DeepLabCut AI Residency

Day 2 Session 2:
Network evaluation & video analysis

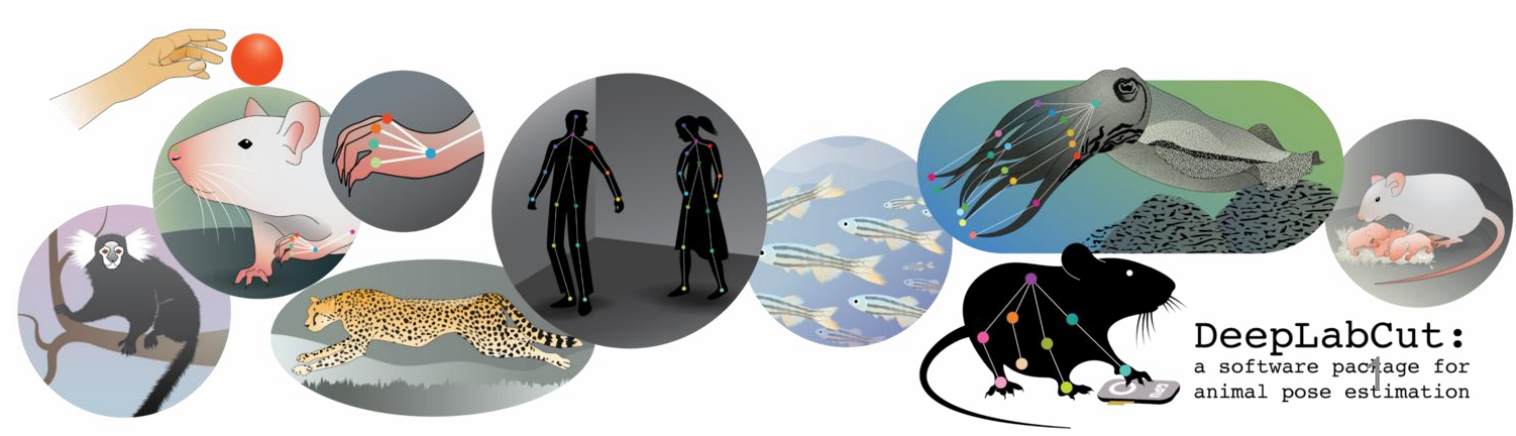
July 30 & August 1, 2025

McGill University, Montreal

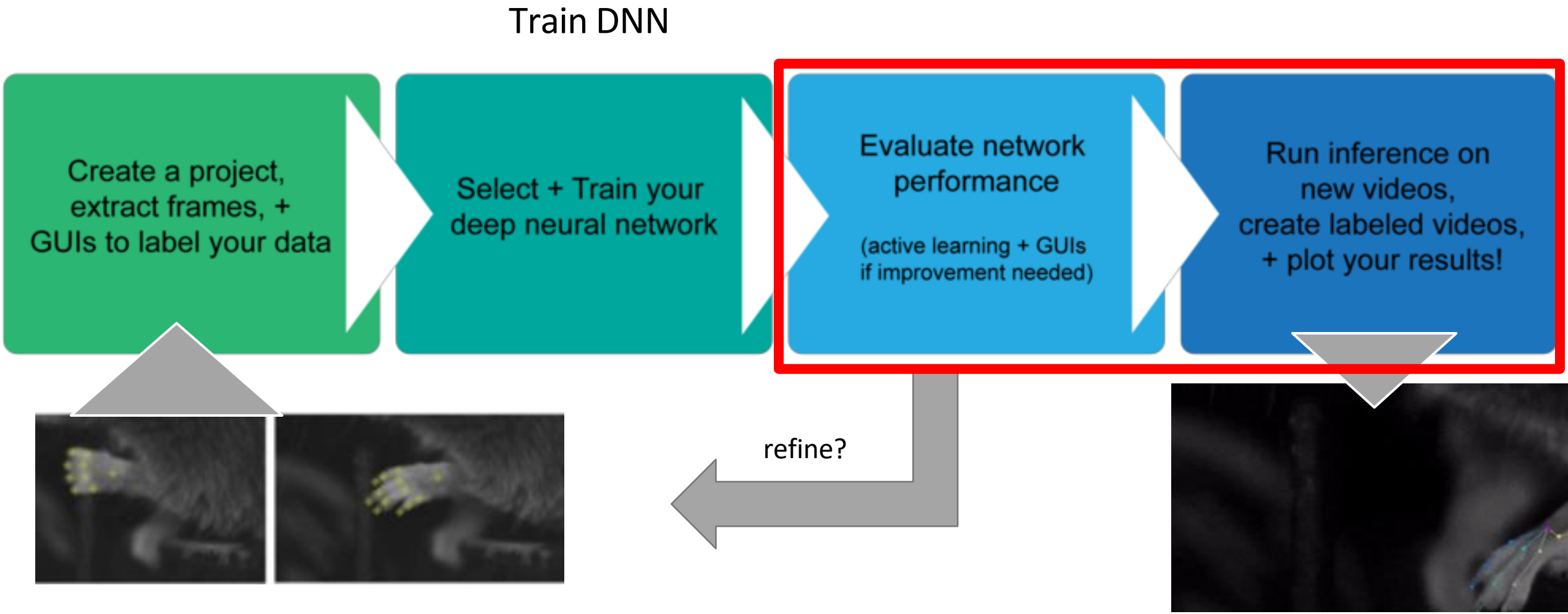
Jiayue Yang

Vic Shao-Chinh Chiang

McGill



Recall: DeepLabCut workflow



Nath et al. (2019). Nature Protocols.

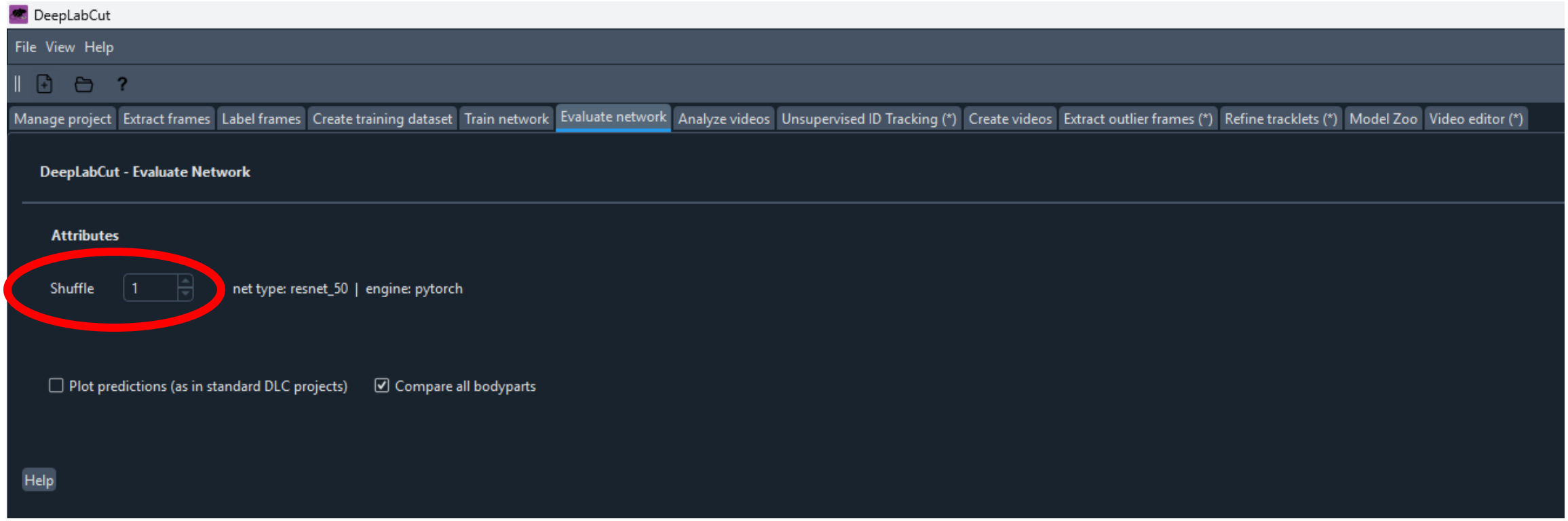
Summary of DLC commands

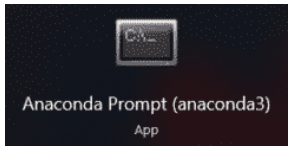
Table 1 | Summary of commands

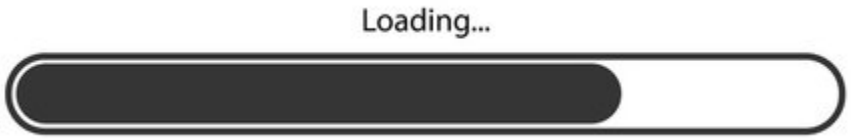
Operation	Command
Open IPython and import DeepLabCut (Step 1)	`ipython` `import deeplabcut`
Create a new project (Step 2)	`deeplabcut.create_new_project('project_name','experimenter',['path of video 1','path of video2',...])`
Set a config_path variable for ease of use (Step 3)	`config_path = '/yourdirectory/project_name/config.yaml'`
Extract frames (Step 4)	`deeplabcut.extract_frames(config_path)`
Label frames (Steps 5 and 6)	`deeplabcut.label_frames(config_path)`
Check labels (optional)(Step 7)	`deeplabcut.check_labels(config_path)`
Create training dataset (Step 8)	`deeplabcut.create_training_dataset(config_path)`
Train the network (Step 9)	`deeplabcut.train_network(config_path)`
Evaluate the trained network (Step 11)	`deeplabcut.evaluate_network(config_path)`
Video analysis and plotting results (Step 11)	`deeplabcut.analyze_videos(config_path, ['path of video 1 or folder','path of video2',…])`
Video analysis and plotting results (Step 12)	`deeplabcut.plot_trajectories(config_path, ['path of video 1','path of video2',...])`
Video analysis and plotting results (Step 13)	`deeplabcut.create_labeled_video(config_path, ['path of video 1','path of video2',...])`
Refinement: extract outlier frames (Step 14)	`deeplabcut.extract_outlier_frames(config_path,['path of video 1','path of video 2'])`
Refine labels (Step 15)	`deeplabcut.refine_labels(config_path)`
Combine datasets (Step 16)	`deeplabcut.merge_datasets(config_path)`

Nath et al. (2019). Nature Protocols.
https://www.nature.com/articles/s41596-019-0176-0/tables/1

