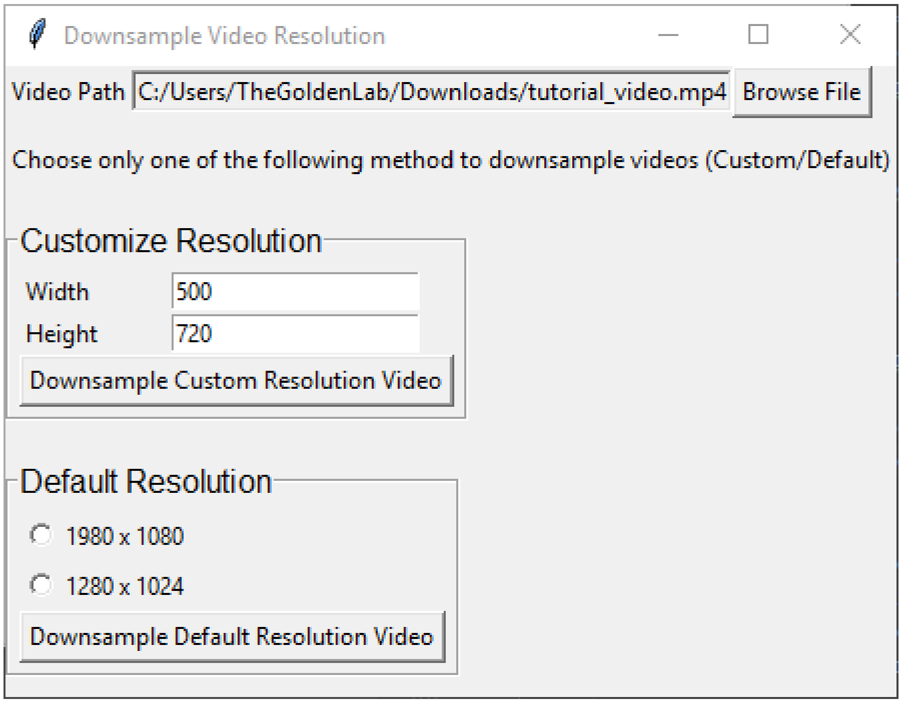
Choose only one of the following method to downsample videos (Custom/Default)

The downsample video tool has two options: **Customize Resolution** and **Default Resolution**.

Customize Resolution

Use this tool to downsample video into any height and width.

1. First, click on `Browse File` to select a video to downsample.
2. Then, enter any values in the `Height` and `Width` entry boxes.



Downsample Video Resolution
Browse File
Downsample Custom Resolution Video
Downsample Default Resolution Video

Video Path: C:/Users/TheGoldenLab/Downloads/tutorial_video.mp4

Choose only one of the following method to downsample videos (Custom/Default)

Customize Resolution

Width: 500

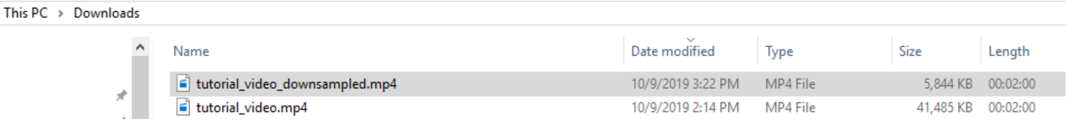
Height: 720

Default Resolution

○ 1980 x 1080

○ 1280 x 1024

3. Click on `Downsample to custom resolution` to downsample the video. The new video will have a name of *Name of original video* + *_downsampled* and will be located in the same folder as the original video.



