

UC Davis

UC Davis Previously Published Works

Title

Kinematic analysis of sensorimotor behavior during variable ladder rung walking in short-tailed opossums (*Monodelphis domestica*)

Permalink

<https://escholarship.org/uc/item/6cs9w5fh>

Journal

STAR Protocols, 2(2)

ISSN

2666-1667

Authors

Englund, Mackenzie
Faridjoo, Samaan
Iyer, Christopher S
et al.

Publication Date

2021-06-01

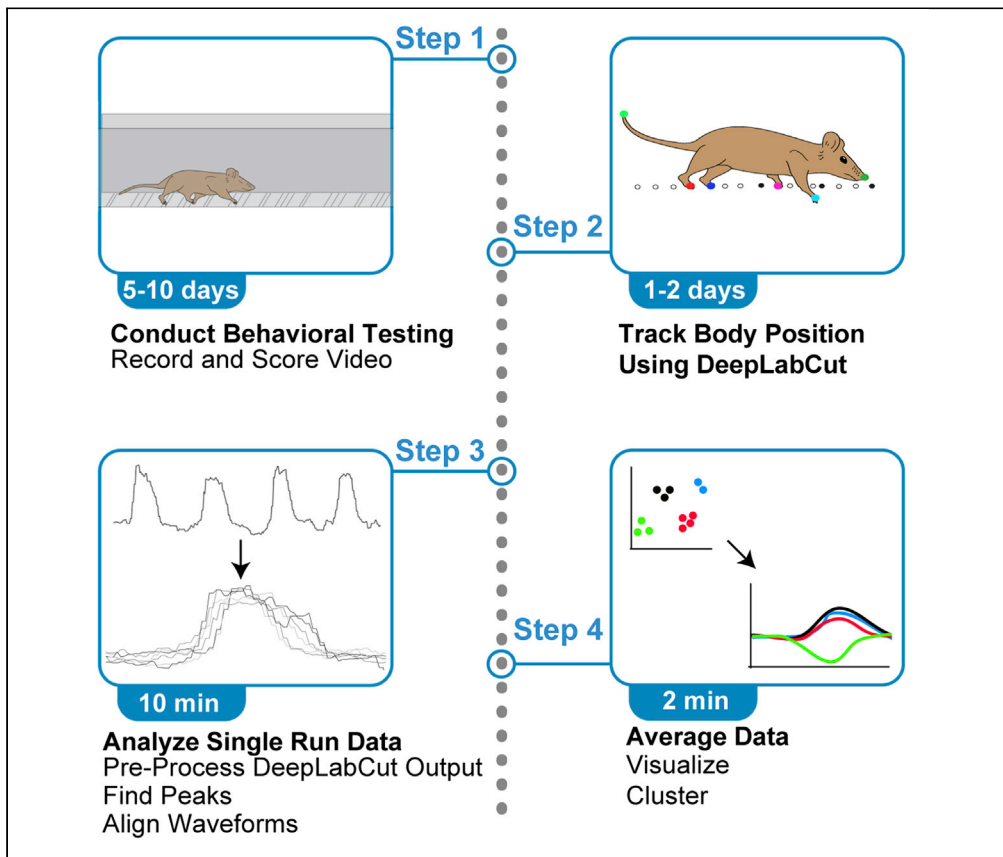
DOI

10.1016/j.xpro.2021.100421

Peer reviewed

Protocol

Kinematic analysis of sensorimotor behavior during variable ladder rung walking in short-tailed opossums (*Monodelphis domestica*)



This protocol presents a workflow for detecting differences in kinematics between experimental conditions. It is tailored for short-tailed opossums but can be applied to any species capable of completing the ladder rung task. There are four phases of this protocol: (1) data collection, (2) pose tracking, (3) analysis of single trials, and (4) cross-condition comparisons. This pipeline implements aspects of machine learning and signal processing, allowing for rapid data analysis that provides insight into how animals perform this task.

Mackenzie Englund,
Samaan Faridjoo,
Christopher S. Iyer,
Leah Krubitzer

menglund@ucdavis.edu

Highlights

Protocol uses a simple sensorimotor task for animal models that requires little training

Provides a pipeline for rapid analysis of pose tracking data

Implements supervised clustering, treating pose data as waveforms

Englund et al., STAR Protocols
2, 100421
June 18, 2021 © 2021 The
Author(s).
<https://doi.org/10.1016/j.xpro.2021.100421>