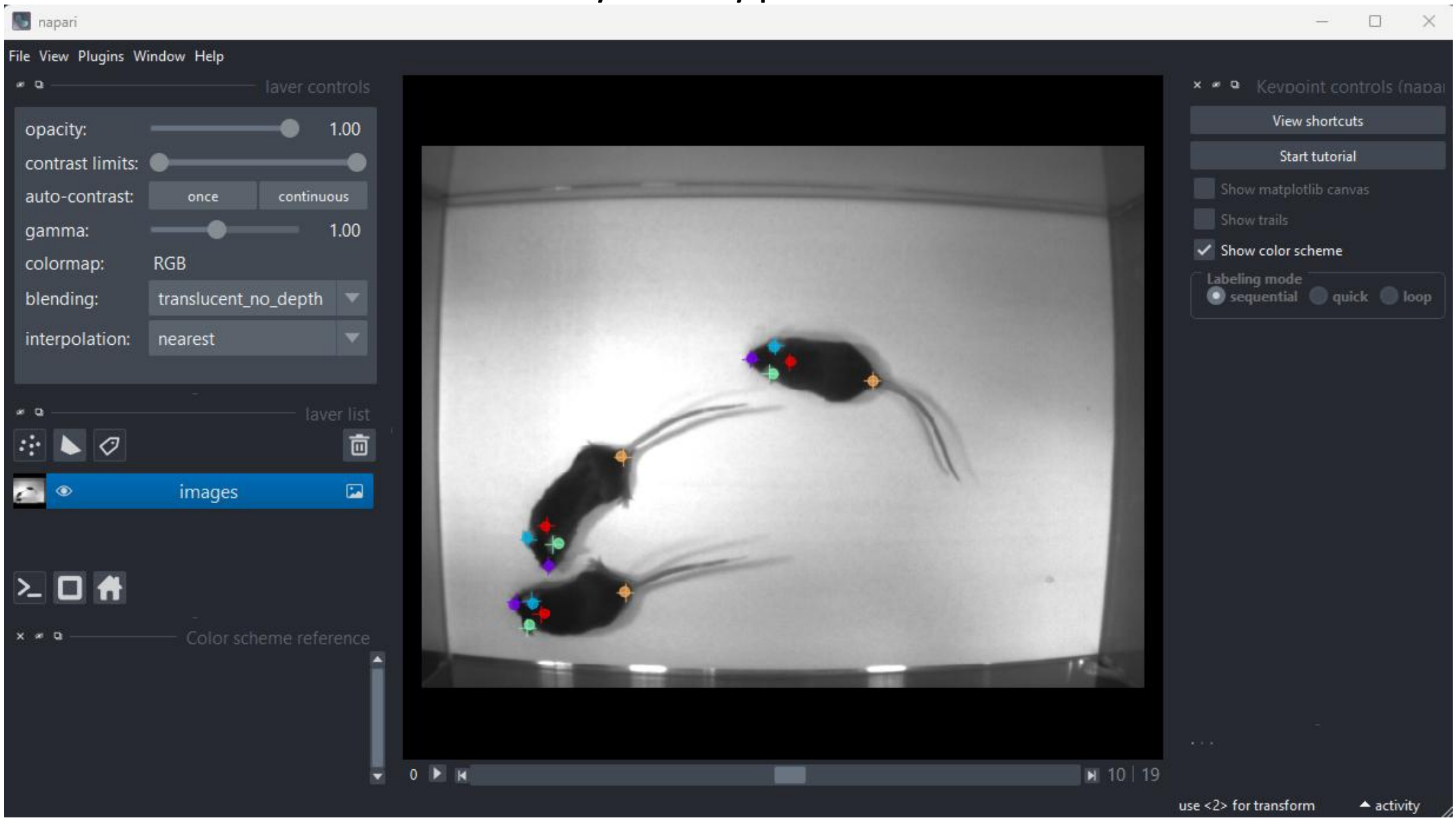
Network evaluation!





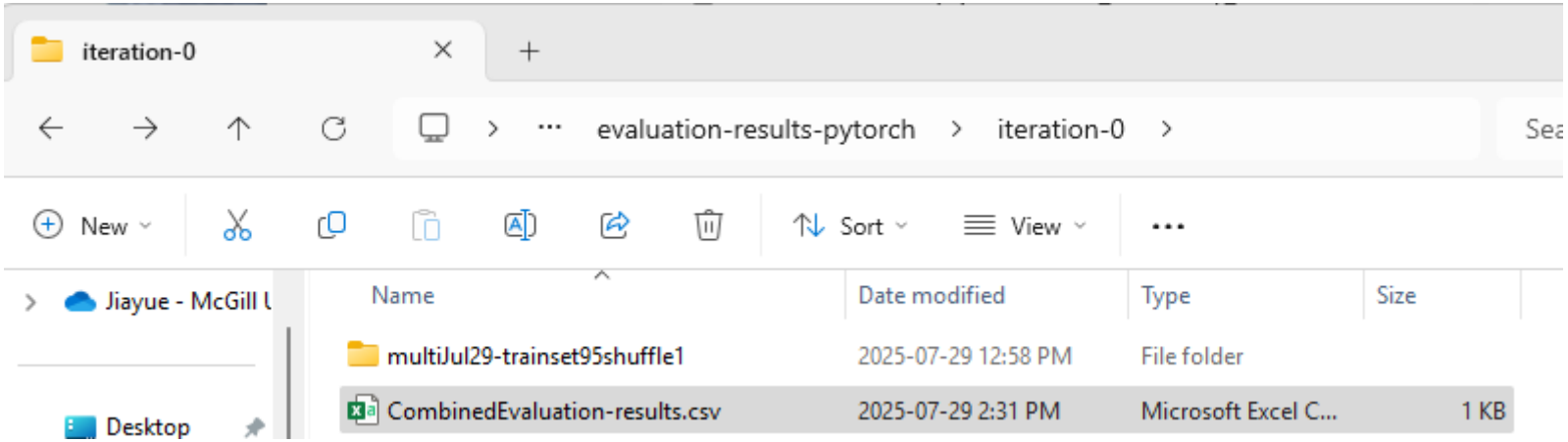


By bodyparts:



Evaluation results

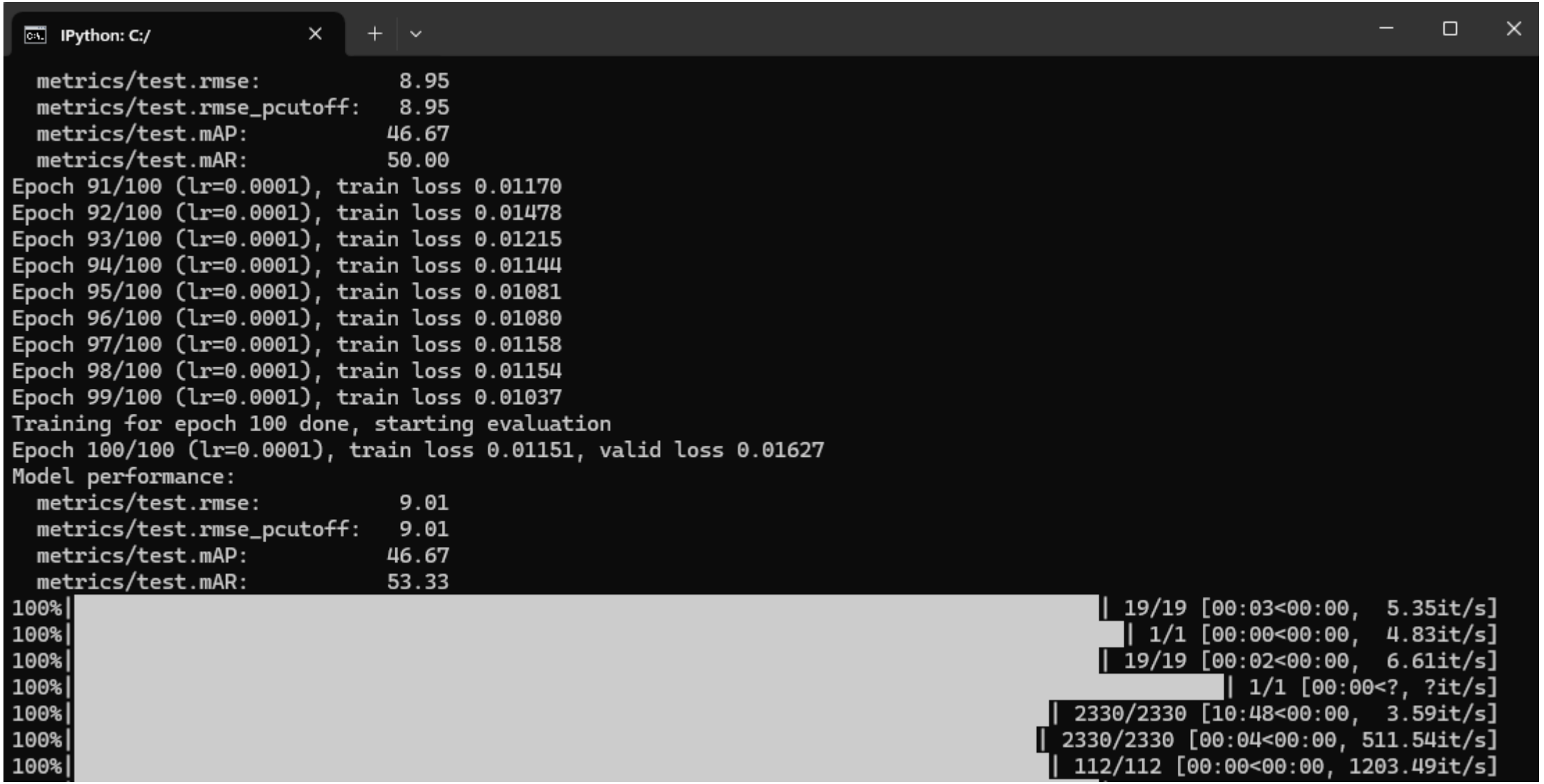
%Training dataset	Shuffle number	Training epochs	Detector epochs (TD only)	pcutoff	train rmse	train rmse_pcutoff	train mAP	train mAR	train id_head_snout_accuracy	train id_head_leftear_accuracy	train id_head_rightear_accuracy	train id_head_tailbase_accuracy	test rmse	test rmse_pcutoff	test mAP	test mAR	test id_head_snout_accuracy	test id_head_leftear_accuracy	test id_head_rightear_accuracy	test id_head_tailbase_accuracy
0.95	1	100	-1	0.6	28.71	24.22	48.37	53.33	0.84	0.82	0.86	0.82	9.01	9.01	46.67	53.33	0.5	0.33	0.67	0.67