This PC › Downloads

Name	Date modified	Type	Size	Length
tutorial_video_downsampled.mp4	10/9/2019 3:22 PM	MP4 File	5,844 KB	00:02:00
tutorial_video.mp4	10/9/2019 2:14 PM	MP4 File	41,485 KB	00:02:00

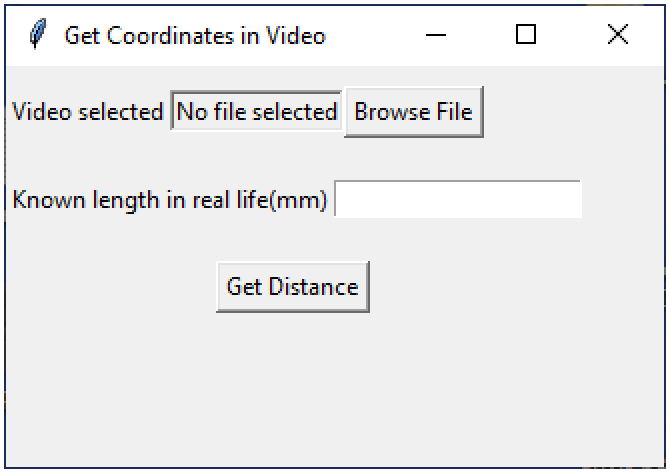
Default resolution

This tool allows the user to downsample a video quickly.

1. First, click on `Browse File` to select a video to downsample.
2. Tick on one of the resolution options.
3. Click on `Downsample to default resolution` and the video will downsample into the selected resolution. The video will be located in the same folder as the original video.

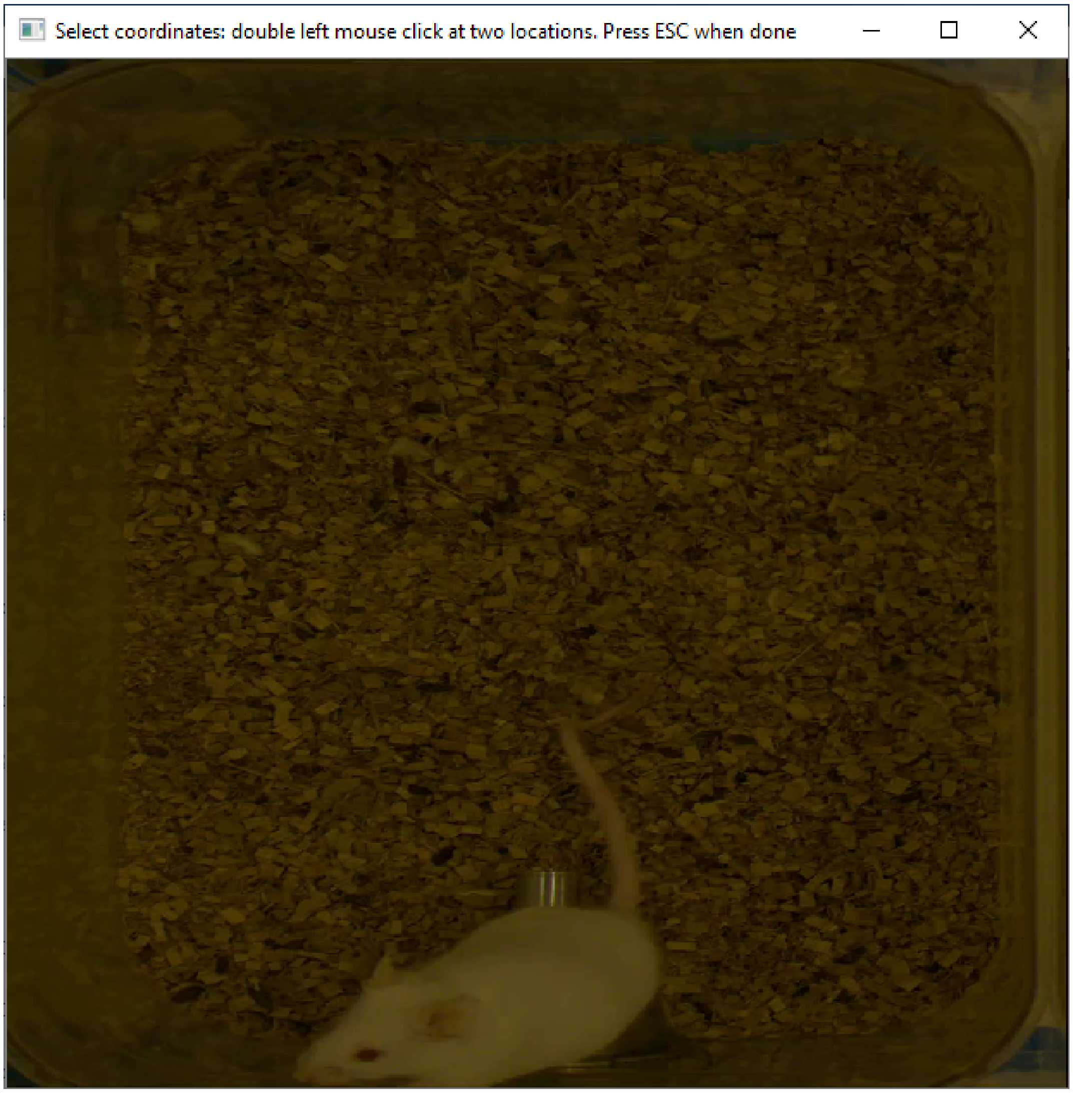
3.10.7 Get Coordinates (calibrate distance)