Protocol

Kinematic analysis of sensorimotor behavior during variable ladder rung walking in short-tailed opossums (*Monodelphis domestica*)Mackenzie Englund,^{1,5,6,*} Samaan Faridjoo,² Christopher S. Iyer,³ and Leah Krubitzer^{1,4}¹A Department of Psychology, University of California, Davis, 135 Young Hall, 1 Shields Avenue, Davis, CA 95616, USA²Department of Molecular and Cellular Biology, University of California, Davis, 149 Briggs Hall, 1 Shields Avenue, Davis, CA 95616, USA³Symbolic Systems Program, Stanford University, 460 Margaret Jacks Hall, 450 Serra Mall, Stanford, CA 94305, USA⁴Center for Neuroscience, University of California, Davis, 1544 Newton Ct., Davis, CA 95618, USA⁵Technical contact⁶Lead contact*Correspondence: menglund@ucdavis.edu
<https://doi.org/10.1016/j.xpro.2021.100421>

SUMMARY

This protocol presents a workflow for detecting differences in kinematics between experimental conditions. It is tailored for short-tailed opossums but can be applied to any species capable of completing the ladder rung task. There are four phases of this protocol: (1) data collection, (2) pose tracking, (3) analysis of single trials, and (4) cross-condition comparisons. This pipeline implements aspects of machine learning and signal processing, allowing for rapid data analysis that provides insight into how animals perform this task.

For complete details on the use and execution of this protocol, please refer to Englund et al. (2020).

BEFORE YOU BEGIN

Construct the ladder rung apparatus and set up camera(s)

⌚ Timing: 3–5 h

1. Obtain clear 3 mm thick Plexiglass and cut to dimensions 1.0 m × 0.2 m (Figure 1A, Metz and Whishaw 2009). These dimensions are ideal for animals 50 g–300 g. However, taller sides can be used for larger animals.
 - a. Complete this process twice, creating the walls of the apparatus.
 - b. Obtain metal rods 3 mm in diameter and 0.15 m in length.
2. Drill holes along each Plexiglass wall at 1 cm spacing, 1 cm from the bottom of each wall (Figure 1B).
 - a. Test that a rung can fit through the first hole before proceeding with drilling the remaining holes.
3. Place metal rods into remaining holes, connecting the two Plexiglass walls by the metal rungs.
 - a. If the apparatus splays at the top, either use two metal rungs to brace this portion or a thin strip of packing tape to hold the apparatus at a consistent width.
4. Place the apparatus on two upside-down clean cages at the start and end of the apparatus. This can be done on a raised surface about 0.5 m from the floor to discourage animals from jumping (Figure 1A).



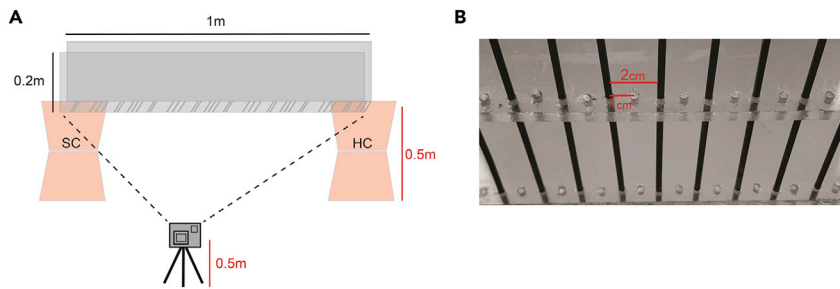


Figure 1. Schematic of the ladder rung apparatus

(A) The start cage (SC) is placed at one end of the apparatus and the animal's home cage (HC) is placed at the other. Using two upside-down cages, the researcher can raise the apparatus from the floor. The camera is placed far enough away from the ladder to capture the entire apparatus in its field-of-view (dotted lines). The camera lens and ladder rungs must be at the same height (vertical red lines) to capture footage at a zero-degree angle, ensuring accurate limb tracking data is collected. Figure reprinted with permission from Englund et al., 2020.

(B) For the standard rung pattern, rungs are placed in every other slot. As the slots are drilled 1 cm apart, this creates a standard rung pattern spacing of 2 cm.

5. Set up camera 0.5–1.0 m from the apparatus, exactly perpendicular and level to the apparatus (Figure 1A).
 - a. Place a scale bar on or in the same plane as the apparatus, in the frame of the camera.

Optional: Multiple cameras can be used to capture the top, bottom, and sides of the ladder rung task.

⚠ **CRITICAL:** The camera must be placed level to the apparatus (Figure 1A). Incorrect placement of the camera will result in data distortion when limb trajectories are analyzed. If the camera cannot be placed directly level, the angle between the lens and the ladder can be used for correction (not recommended).

Alternatives: If Plexiglass is not available, any sturdy clear acrylic is acceptable to use when constructing the walls of the apparatus. Additionally, if steel rungs are not available, acrylic or plastic rungs can be used.