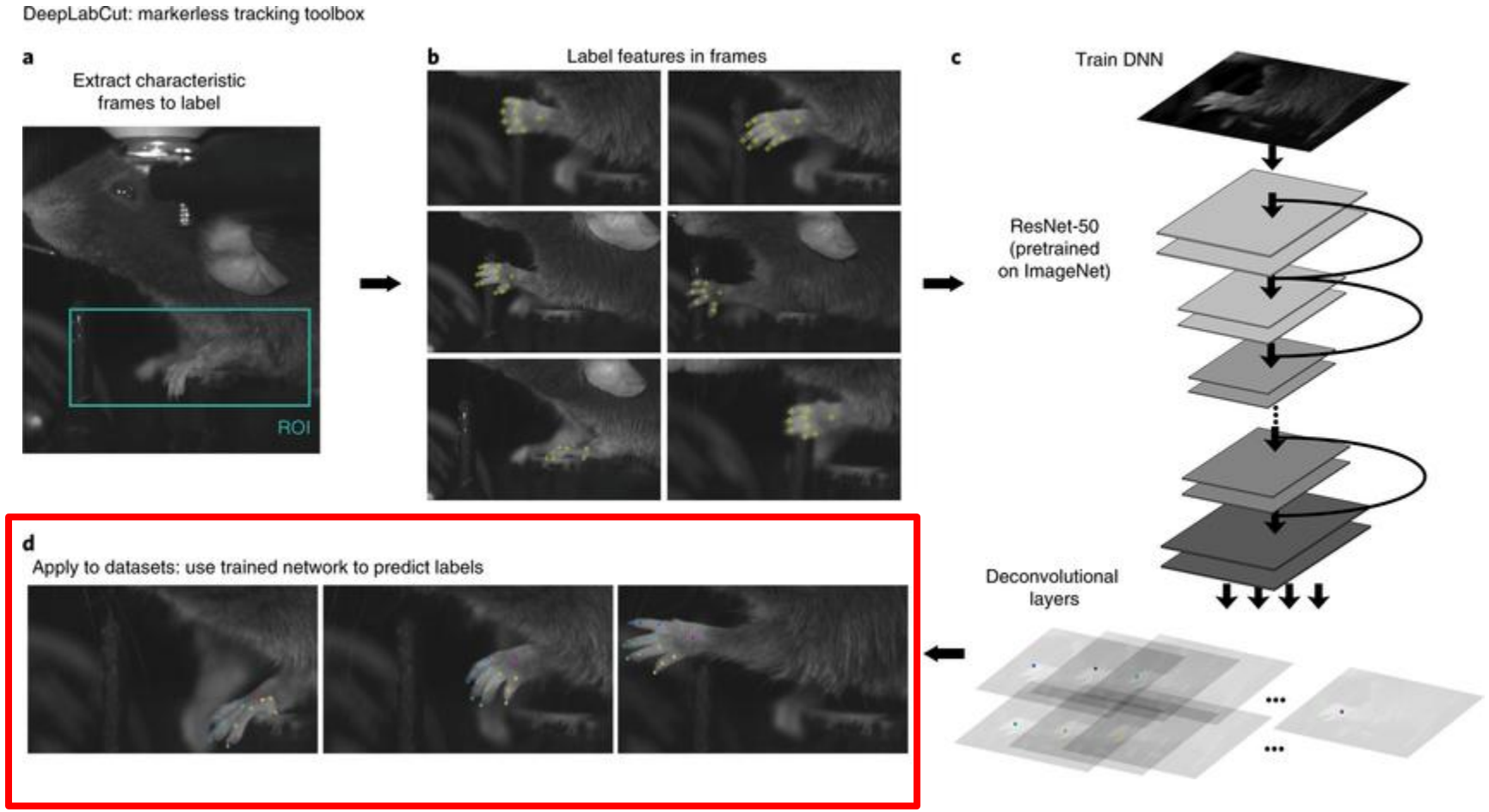
What do they mean?

Columns	Meanings
%Training dataset	0.95 = 95% labeled data for training. 5% held separated for testing
Shuffle number	Shuffle split (for cross-validation & training runs)
Training epochs	100 = # of training iterations
Detector epochs (TD only)	-1 = N/A (only for a top-down detector)
pcutoff	0.6 = confidence cutoff (prediction < 0.6 ignored)
train rmse	Root Mean Square Error on training set for all points, in pixels
train rmse_pcutoff	Training RMSE for predictions > 0.6
train mAP	Mean Average Precision (how precise key points localized in training set)
train mAR	Mean Average Recall (how good all points are detected)
test rmse	RMSE on testing set for all points, in pixels
test rmse_pcutoff	Testing RMSE for predictions > 0.6
test mAP	mAP on test set
test mAR	mAR on test set

Video analysis



Video analysis

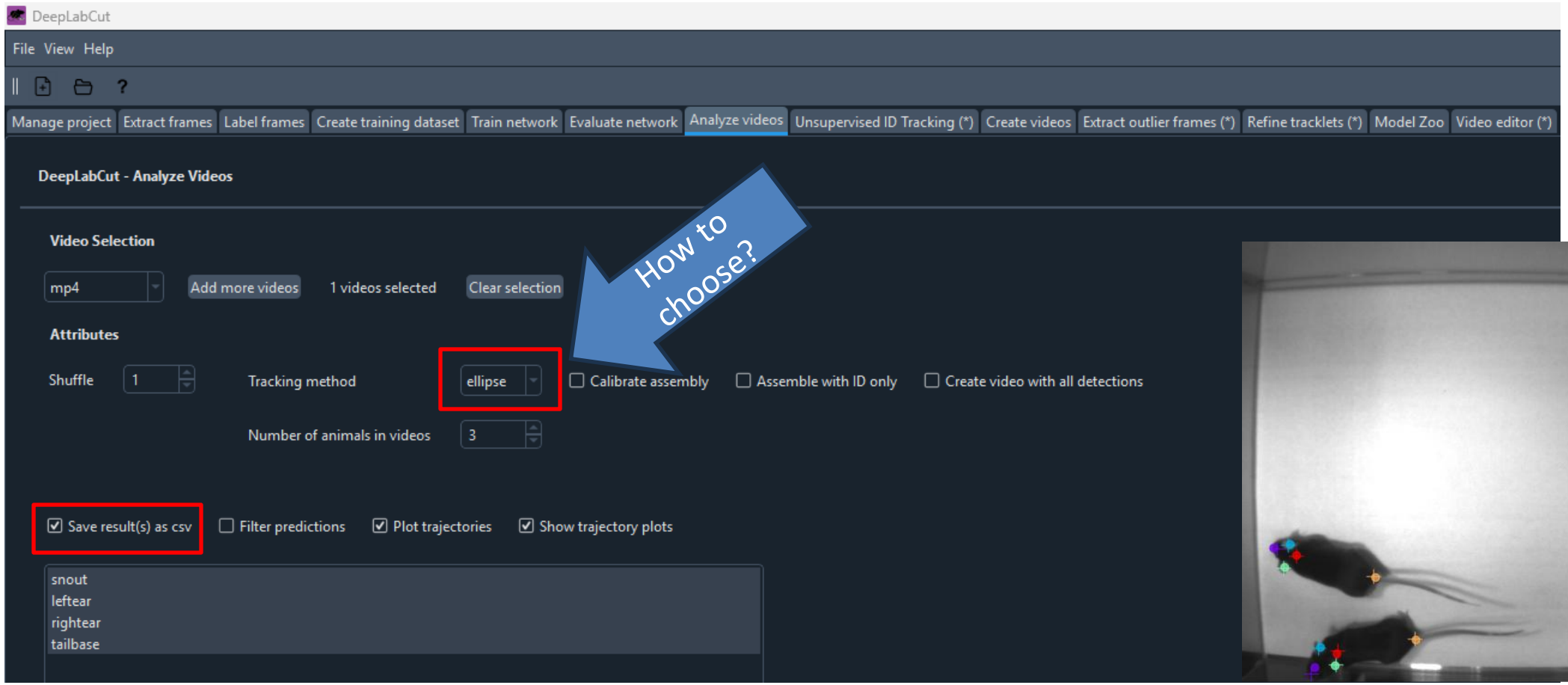


Can still edit config.yaml for parameters such as *batch_size*

You can simply **add new videos**, no need to add them in the cofig file!

Video analysis

deeplabcut.analyze_videos(path_to_config, [video_list], **save_as_csv=True**, ...)

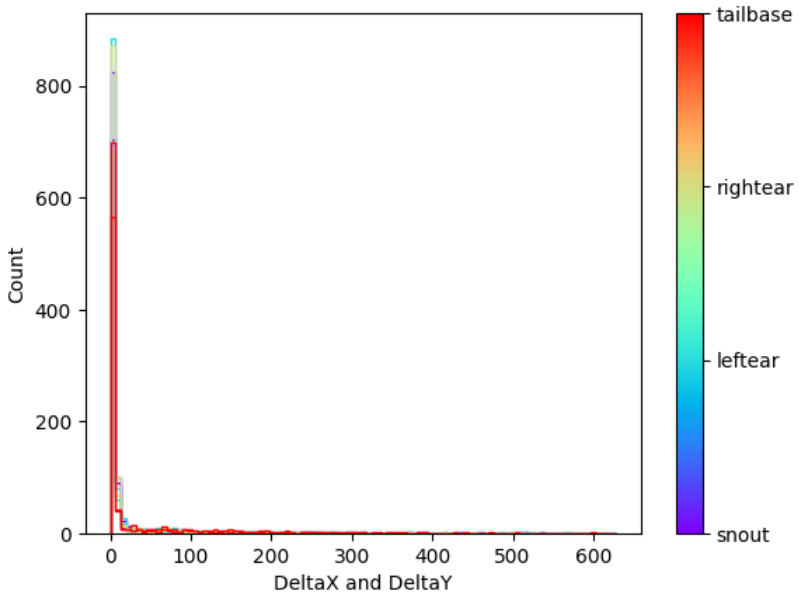


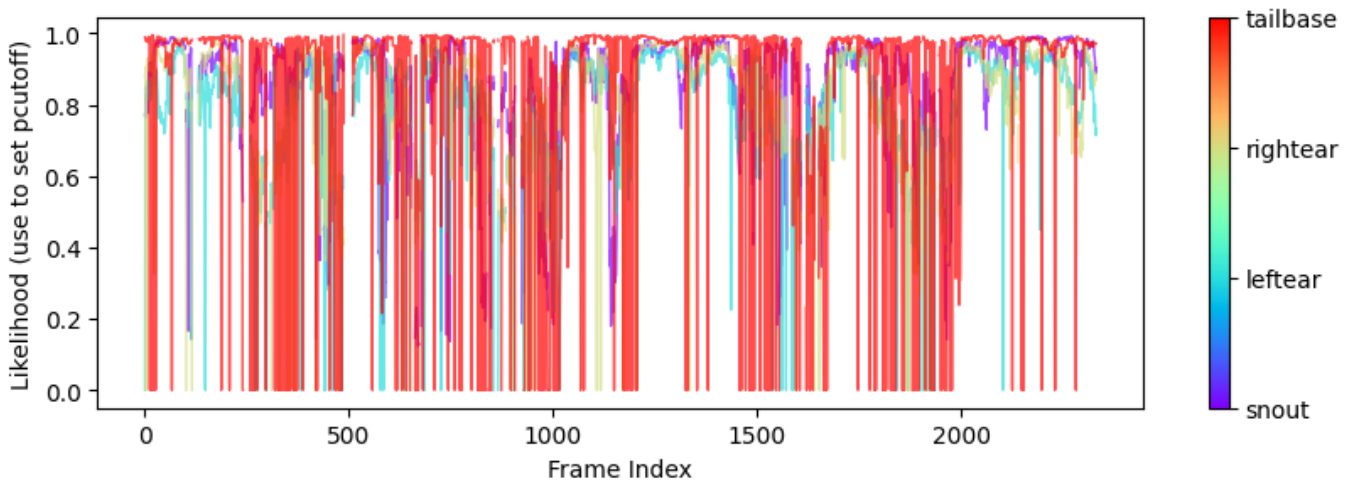
Tracking methods: same animal or not?