This tool is to get the length (millimeter) per pixel of a video.



Let’s say we want to find out the metric length per pixel of a video of a mouse in a cage, and we know the width of cage is 130 millimeters (it’s a tight one).

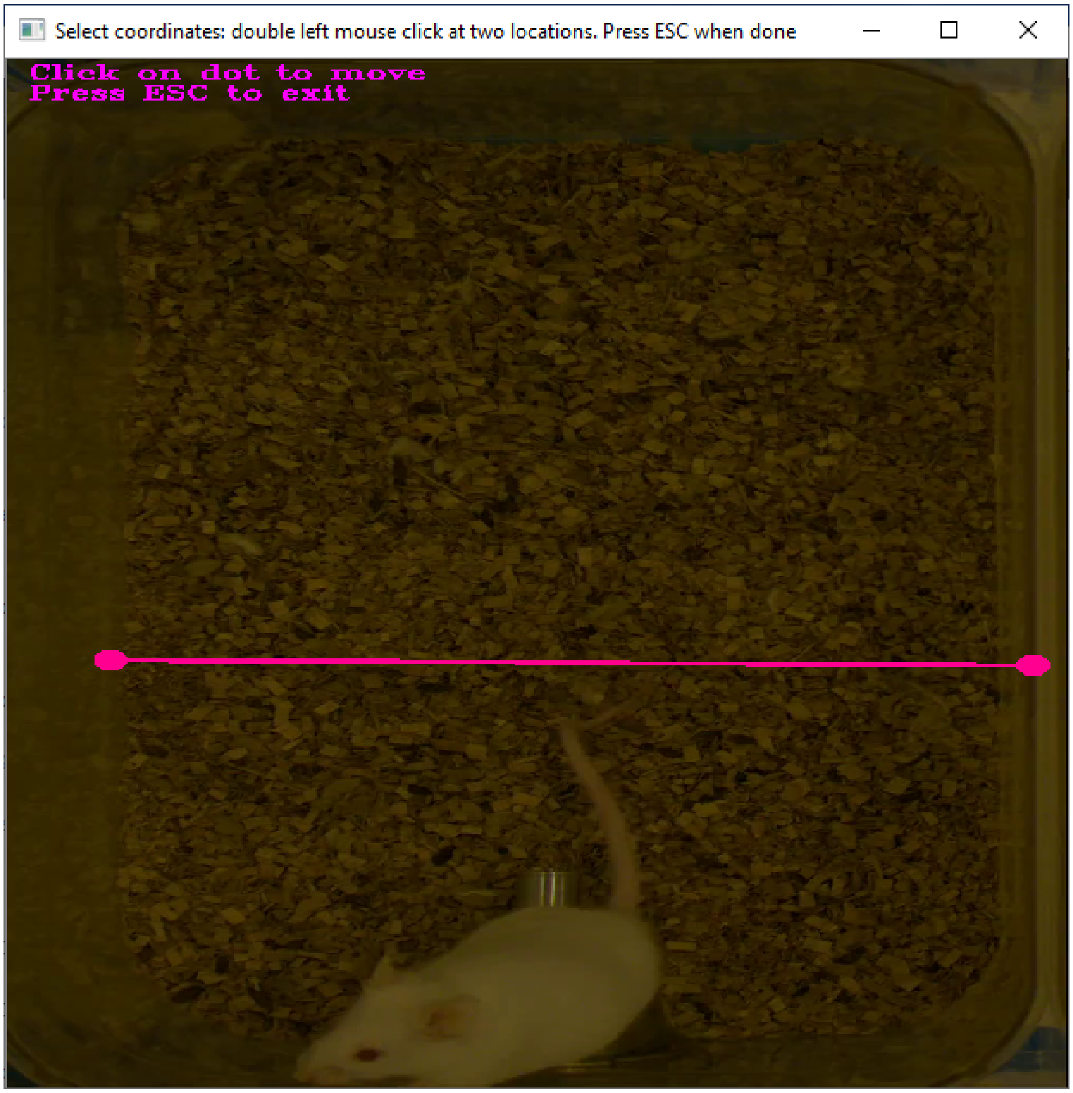
1. First, click on `Browse File` to select a video.
2. Enter *130* in the `Known length in real life(mm)` entry box.
3. Click on `Get Distance`, and the following window will pop up.