Install DeepLabCut and download Jupyter notebooks

⌚ Timing: 5–7 h

6. Install Python3 (<https://www.python.org/downloads/>).
7. Install Anaconda and Jupyter notebook (<https://jupyter.org/install>).
8. Follow the protocol for the installation of DeepLabCut (<https://github.com/DeepLabCut/DeepLabCut>). While we used the original DeepLabCut for our analyses (Mathis et al., 2018), newer versions of DeepLabCut (2.0 and greater; Nath, Mathis et al., 2019) are compatible with this protocol, so long as the output files contain the X and Y coordinates of body trackers and likelihood columns. If the user wishes to utilize the easy readability of DeepLabCut's h5 file format, we point them to <https://docs.h5py.org/en/stable/quick.html> for a quick start guide on how to read and explore h5 files in python. For examples on how to visualize data and utilize p-cutoff values (other than those discussed in this protocol) refer to: <https://github.com/DeepLabCut/DLCutils>
9. Download all analysis scripts as Jupyter Notebooks from the Github repository (https://github.com/maceng4/Monodelphis_Ladder_Rung).

KEY RESOURCES TABLE

REAGENT or RESOURCE	SOURCE	IDENTIFIER
Deposited data		
Test csv file: 'Animal_20984_Light_Var1.csv'	Englund et al., 2020	https://doi.org/10.1016/j.isci.2020.101527 , https://github.com/maceng4/Monodelphis_Ladder_Rung/blob/master/Animal_20984_Light_Var1.csv
Software and algorithms		
Python 3	Python Software Foundation	https://www.python.org/downloads/
DeepLabCut: markerless pose estimation	Mathis et al., 2018	https://github.com/DeepLabCut/DeepLabCut
Jupyter Notebook	Project Jupyter	https://jupyter.org/
Example notebooks:Pre-Process (Code Notebook).ipynb Single_Trial_Code_Notebook.ipynb Combine_Files.ipynb Combine_Files_Your_Data.ipynb	This paper	https://github.com/maceng4/Monodelphis_Ladder_Rung
Other		
Camera: GoPro Hero 6 and above	GoPro	https://gopro.com/en/us/
3 mm Plexiglass sheet	Sibe-R-Plastics	https://www.amazon.com/Sibe-R-Plastics-Supply-0-118-Acrylic-Plexiglass/dp/B084VW78B9/ref=sr_1_5?dchild=1&keywords=Plexiglass+Sheet+3mm+36%22&qid=1614223785&sr=8-5
3 mm Metal rods for ladder rungs	Uxcell	https://www.amazon.com/uxcell-Lathe-Round-Solid-Length/dp/B0868DLG65
Experimental models: organisms/strains		
<i>Monodelphis domestica</i> (> 120 days old, male & female)	UC Davis breeding colony	N/A

STEP-BY-STEP METHOD DETAILS

Conduct behavioral testing

⌚ Timing: 5–10 days per animal

In this major step, animals are tested on the ladder rung apparatus. Animals should undergo no more than ten trials per day and should be tested over multiple days to address variability. Animals should be habituated to the ladder apparatus prior to testing variable patterns. For at least the first day, all trials should consist of standard (2 cm) rung spacing (Figure 1B). This allows the animal to habituate to the apparatus. It is possible more habituation days are required if the animal does not willingly cross the apparatus on Day 1. You may choose the number of habituation days or use an accuracy criterion score on the 2 cm spacing before beginning testing on variable patterns. Note that the effects of training may alter the effect of studying naturalistic behavior. On variable testing days, use 2–3 trials of standard rung spacing prior to testing variable patterns to re-habituate animals to the task. During test trials, the rungs are spaced at irregular intervals, requiring the animal to make new sensory discriminations to determine the location of future rungs. We suggest using a random pattern generator to determine these patterns. Variable patterns can either be used once, or the same variable pattern be run multiple times to study how animals learn variable patterns. Importantly, you must choose a single study design to use before testing and limit the total number of trials per animal per day to ensure animals are attending to the task and do not get fatigued.

1. To begin a single test trial or habituation trial, place a clean start cage on top of the upside-down start cage and place the animal's home cage on top of the upside-down end cage (Figure 1A). The placement of the home cage at the end of the ladder apparatus encourages ladder crossing without requiring a food reward.

2. Begin recording video for the trial. Make note of the run, test day, and variable or standard pattern.
 - a. Video should be recorded using a linear wide-field lens. We recommend using the GoPro Hero 6 and above. Video should be recorded at 120 frames per second or above (240 fps recommended). [Troubleshooting 1](#)
3. Gently remove the animal from their home cage and place the animal in the start (neutral) cage. [Troubleshooting 2](#)
 - a. We recommend using a transfer cup or box, so that the animal is never directly handled. This reduces stress for the animal, which can impact performance.
 - b. If the animal is too small to climb out of the cage and onto the ladder apparatus, we recommend placing a plastic block in the start cage that the animal can use as a step. Alternatively, the neutral cage can be placed upside down, and the animal can be set on top of the cage.
 - c. After placing the animal in or on the start cage, step away from the apparatus and out of the animal's field of view. It is key that the animal's attention remains on crossing the apparatus.
4. Stop recording after the animal has reached the home cage. Again, a plastic block can be used to assist the animal in stepping off the apparatus.
5. Videos can be scored manually, or by adapting the provided Jupyter notebooks to count instances where limb trackers broke the ladder threshold. For scoring measures see [Metz and Whishaw, 2002, 2009](#).