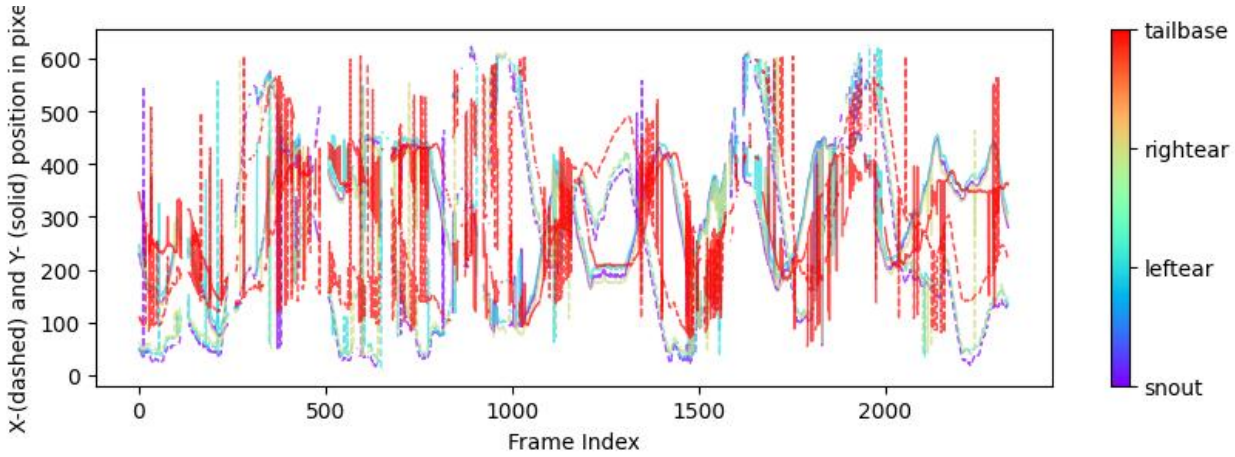
Ways to **link detections** of body parts of **consistent animals** across time

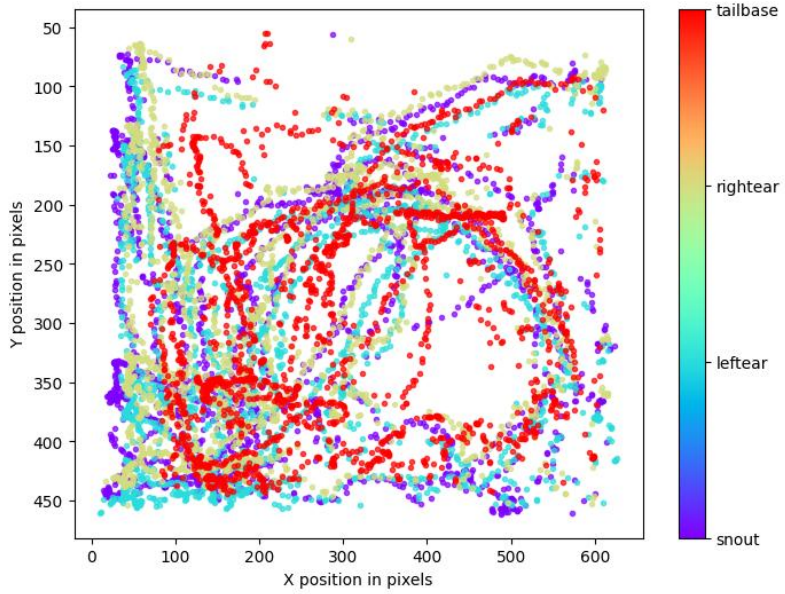
	What does it do?	Pro	Con
Ellipse	- Draw an ellipse to the key bodyparts/points of each detected animal - Tracks both spatial location and orientation (by angle of ellipse) of the animal across frames	- Fast - Good identity tracking when animals are close or rotating (**moderate interaction**)	- May mistake if similar animal (shape and pose) - Can mix-up while heavy occlusions or entangled animals
Box	- Draw a bounding box around all prediction of an animal to define its location in each frame - Tracks same animal by matching boxes at their locations across frames	- Fast - Good for **simple, spaced animal** - Not complex body parts - Not closely interacting	- Can be bad if animals overlap or interact close together - Only use spatial location (not internal structure)
Skeleton	- Draw a complete skeleton structure (considering spatial orientation of all key bodyparts and their connections) - Track same animal by matching shape, orientation, and geometry of the pose/skeleton	- Slower - Can track **complex behaviors** (grooming, crossing, etc.) - Dataset animals need to be consistent identity	- Requires higher computational power - Requires a well-defined and accurate skeleton structure for the animal