Select coordinates: double left mouse click at two locations. Press ESC when done

Note: When the frame is displayed, it may not be shown at the correct aspect ratio. To fix this, drag the window corner to the correct aspect ratio.

4. Use your mouse to double *Left click* at the left side of the cage and double *Left click* again on the right side of the cage. These are the known distance of 130 mm.

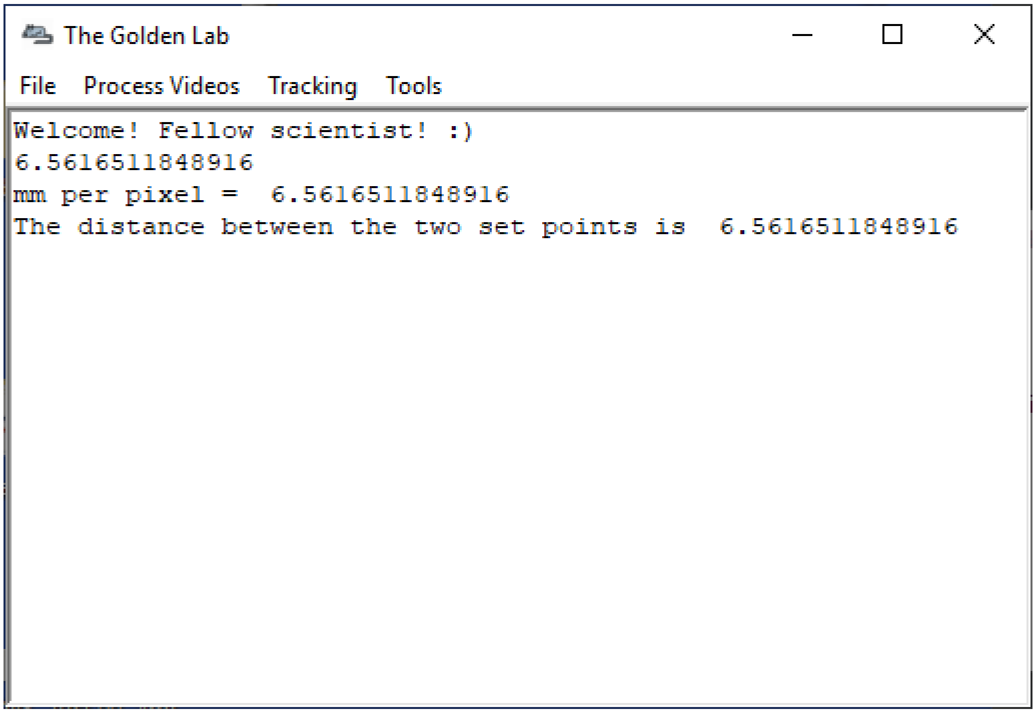


Select coordinates: double left mouse click at two locations. Press ESC when done

Click on dot to move
Press ESC to exit

Tip: You can double click any point again to change the location of the point.

5. Once two points are selected, hit `Esc` button. The millimeter per pixel value is printed in the main SimBA interface.