⚠ **CRITICAL:** Do not place the animal directly on the ladder apparatus or place a hand near the apparatus to encourage the animal to cross. It is critical for performance measures that the animal enters the apparatus on its own.

Track body position using deeplabcut

⌚ **Timing:** 24 – 48 h (labeling + training)

In this major step, the researcher trains DeepLabCut to track the tail, snout, and limbs.

6. Follow the protocol for training videos in DeepLabCut ([Mathis et al., 2018](#)), providing sample videos from multiple trials.
 - a. Label the tail, snout, and limbs during the labeling step. We recommend labeling ~200 frames, and training for ~400,000 iterations. [Troubleshooting 3](#)
7. Run the analyze videos function in DeepLabCut for all trials.
8. Save DeepLabCut output comma-separated value sheets (csv's) for all trials. If h5 files are used in lieu of csv's, you must change the file variable in step 9 to the correct path.

Optional: You can choose to label and track additional parts of the body as well (such as multiple points along the tail). If so, Jupyter notebooks must be amended to include the additional trackers.

Analyze single run data

⌚ **Timing:** 10 min per trial

In this major step, data is processed from raw data output (from DeepLabCut) to kinematic waveforms.

9. Run the pre-process code notebook ([Data S1](#)).
 - a. Enter the name of the DeepLabCut output file in cell #2.
 - b. Looking at the output of cell #2, ensure that the columns of the new data-frame match the original file.

- c. Enter the name you wish to save the pre-processed file as in cell #3
- d. In cells #4–7, test plots are generated, showing all values (cell #4), only values above your chosen p-cutoff (cell #6), and values below the p-cutoff (#7). This visualization shows the location of low p-cutoff values (and possible bad tracking). [Troubleshooting 4](#)
10. Open `Single_Trial_Code_Notebook.ipynb` and follow workflow. [Troubleshooting 5](#) During this step you can also refer to the Single Trial Code Notebook PDF (located in GitHub folder).
11. In cell #1, enter the required trial information, scale (x pixels = 1 mm), and read in a single trial pre-processed csv (csv name chosen during step 9). Run cell #1.
 - a. If you did not place a scale in the video initially, one can be retrieved by measuring the inter-rung distance in the video and comparing this to the real inter-rung distance (1 cm).
 - b. `ladder_position` is equal to the vertical position of the ladder (in pixels) in the analyzed video. If the camera was held constant for each trial, and all videos were cropped similarly, this will be a constant number, else it will vary depending on how videos were cropped.
 - c. `height_of_video` is equal to the height of the entire video in pixels.
12. Run cell #2. This will plot the waveform of the right forelimb over time ([Figure 2A](#)).
 - a. The positions of all other body parts are also recorded during this time. 'Time' can be adjusted by the frame rate at which the videos were captured.
13. In cell #3, peak detection is accomplished. Enter the height threshold and inter-peak distance thresholds chosen for your desired analysis ([Figure 2B](#)).
 - a. Only peaks that are detected in this step will be used for further analysis.
 - b. Run cell #3.
 - c. Observe the output of cell #3. Adjust peak height/distance and re-run cell #3 if output is not as desired.
14. Run cell #4. This will print the location of peaks detected in cell #3.
15. In cell #5, remove waveforms you do not wish to analyze by typing in their peak location (as printed in cell #4).
 - a. Run cell #5 only after you have removed undesirable waveforms.
 - b. The output of cell #5 is a plot of the waveforms chosen for analysis ([Figure 2C](#)).
16. In cell #6, choose a p-cutoff value and run.
 - a. The output of cell #6 is a list of values that fall below the chosen p-cutoff.
17. Cell #7 runs a 1-dimensional interpolation, removing datapoints below the p-cutoff, and creating a new column 'RFLy_disp_interp' which contains interpolated values.
18. Run Cells #8 and #9. The output of these cells shows scatterplots of values before (#8) and after (#9) interpolation. Notice that they are the same for the example notebook, as no values fell below the p-cutoff.
19. Run cell #10.
 - a. The output of cell #10 is a line plot of all forelimb waveforms aligned by peak ([Figure 2D](#).)
20. Run cell #11.
 - a. The output of cell #11 is an average line plot of all waveforms of the right forelimb y-component ([Figure 2E](#)).
 - b. At this stage, any other body part can be plotted and viewed by changing "y=RFLy_disp" to a different body part.
21. In cell #12, adjust the name of the csv to reflect the current trial and animal number you wish to save this current data as. Run cell #12.