Trajectory: video analysis results



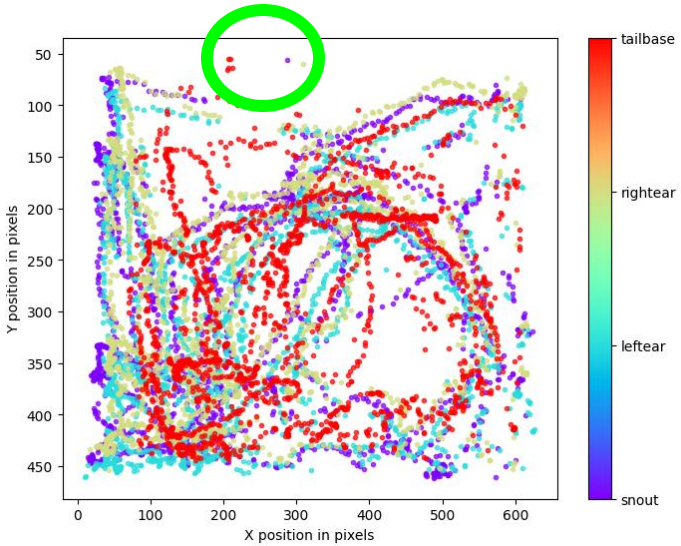


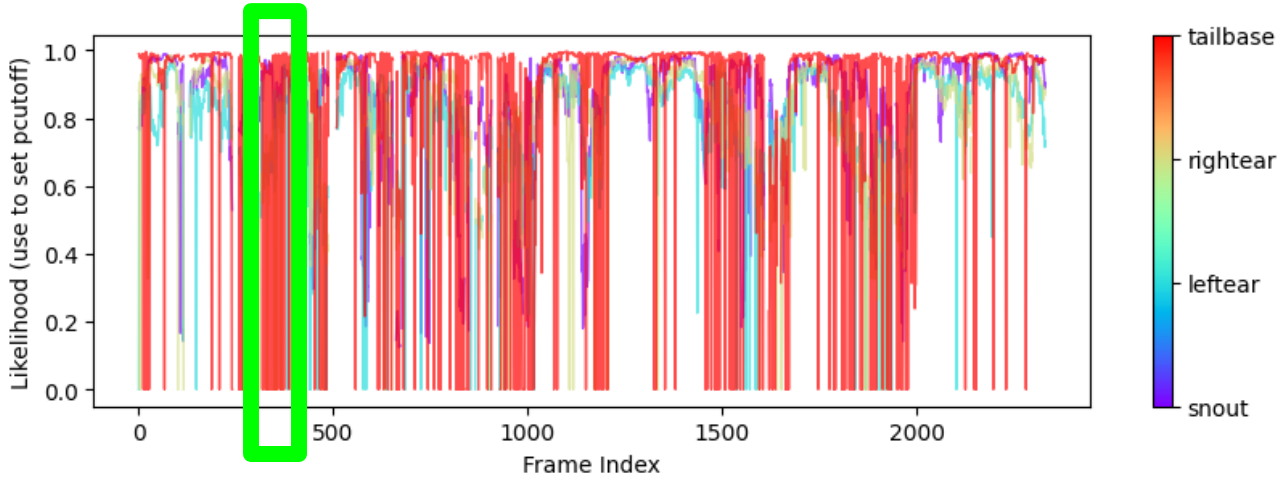




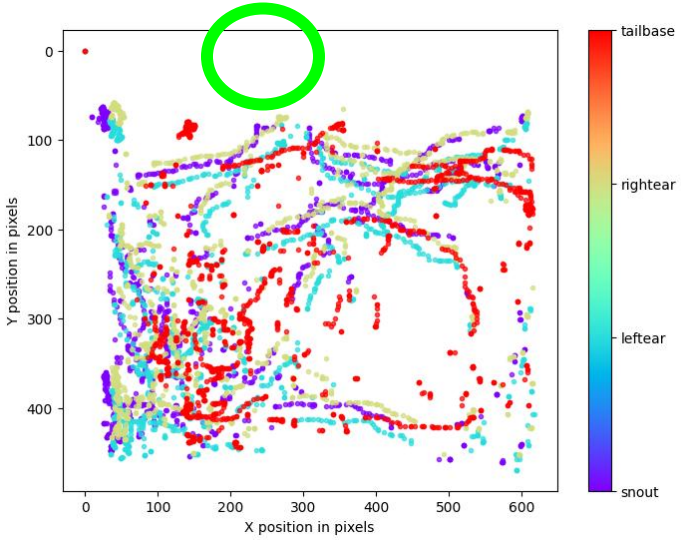
Filtering vs. unfiltered

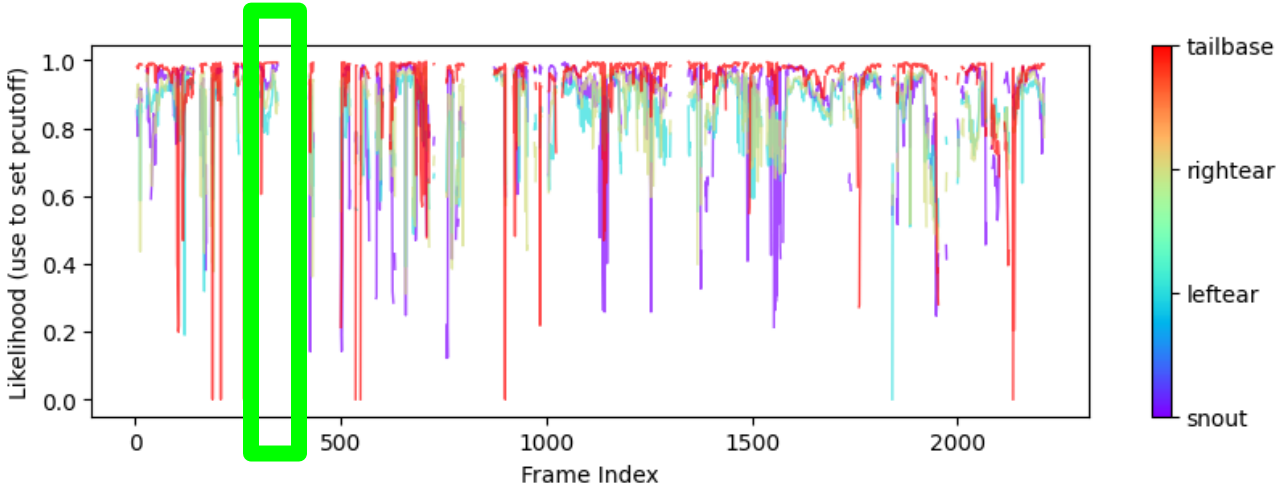
Ellipse Unfiltered





Skeleton Filtering





Labeled video creation

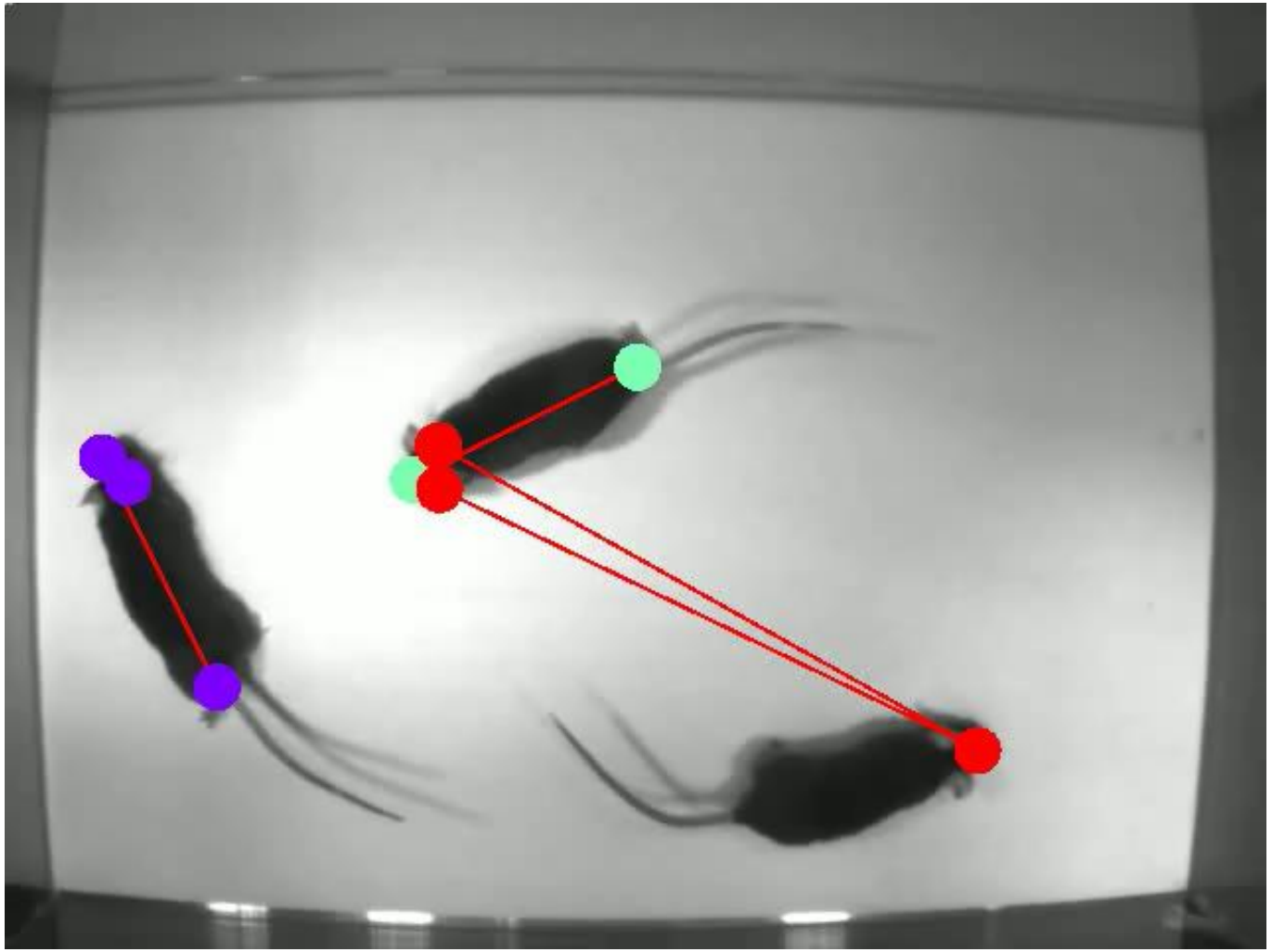
```
deeplabcut.create_labeled_video(config_path,['fullpath/afolderofvideos'], videotype='.mp4', filtered=True)
```

```
deeplabcut.create_labeled_video(config_path,['fullpath/afolderofvideos'], videotype='.mp4', draw_skeleton=True)
```

```
deeplabcut.create_labeled_video(config_path,['fullpath/afolderofvideos'], videotype='.mp4', trailpoints=10)
```

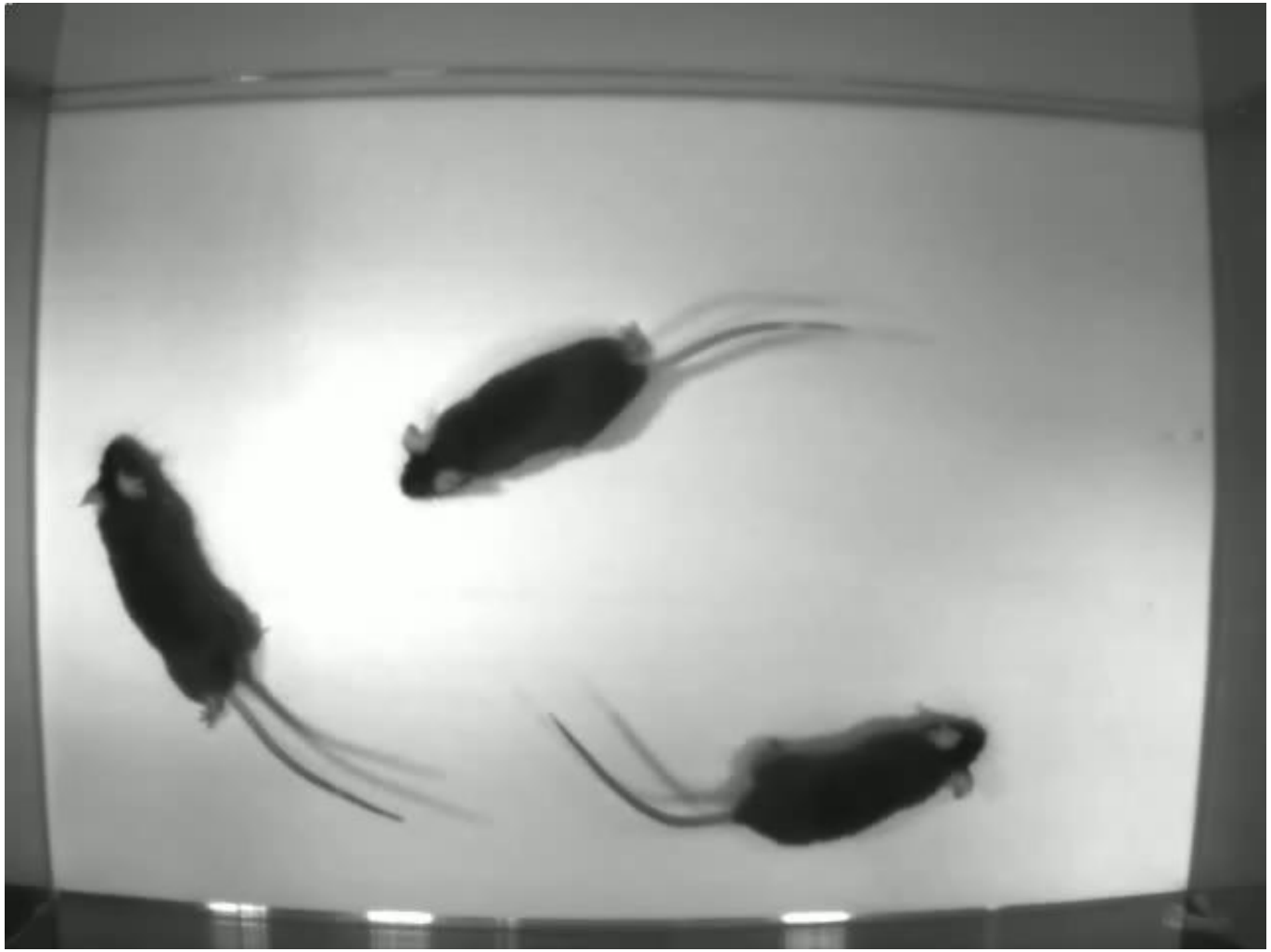
```
deeplabcut.create_labeled_video(config_path,['fullpath/afolderofvideos'], save_frames=True/False)
```