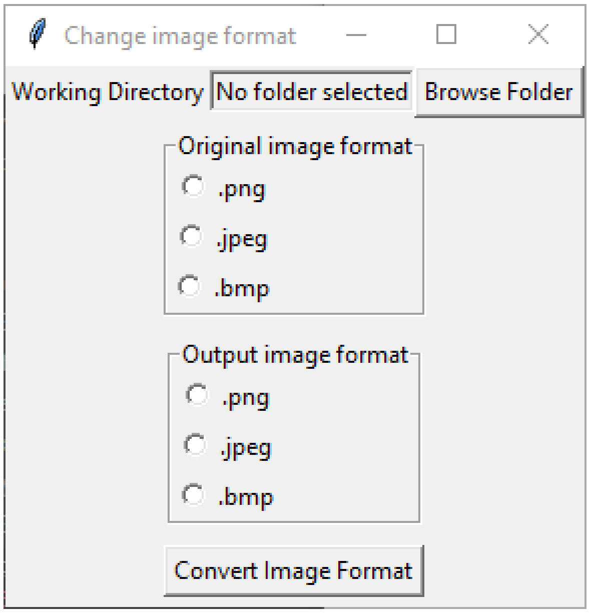
```
Welcome! Fellow scientist! :)
6.5616511848916
mm per pixel =  6.5616511848916
The distance between the two set points is  6.5616511848916
```

3.10.8 Change formats

This menu includes **Change image formats** and **Change video formats**

Change image formats

This tool allows the user to select a folder containing multiple images and convert the formats.

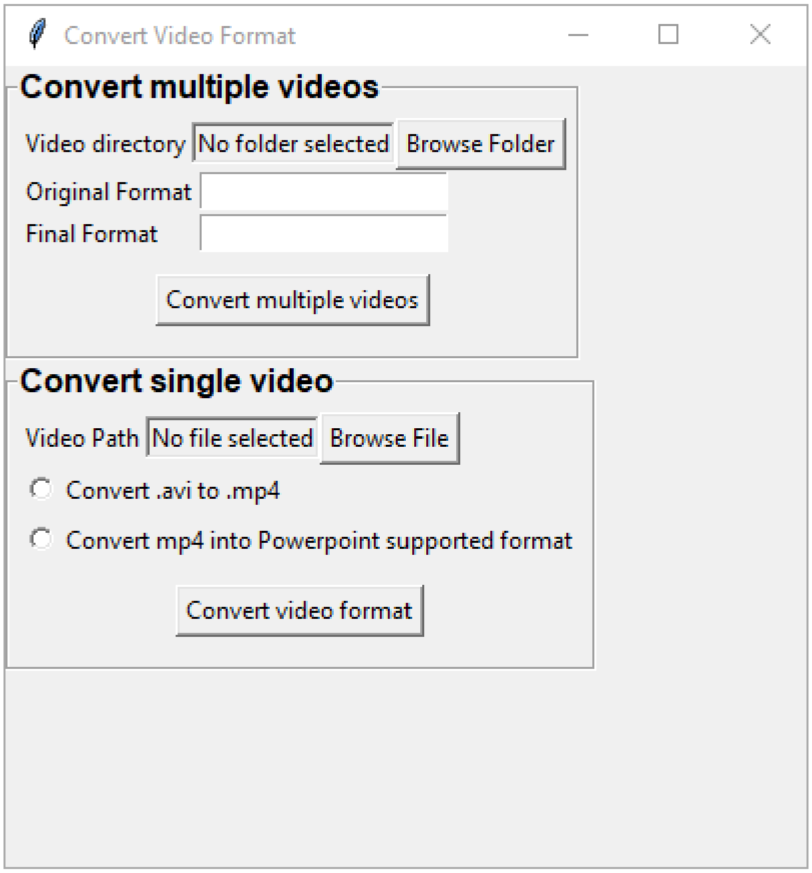


1. Click on `Browse Folder` to select a folder that contains multiple images.

2. Choose the original format of the images in the selected folder.
3. Choose the desired output image format.
4. Click on `Convert image file format`.

Change video format

This tool allows the user to convert the file format of a single or multiple videos.