Optional: This major step can be completed multiple times if desired, saving output csv's of correct placements or misses, or for data centered around a body part other than the forelimb.

Note: The data for all body parts is now centered on the movement of the right forelimb. If the user desires other centering, they must change `RFLy_disp` to another body part.

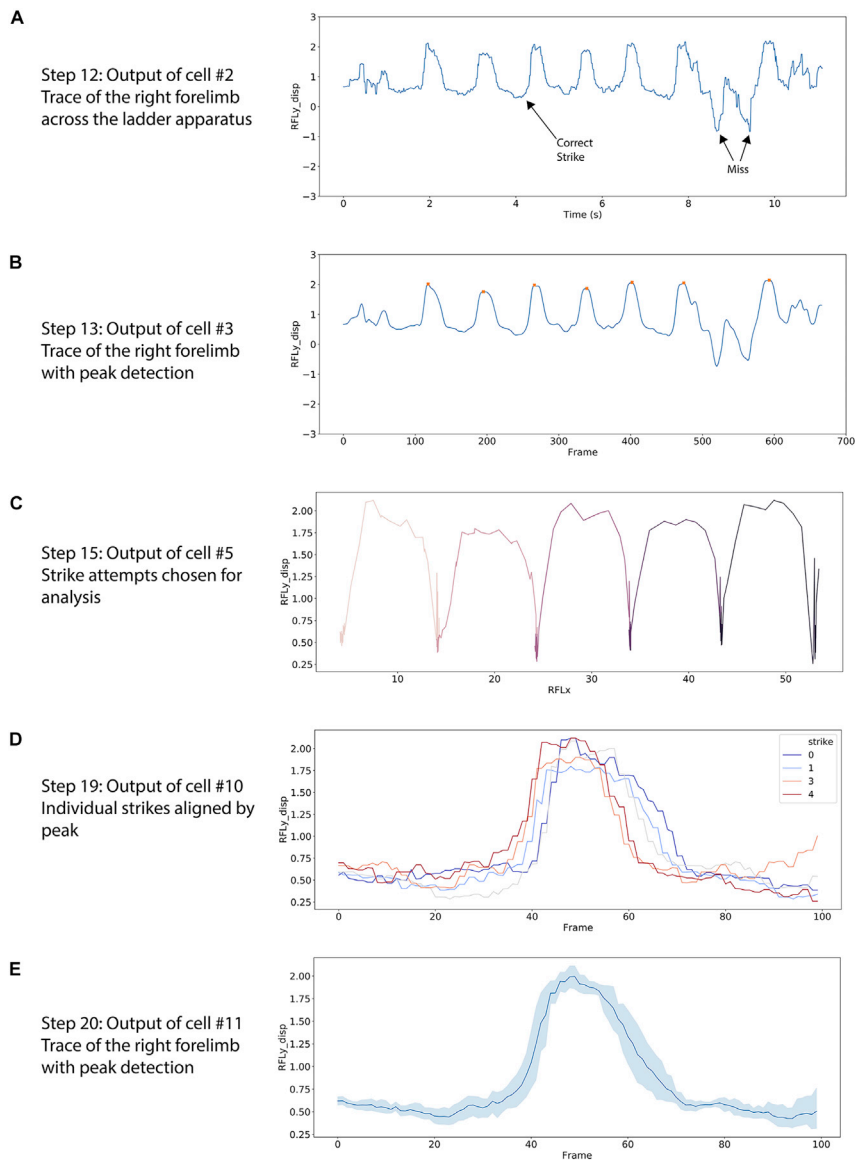


Figure 2. Screenshots of output from Analyze Single Run Data (steps 9–21) Single_Trial_Code_Notebook

(A) Lineplot of the right forelimb Y-component across time. Each waveform represents an individual forelimb ‘strike’ in which the animal lifted its limb from one ladder rung and attempted to place it on another rung.
 (B) Lineplot of peaks chosen for further analysis. Peak detection is decided by user input.
 (C) Lineplot of individual strikes chosen for analysis. Here, the x-axis denotes the actual x-distance traveled.
 (D and E) (D) Individual strikes are then aligned by peak and (E) averaged to create a single correct strike waveform for an individual trial.

Average data and perform clustering

⌚ Timing: 2 min per trial

In this major step, data across trials is aggregated into a single data-frame by loading csv files of all trials. Cross-group comparisons and clustering are also accomplished in this step.

22. Open and run the ‘Combine_Files.ipynb’ or the ‘Combine_Files_Your_Data.ipynb’ file. During this step you can also refer to the Combine Files PDF code notebook ([Data S3](#)).