Unfiltered, ellipse



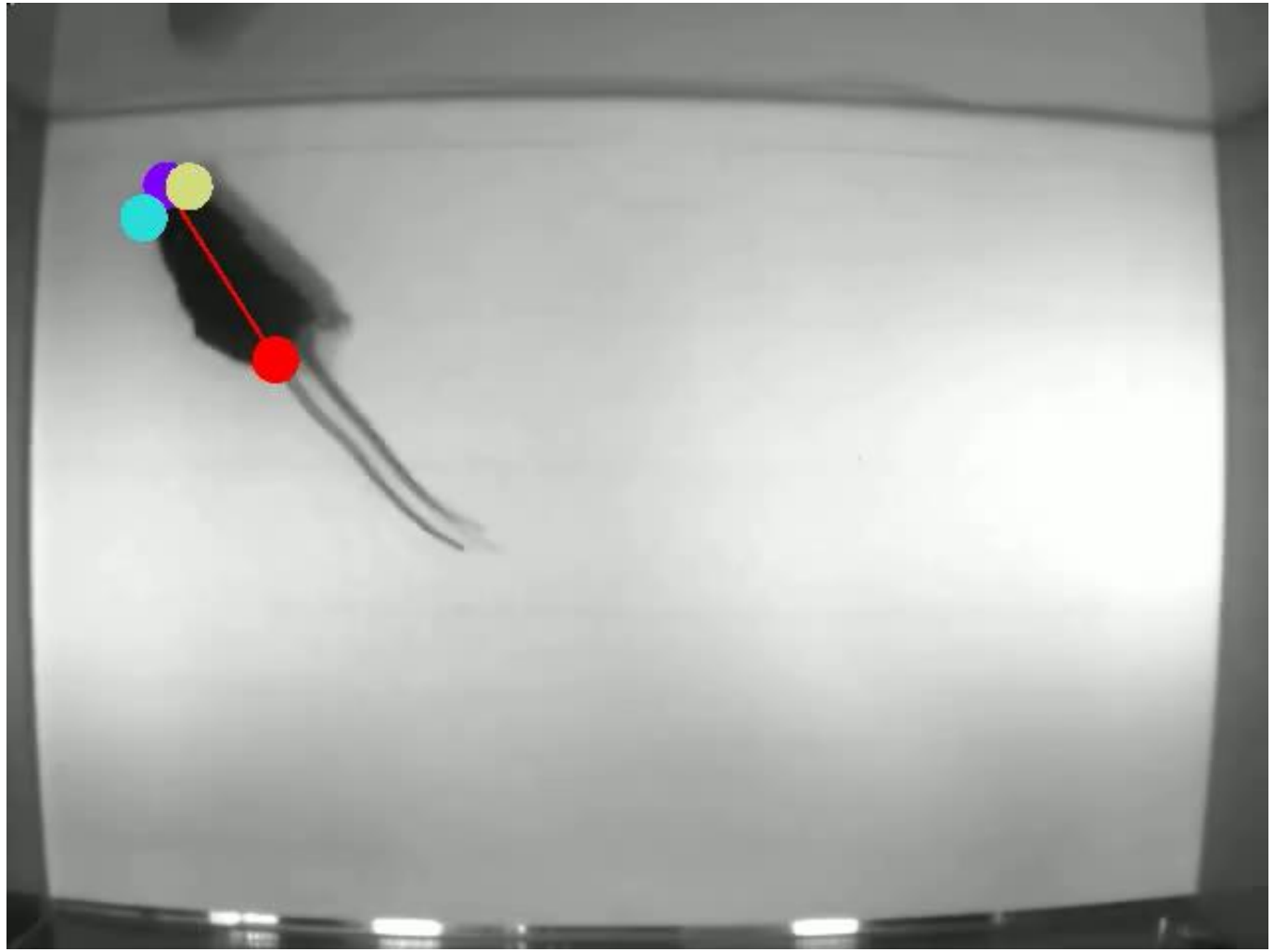
Under-trained!!!

Filtering, skeleton



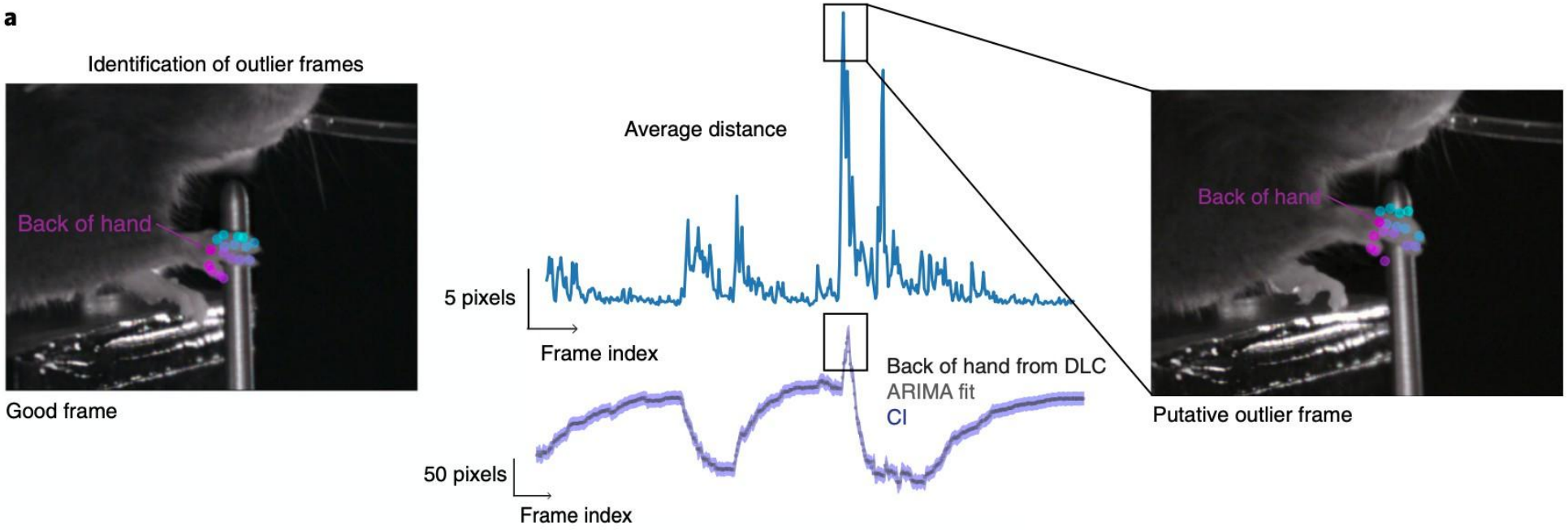
Under-trained!!!

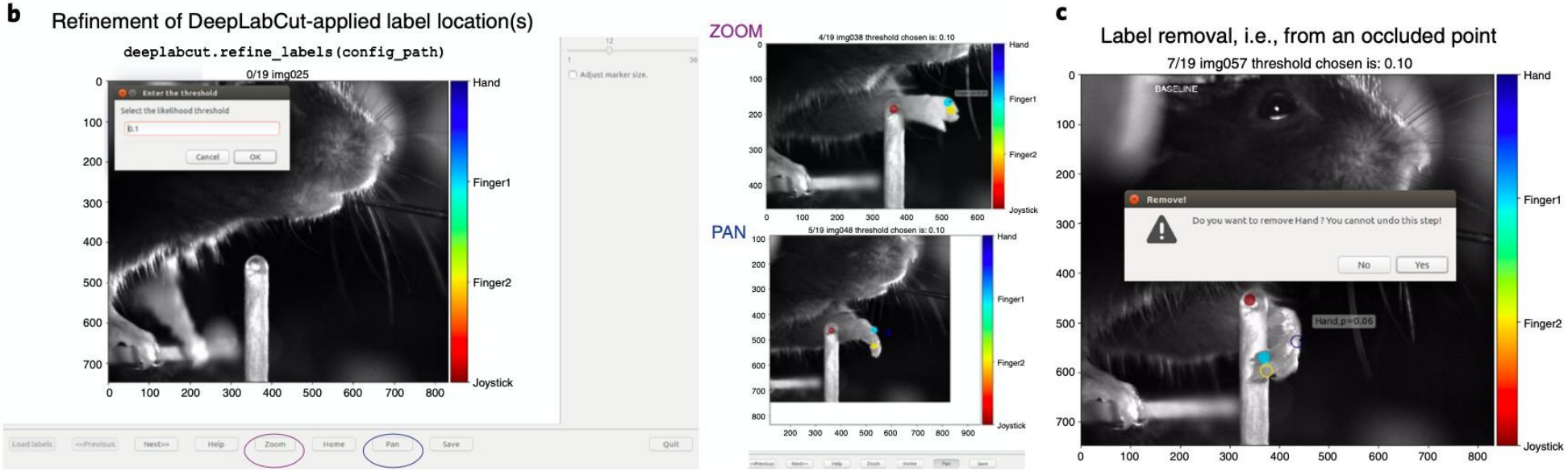
Single mice

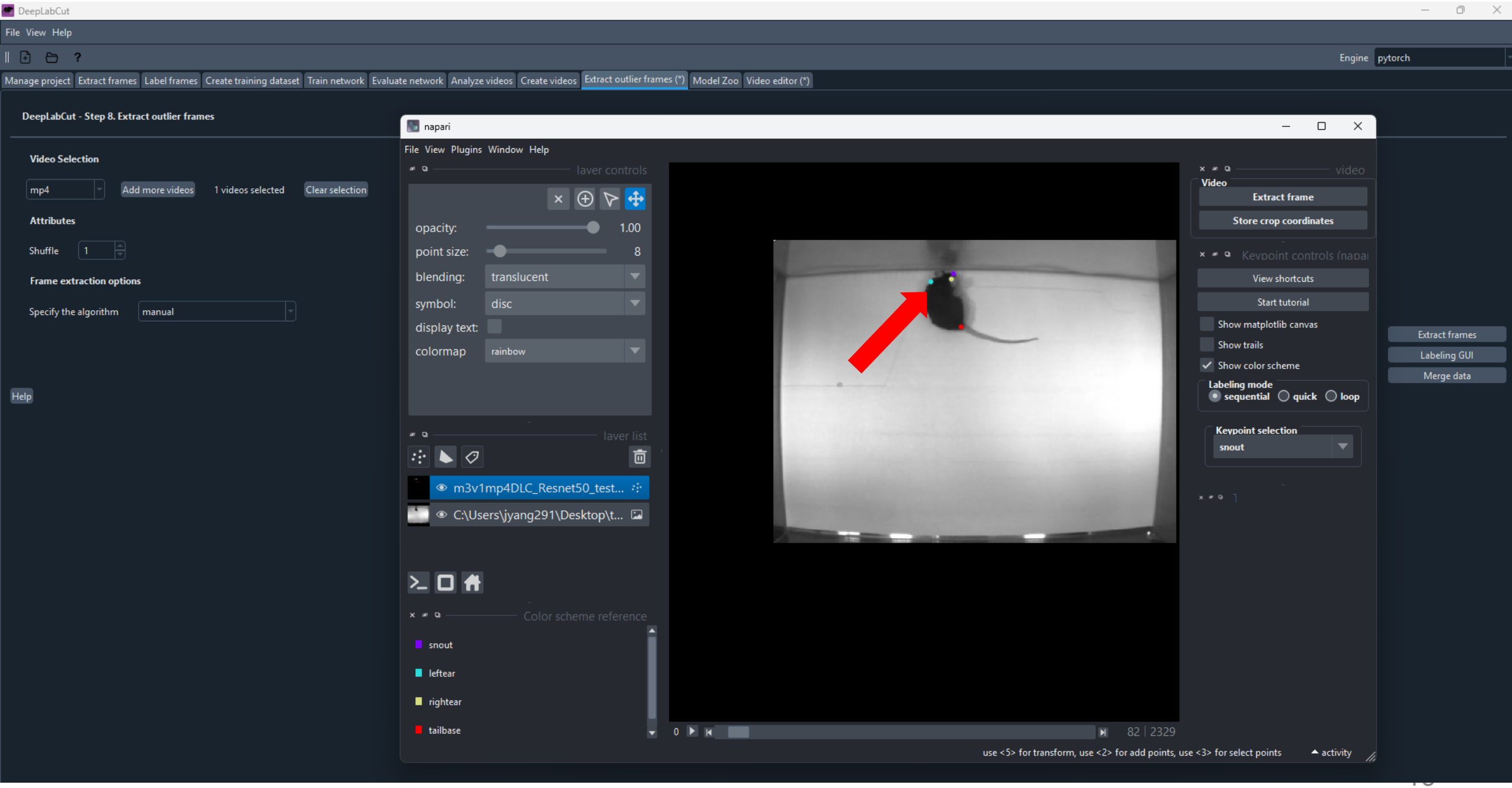


Under-trained!!!