Convert multiple videos

1. Click on `Browse Folder` to select the directory that contains the videos that you want to convert.
2. Enter the original file format (eg: mp4, flv, avi etc.) in the `Input format` entry box. **Note: do not put dots ('.') in the file format name (eg: mp4 or flv, etc)**.
3. Enter the desired output format in the `Output format` entry box .
4. Click on `Convert multiple videos`.

Convert single video

1. Click on `Browse File` to select a video to convert.
2. Choose one of the following `Convert .avi to .mp4` or `Convert mp4 into Powerpoint supported format`
3. Click on `Convert video format`.

3.10.9 CLAHE enhance video

1. Click on `Browse File` and select a video file.
2. Click `Apply CLAHE`. The new video will have a name of *CLAHE_ Name of original video*. The new file will be in a **.avi** format will be located in the same folder as the original video.

3.10.10 Superimpose frame numbers on video

This tool creates a video with the frame numbers printed on top of the video.

1. Click on `Superimpose frame numbers on video` and a new window will pop up.
2. Select a video and click on `Open`.
3. The new version of the video will be created with the name *Name of original video* + *_frame_no* and will be located in the same folder as the original video.

3.10.11 Convert to grayscale

1. Click on `Convert to grayscale` and a new window will pop up.
2. Select video and click `Open`.
3. The new greyscale version of the video will be created and have the name *Name of original video* + *_grayscale*. The new video will be located in the same folder as the original video.

3.10.12 Merge images to video

1. Click on `Browse Folder` to select a folder containing multiple frames.
2. Enter the input image format of the frames. Eg: if the image name is *Image001.PNG*, enter *PNG* in the `Image format` entry box.
3. Enter the desire output video format. Eg: if the video should be in *.mp4* file format, enter *mp4* in the `Video format` entry box.
4. Enter the desire frames per second in the `fps` entry box.
5. Enter the desired bitrate for the video in the `Bitrate` entry box.
6. Click on `Merge Images` to create the video.

3.10.13 Generate gifs

1. Click on `Browse File` and select a video to convert to a GIF.
2. Enter the starting time of the GIF from the video in the `Start times(s)` entry box.
3. Enter the duration of the GIF in the `Duration(s)` entry box.
4. Enter the size of the GIF in the `Width` entry box. The output GIF will be scale automatically.
5. Click on `Generate gif` to create the gif.

3.10.14 Extract Frames

The Extract frames menu has two options: **Extract defined frames**, **Extract frames**, and **Extract frames from seq files**.

Extract defined Frames

This tool allows users to extract frames from a video by inputting specific start- and end-frame numbers. This is useful if you want to extract a subset of frames from a larger video, without first needing to generate a new video of the desired length.

1. Click `Browse File` to select a video.
2. Enter the starting frame number in the `Start Frame` entry box.
3. Enter the ending frame number in the `End Frame` entry box.
4. Click on `Extract Frames` to extract the frames from the `Start Frame` to the `End Frame`.
5. A folder with the video name will be generated and the all the extracted frames will be located in the folder. The frames will be in *.png* file format.

Extract frames

Use this tool to extract every frame from a single video or multiple videos.

Single video

1. Click on `Browse File` to select a video file.
2. Click on `Extract Frames(Single video)` to extract every frame from the video.
3. A folder with the video name will be generated and the all the extracted frames will be located in the folder. The frames will be in *.png* file format.

Multiple videos

1. Click on `Browse Folder` to select the folder with videos.
2. Click on `Extract Frames(Multiple videos)` to extract every frame from the video.
3. Folders with the video name will be generated and the all the extracted frames will be located in the folders. The frames will be in *.png* file format.

Extract frames from seq files

Use this tool to extract all the frames from a video in **seq** file format.

1. Click on `Browse File` to select a video.
2. Click on `Extract All Frames` to extract all the frames from the video.
3. A folder with the video name will be generated and the all the extracted frames will be located in the folder. The frames will be in *.png* file format.

Convert . seq files to .mp4 files

Use this tool to convert .seq files to .mp4 files.

1. Click on `Tools`, then `Change formats`, and click on `Change .seq to .mp4`.
2. A window will pop up and you can then navigate and select the video folder that contains the mp4's.
3. The conversion progress can be followed through the progress bar printed in the terminal window.