- a. Enter the name of your csv files in cell #1 with respect to subject. Run cell #1, importing packages and loading csv files.
23. In cell #2, enter a chosen scale value such as shoulder height (or body weight) to normalize your dataset for animals of different sizes.
 - a. It is preferable to choose a scale related to kinematics such as arm length or shoulder height.
 - b. If you chose to use interpolation, change 'RFLy_disp' to 'RFLy_disp_interp' or add a separate column for 'RFLy_disp_scale_interp' (or any other interpolation you accomplished in the previous major steps). It is important for concatenation that the column names match across all animals/trials. Thus, if interpolation is used, it must be used similarly across animals/trials.
 - c. Run cell #2.
24. Concatenate data-frames by inputting imported data-frame names by category/condition in cell #3.
 - a. This step determines what data you will aggregate. Examples are biological sex, correct placements, incorrect placements, etc.
 - b. Run cell #3.
 - c. The output of cell #3 is the number of strikes by condition multiplied by the number of frames per strike.
25. In cell #4, an aggregated output plot is generated using Seaborn (Figure 3A). Select which factors and from what dataset to compare in this visualization.
 - a. Note that the example notebook chooses to plot the average early blind (eb) and sighted (sc) forelimb trajectory Y-component before and after whisker trimming (ebwt, scwt).
 - b. Run cell #4.
26. The next five cells give examples of other body parts or aspects of the collected data that can be analyzed (e.g., X-component trajectories, Hindlimb Trajectory, Snout Trajectory).
 - a. Explore your data by running cells #5 – #9 (Figures 3B and 3C).
27. Begin clustering.
 - a. In cell #10, input the name of the data-frames you would like to concatenate for clustering or use a data-frame you aggregated previously in this notebook.
 - b. Note that we are re-indexing our data. Since each forelimb strike is 100 frames, we are indexing by 100 to re-number the strikes. If you adjusted this number in the previous notebook, it must also be adjusted here.
 - c. Run cell #10.
28. Cell #11 runs a basic principal component analysis with K-Means clustering for cluster visualization purposes. Run cell #11 and observe the distribution of your data.
29. Cell #12 uses the elbow point method to visualize the distortion level of different numbers of clusters. Input the number of desired clusters to test as x in: $K = \text{range}(1, X)$.
 - a. Run cell #12.
 - b. Observe both the printed distortion values and line plot (Figure 3D). For your analysis, choose the cluster that follows the end of exponential decay.
30. Cell #13 calculates the silhouette coefficient for different cluster numbers (Zhou and Gao, 2014).
 - a. Run cell #13.
 - b. Choose the cluster number with the highest silhouette coefficient.
 - c. If there is a mismatch between the elbow point and silhouette coefficient, incorporate both cluster numbers in your analysis or choose one based on a tertiary measure.
31. Cell #14 clusters all waveforms into the chosen number of clusters.
 - a. Enter the number of clusters you would like and run cell #14 (Figure 3E).
32. Save datasets as csv files by entering the name of the chosen dataset and save file name (Cell #15).

EXPECTED OUTCOMES

This protocol produces a single, organized data-frame of body position information from multiple trials, animals, and experimental conditions which can then be analyzed in many different ways