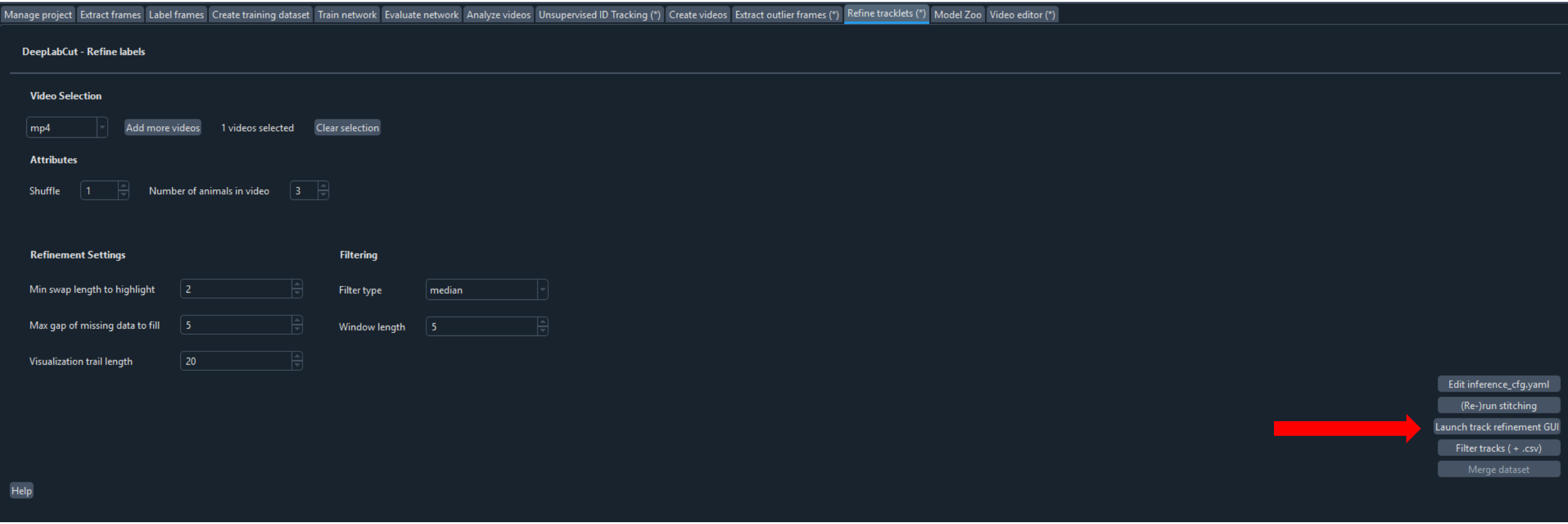
Nath et al. (2019). Nature Protocols.



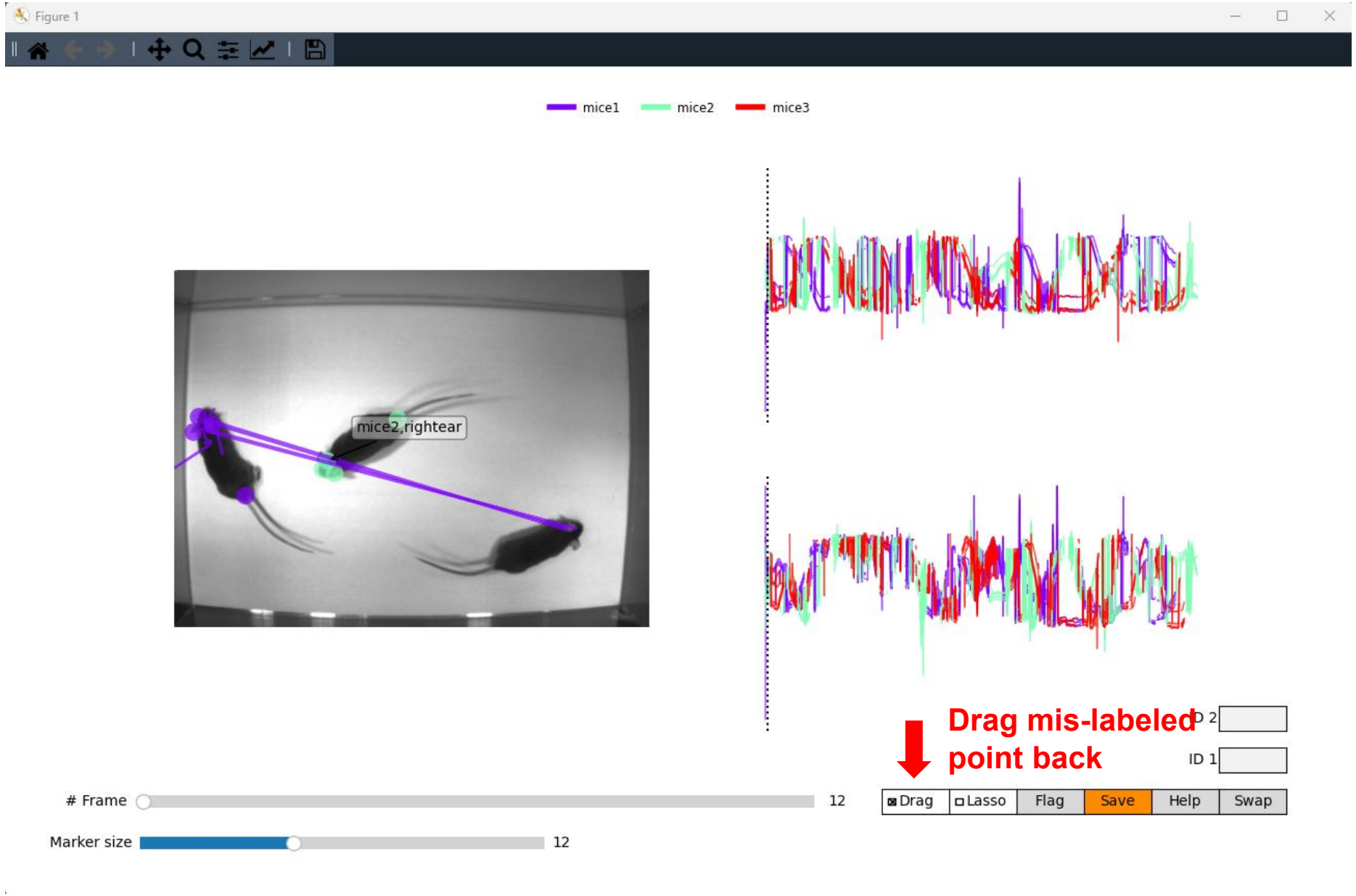



DeepLabCut
File View Help
Engine pytorch
Manage project Extract frames Label frames Create training dataset Train network Evaluate network Analyze videos Create videos Extract outlier frames (*) Model Zoo Video editor (*)
DeepLabCut - Step 8. Extract outlier frames
Video Selection
mp4
Add more videos
1 videos selected
Clear selection
Attributes
Shuffle
1
Frame extraction options
Specify the algorithm
manual
Help
napari
File View Plugins Window Help
layer controls
opacity:
1.00
point size:
8
blending:
translucent
symbol:
disc
display text:
colormap
rainbow
layer list
m3v1mp4DLC_Resnet50_test...
C:\Users\jyang291\Desktop\t...
Color scheme reference
snout
leftear
rightear
tailbase
video
Video
Extract frame
Store crop coordinates
Keypoint controls (napari)
View shortcuts
Start tutorial
Show matplotlib canvas
Show trails
Show color scheme
Labeling mode
sequential
quick
loop
Keypoint selection
snout
0
82 | 2329
use <5> for transform, use <2> for add points, use <3> for select points
activity
Extract frames
Labeling GUI
Merge data

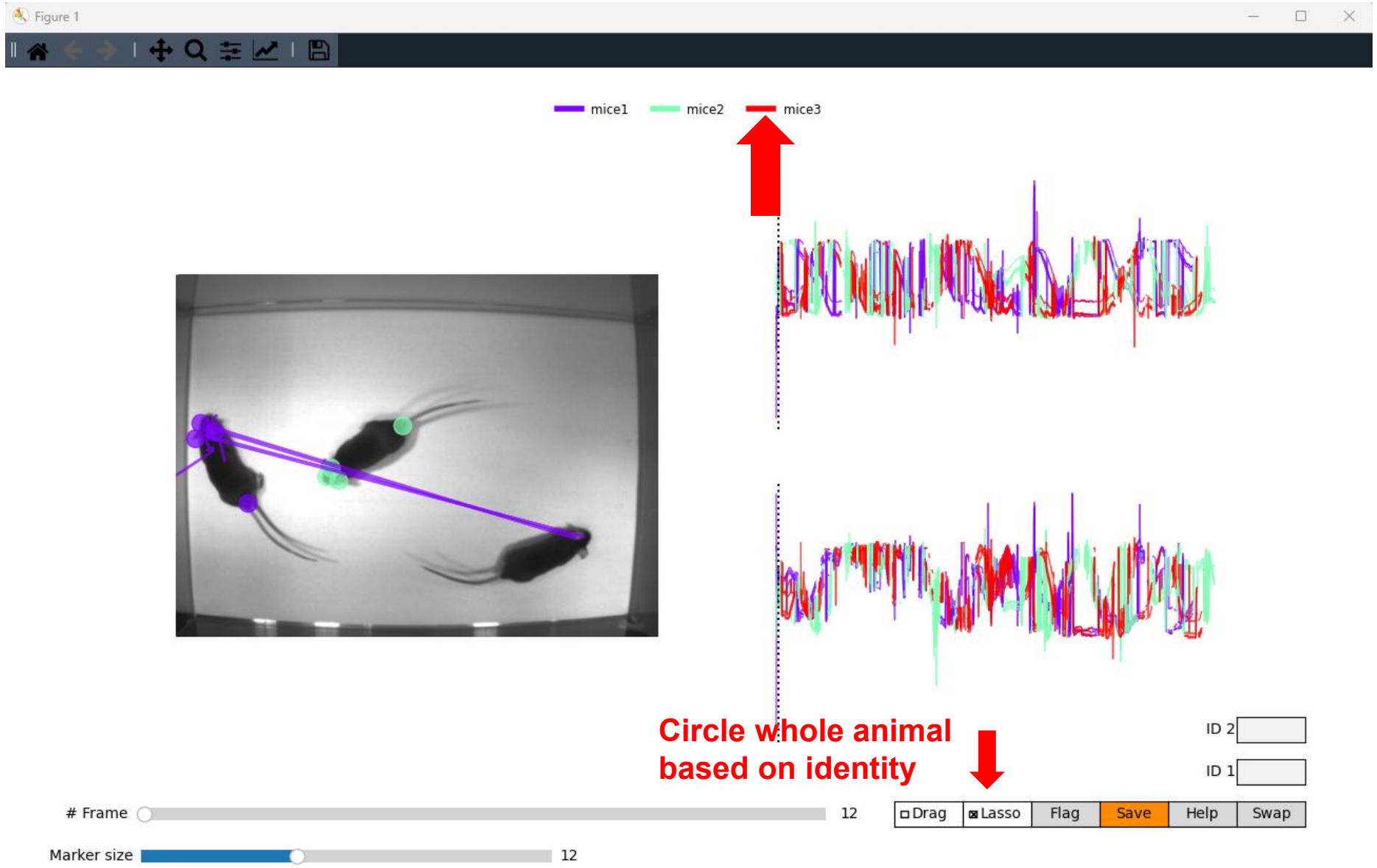
Refine tracklets



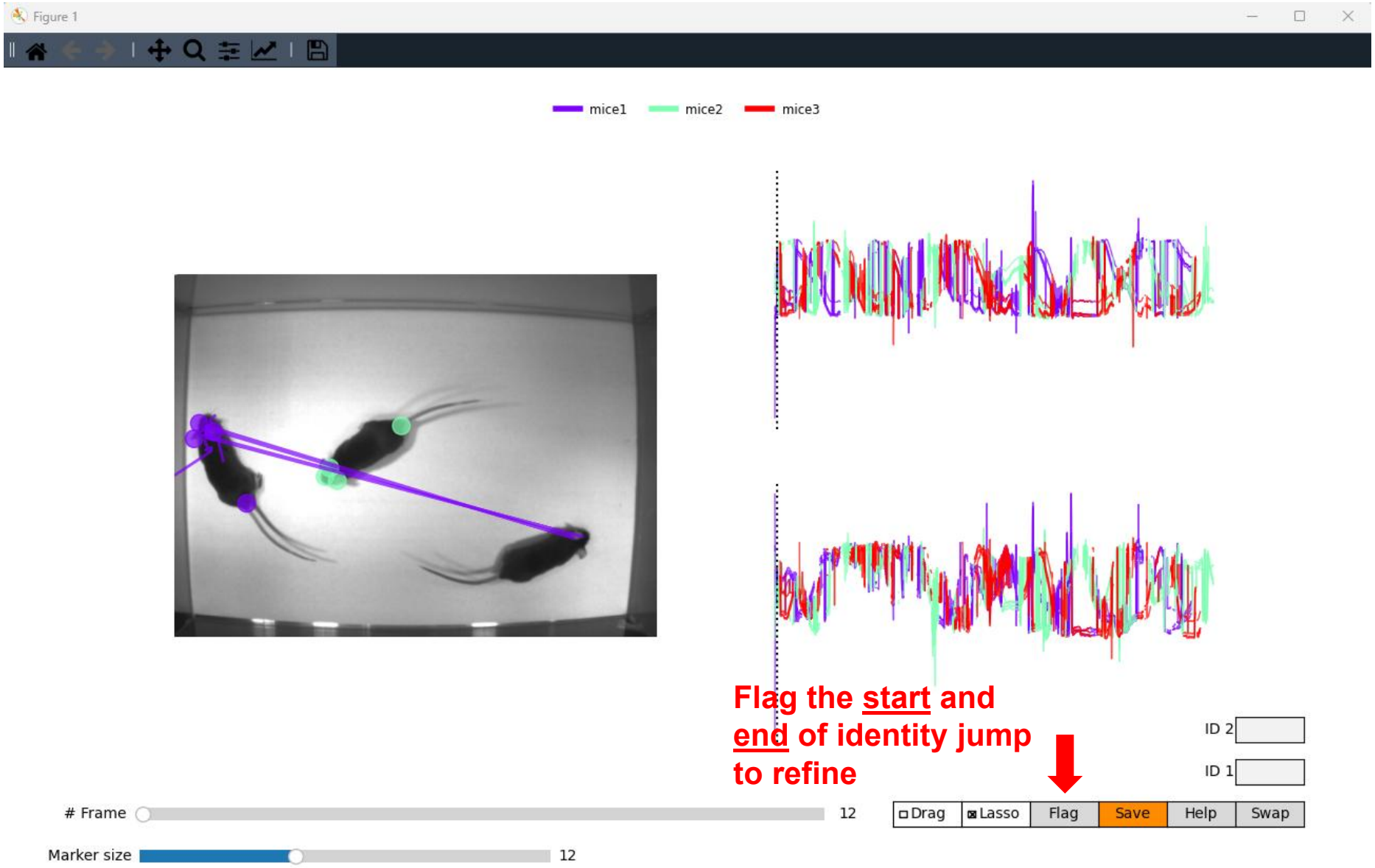
Refine tracklets



Refine tracklets



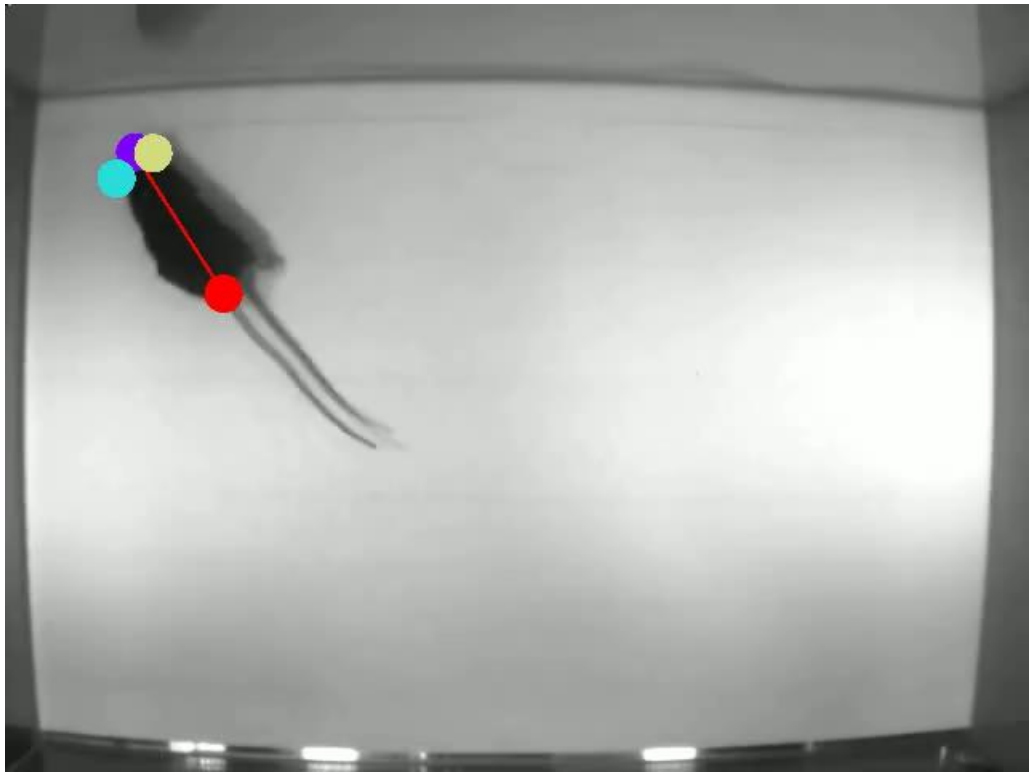
Refine tracklets



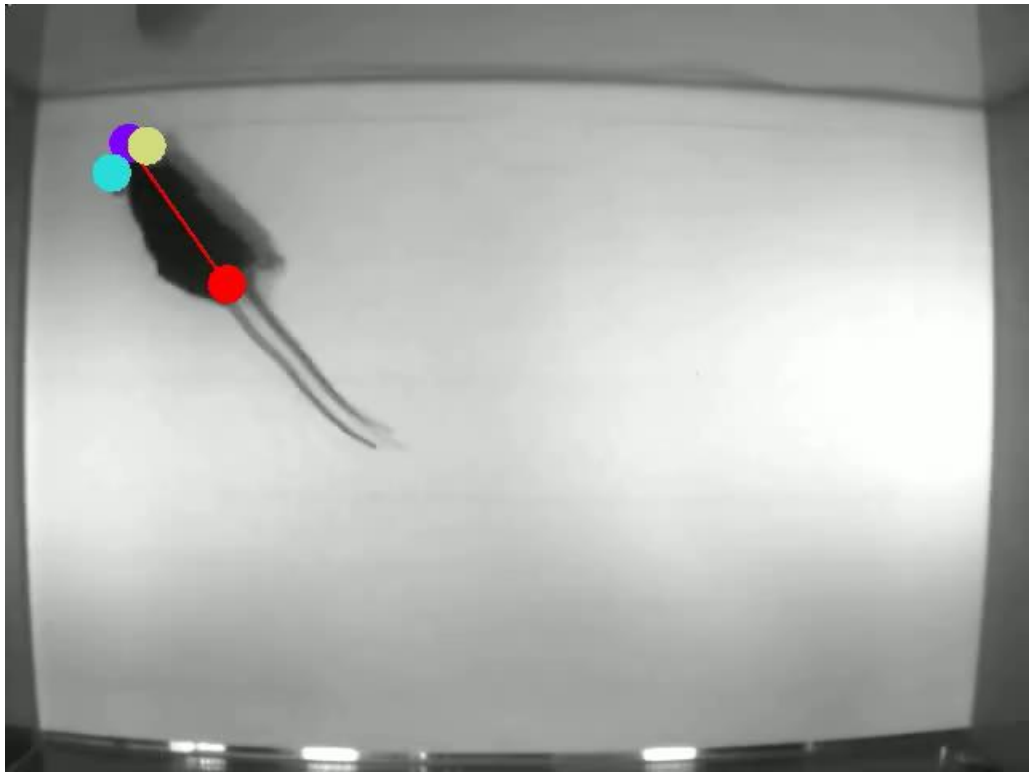
Next…

- After correcting all labels (refine tracklet & extract outliers), you could merge datasets to make a new one
  - Create a new training dataset to re-train the network
- Weights of each bodyparts will be re-initialized from ImageNet
- You can manually change the *init_weights in pose_cfg.yaml* to the snapshot from previous iteration before training
- Analysis using prediction (.csv)?
- Other tools you could use:
  - Deeplabcut 3D
  - YOLO (You Only Look Once)
  - Model Zoo

After outlier extraction and re-train



Iteration 0 (50 epochs, 27 training set images)



Iteration 1 (50 epochs, 115 training set images)