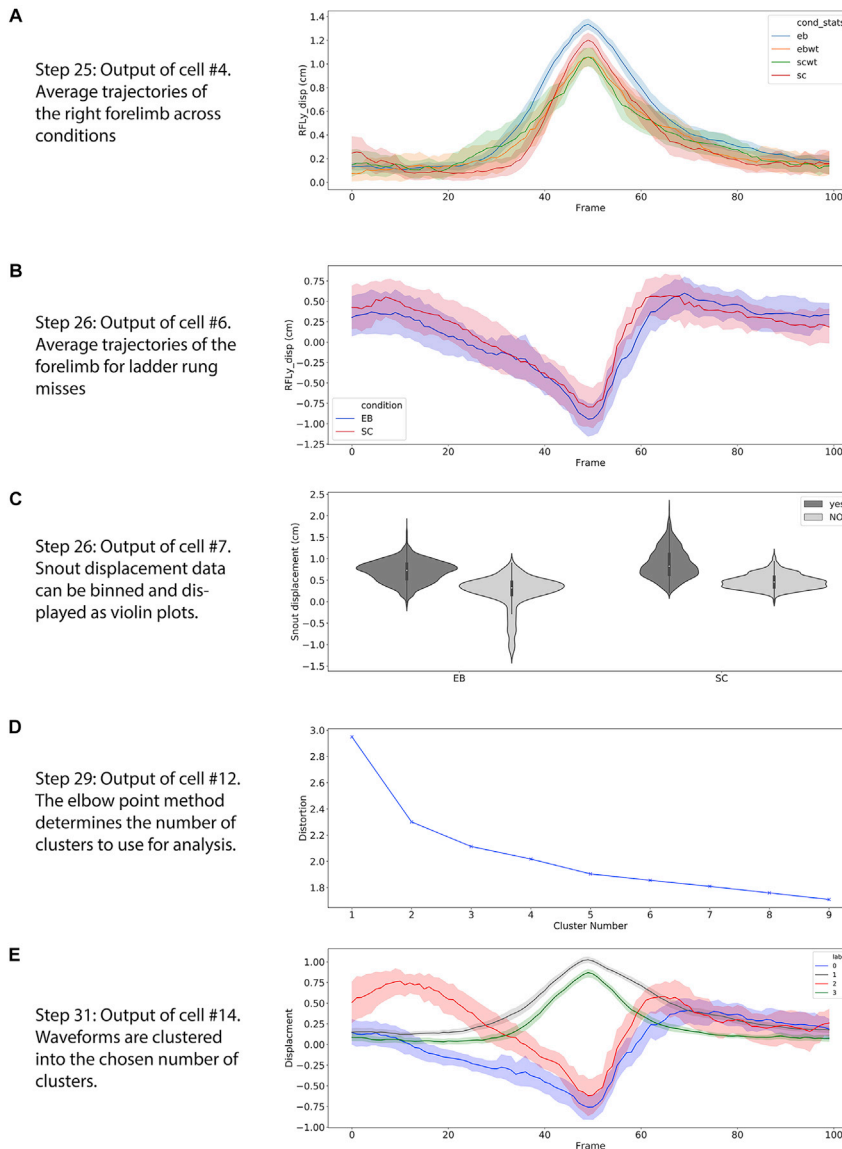


Figure 3. Screenshots of output from Average Data and Perform Clustering (steps 22–32) Combine_Files Jupyter Notebook

(A) Average lineplots of forelimb displacement across experimental condition; eb: early blind, ebwt: early blind whisker-trim, sc: sighted control, scwt: sighted control whisker-trim.

(B) Average lineplots of miss trajectories for early blind (blue) and sighted (red) opossums.

(C) Violin plots of average snout displacement during forelimb motions.

(D) Lineplot showing the level of distortion associated with each cluster number.

(E) Automatically clustered waveforms of forelimb displacement showing two stereotypical correct strikes, and two stereotypical misses.

(step 32). Average limb waveforms should show little variation within-subject across trials due to the constraints of the musculoskeletal system during locomotion and waveform alignment by peak (see Englund et al., 2020, Figures 2D and 3A). Correct forelimb placements of a single subject should cluster into few stereotypical waveform types due to these constraints. However, experimental manipulations (such as those that affect the musculoskeletal, sensory or motor systems) can result in easy-to-observe differences in aspects of limb kinematics such as peak-height, width, and average displacement (Figures 3A and 3C).

QUANTIFICATION AND STATISTICAL ANALYSIS

Data can be binned, averaged, and visualized using violin, box, or bar plots (Figure 3C). Linear models run using Python's Statsmodels package (<https://www.statsmodels.org/stable/index.html>) can detect between-group differences. Additionally, aspects of kinematic waveforms such as peak-height and width at half-maximum can be calculated across subject or experimental condition. One way to accomplish this is to run peak detection via the Single_Trial_Code_Notebook (Data S2) and store the peak values in a separate csv file. During any kinematic analysis it is critical that only data with accurate tracking is included.

LIMITATIONS

Filming the animal from only one angle restricts analysis to two dimensions. Adding a top, bottom, or second side-view camera ameliorates this issue. The resolution and frames per second captured by the camera also contribute to data quality. These aspects of data collection factor into DeepLabCut's ability to track the position of the limbs, snout, and tail. If initial video quality is poor, the number of forelimb strikes with accurate trackers will be lower, decreasing the total number of strikes that can be analyzed in major steps 3 and 4. This is the main bottleneck of the protocol. However, DeepLabCut continues to update software with increasing accuracy. Another limitation of this protocol is that the analysis is centered around forelimb kinematics. All analyses are accomplished with respect to forelimb motion. However, as noted above, the user can alter the included Jupyter notebooks to center analysis on a different aspect of body posture (e.g center on peaks of the snout as it dips down to touch a future rung).

TROUBLESHOOTING

Problem 1

Using a wide-field lens can cause distortions in data acquisition (step 2).

Potential solution

Using a wide-field lens can lead to distortions in data, even if the camera is equipped with a linear processor to flatten the image. To ensure that your wide field lens is not causing distortion, place a grid 1 cm by 1 cm perpendicular to the camera at a distance equal to that of the ladder apparatus. By filming the equally spaced grid and measuring the inter-grid distance from the recorded film (in Photoshop or other program), one can determine the extent of distortion caused by the wide-angle lens. If there is distortion, it is recommended that a different camera be used or that the distortion be fixed using image processing tools prior to labeling data in DeepLabCut.

Problem 2

The animal does not cross the ladder apparatus correctly (step 3).

Potential solution

There are many reasons why the animal may not cross the apparatus. Some animals freeze at the beginning of the apparatus and do not cross at all. Others may move part way and freeze/groom or attempt to turn around. To ameliorate these issues, ensure that the width of the apparatus is wide enough for the animal to freely cross, but narrow enough to discourage the animal from turning around. It is also possible that more habituation trials are required for an animal to learn it must cross the apparatus. If the animal still does not cross the apparatus after multiple days of habituation, the researcher can decide to remove this animal from the testing pool. Researchers can also place treats in the home cage to encourage animals to cross. However, if this is done for one animal, it should be done for all animals. The number of habituation trials/day should be recorded for each animal to test against learning effects.

Problem 3

Inaccurate tracking of the limbs with DeepLabCut (step 6a).

Potential solution

Label more frames and use additional videos from multiple animals/trials/conditions to train the neural network. You can also use a higher resolution camera or pre-process videos by reducing their speed. If you choose to do so, you must remember to adjust the frame rate in your final calculation for 1 frame = x seconds. A third solution is to place the camera closer to the apparatus to increase resolution. If you do this, it is recommended to use a remote start/stop for the camera so that the animal is not disturbed during the task.

Problem 4

Visualization plots produced by the pre-process data notebook are not accurate (step 9d).

Potential solution

Due to formatting of the output file from DeepLabCut, it is possible that the csv file is not read correctly by pandas in python. We have ameliorated this issue by saving the csv file within python and reloading it before plotting (step 9c). If errors still occur, manually open and save the csv, ensuring that it is in the proper file format.

Problem 5

Missing necessary Python packages (step 10).

Potential solution

In Jupyter Notebook, use:

```
!pip install package_name
```

This will install the missing Python package.

RESOURCE AVAILABILITY**Lead contact**

Further information and requests for resources should be directed to and will be fulfilled by the lead contact, Mackenzie Englund (menglund@ucdavis.edu).

Materials availability

There are no unique materials.

Data and code availability

Example datasets and code are available at https://github.com/maceng4/Monodelphis_Ladder_Rung. Code from the example Jupyter Notebook “Combine_Files.ipynb” uses data from Englund et al., 2020.

SUPPLEMENTAL INFORMATION

Supplemental information can be found online at <https://doi.org/10.1016/j.xpro.2021.100421>.

ACKNOWLEDGMENTS

We greatly appreciate the immense time and effort put forth by the creators of DeepLabCut to bring such a powerful research tool to the scientific community. DeepLabCut will continue to be the inspiration for many publications over the coming years. This research was supported by the McDonnell Foundation (Grant 220020516 to L.K.) and the National Institute of Neurological Disorders and Stroke (Grant 1 F31 NS115242-01 to M.E).

AUTHOR CONTRIBUTIONS

Conceptualization, M.E. and L.K.; methodology, M.E.; investigation, M.E., S.F., C.S.I., and L.K.; writing and editing, M.E. and L.K.; funding acquisition, L.K.; supervision, L.K.

DECLARATION OF INTERESTS

The authors declare no competing interests.

REFERENCES

Englund, M., Faridjoo, S., Iyer, C.S., and Krubitzer, L. (2020). Available sensory input determines motor performance and strategy in early blind and sighted short-tailed opossums. *iScience* 23, 101527.

Mathis, A., Mamidanna, P., Cury, K.M., Abe, T., Murthy, V.N., Mathis, M.W., and Bethge, M. (2018). DeepLabCut: markerless pose estimation of user-defined body parts with deep learning. *Nat. Neurosci.* 21, 1281–1289.

Metz, G.A., and Whishaw, I.Q. (2009). The ladder rung walking task: a scoring system and its practical application. *J. Vis. Exp.* e1204.

Metz, G.A., and Whishaw, I.Q. (2002). Cortical and subcortical lesions impair skilled walking in the ladder rung walking test: a new task to evaluate fore-and hindlimb stepping, placing, and co-ordination. *J. Neurosci. Methods* 115, 169–179.

Nath, T., Mathis, A., Chen, A.C., Patel, A., Bethge, M., and Mathis, M.W. (2019). Using DeepLabCut for 3D markerless pose estimation across species and behaviors. *Nat. Protoc.* 14, 2152–2176.

Zhou, H.B., and Gao, J.T. (2014). Automatic method for determining cluster number based on silhouette coefficient. *Advanced Materials Research*, vol. 951 (Trans Tech Publications Ltd.), pp. 227–230.