Extended discussion

Data complexity can be resolved by active learning

In studies investigating the behavior of animals in social contexts, approaches that have been trying to disentangle the high data complexity of multiple interacting individuals (unsupervised), oftentimes struggle to produce results that align with human consensus. This is partly because many existing solutions are predisposed to focus on single animal behaviors [1–6] as the within-animal spatiotemporal dynamics are generally more conserved and less complex than intra-animal dynamics.
On the other hand, most supervised solutions require a sizable ground truth data set to reliably reproduce human expectations. One reason for this is that most collected data sets are inherently unbalanced. They consist of high-frequency behaviors (e.g., investigation) accompanied by low-frequency behaviors (e.g., attack). Behaviors that are underrepresented in the data set are then underrepresented in training and will result in poor performance levels. In both cases, therefore, the complexity of the expected outcome and the composition of the data inhibit the reproduction of the human reference frame. A-SOiD solves this challenge by employing an active learning regime. By explicitly refining low-confidence predictions, the algorithm focuses on unclear decision boundaries between classes and continuously learns to reproduce the expert's definitions with high consistency (Fig. 1h-i bottom). By starting with an absolute minimum number of annotations (yet still maintaining the size differences), we let A-SOiD determine which features are to be annotated. This approach effectively sparsifies data means, while focusing on the outliers. Over iterations, the training data is therefore auto-balanced across classes. This capacity eradicates the need for additional data augmentation or filtering steps to balance the underlying data (see Suppl. Table 2). Additionally, this approach reduces necessary ground truth annotation for uncommon, low-frequency behaviors (Fig. 1h-i bottom) and the cost required to implement such a solution in a dynamic analysis pipeline.
However, a general challenge of behavioral analysis is the temporal scale at which behavior forms and changes in observed animals. Specifically, behavioral expressions can be subject to changes through previous experiences and different contexts. The possible range of observable behavior is further limited by the context and duration of each recording session. Therefore, it is unrealistic for most projects to collect an extensive ground truth encompassing the entire behavioral repertoire. Insufficient ground truth data may lead to inaccurate predictions at different times in the animal's life or across a wide range of behavioral assays. Here, an optimal solution would be able to continuously build upon the observed repertoire and retrain the algorithm with novel variations. Unfortunately, many approaches, especially unsupervised, would require a complete restart and realignment to human reference frames, increasing the cost to employ such methods drastically. In contrast, iterative approaches such as A-SOiD have the potential to be continuously updated

with new observations using the active learning scheme.

Utility of creating a equally emphasized network

Further explanation of a behavioral classifier can provide insight into difference in predictive performance, which subsequently can be used to understand behavioral differences between experimental conditions. A classifier can be predict at a similar overall performance with a wide variety of training regime. It is then required to dissect the model independently. Recent work on explainable AI [7, 8] has allowed machine learning engineers to rank features given the labels. In addition to better performing small classes, the benefit of having a balanced representation is for explanatory approaches like SHAP to independently identify behavioral differences without being subject to overprioritizing one class over the other. In our results, we have demonstrated a albeit similar performing models, our iterative learning schemes provided a precise separation between feature value and it's impact on model (SHAP value). If we used an imbalanced dataset to fit a classifier, the feature value impact on model output will be intermixed since all the original model had to do was overclassifying this large class.
Strategies to increase transparency of behavioral classification To quickly asses the differences between discovered patterns (e.g., behavioral sub-types), we previously employed motion energy (see Methods) [2, 9]. Motion energy is an intuitive and informative way to generate visual summary of the action within a found cluster (Fig. 3b-c). In our hands, we utilized motion energy images to quickly differentiate sub-types of anogenital investigation (Fig. 3). Note, that also further analysis can be done by comparing variability within and across groups providing a valuable statistic for cluster quality [2]. Another approach is using SHAP-based reporting of the underlying feature importance which can help to share and compare conserved patterns across studies (for a review see [10]). The feature value impact and ranking not only describe the refinement process but also provide an insight into the intuitive human definition once the classifier reaches high performance (Fig. 2b and Fig. 4b). Here, SHAP-based reporting can serve as a looking glass into the underlying intuitive human reference frames by translating reproductions of human annotations into transparent, operationalized definitions. In this study, we employed SHAP-analysis to investigate the learning process across multiple iterations during active learning and could identify that specific sets of features accounted for the increased performance of our classifiers (compare TOP-5 features across iterations in Fig. 2c and Fig. 4c). Moreover, the feature importance did not change after the classifier's performance plateaued, indicating that this ranking was able to reproduce human annotations with high reliability. Consequently, the final set can be used to explain and compare the intuitive reference frame in the context of extracted features.

Active learning framework for user-defined data

To allow the integration of our developed approach into already existing behavior analysis pipelines, we created a streamlit-based application that integrates the core features of A-SOiD into a user-friendly, no-coding required GUI solution that can be downloaded and used on custom data. For this we developed a multi-step pipeline that guides users, independent of their previous machine learning abilities through the process of generating a well-trained, semi-supervised classifier for their own use-case. While the underlying code is based on the open-source language python and available on GitHub (https://github.com/YttriLab/A-SOID), the use of A-SOiD's core feature (active learning) to reduce the amount of necessary ground truth data considerably can be directly used by installing the app on a local computer. In general, users are required to provide a small labeled data set (ground truth) with behavioral categories of their choice using one of the many available labeling tools (e.g. BORIS; [11]) or import their previous supervised machine learning data sets. Following the upload of data (see Supp. Fig. S1a), a A-SOiD project is created, including several parameters that further enable users to select individual animals (in social data) and exclude body parts from the feature extraction. Based on the configuration, the feature extraction (Supp. Fig. S1b top) can be further customized by defining a "bout length" referring to the temporal resolution in which single motifs are expected to appear (e.g. the shortest duration a definable component of the designated behavior is expected to last; see also Fig. 1). The extracted features are then used in combination with the labeled ground truth to train a baseline model. Here, an initial evaluation will give users insight into the performance on their base data set (Supp. Fig. S1b bottom). Note, that different splits are used to allow for a more thorough analysis (see Methods for further details).
The baseline classification will then be used as a basis for the first active learning iteration, where users are prompted by the app to view and refine bouts that were classified with low confidence by the baseline model (Supp. Fig. S1c left). Bouts are visualized by showing an animated sequence of the provided pose information and designated body parts and the viewer can be utilized to show the bouts in several different options, including increased/decreased speed, reverse view and frame-by-frame view. After submission of a refined bout, a new bout is shown at its place and the refinement continues for a user-defined amount of low confidence bouts. Following refinement, a new iteration of the model is trained and its performance can be viewed (Supp. Fig. S1c right) in comparison to previous iterations. This process is then repeated until the user is satisfied with the model's performance or until a plateau has been reached (see Fig. 2).
Finally, users can upload and classify new data using the app and the previously trained classifier (Supp. Fig. S1d). To gain further insight into the results of the classification, the app offers a reporting tab that allows users to view and export a selected set of analysis reports, including the common ethogram and statistics (Supp. Fig. S1d).

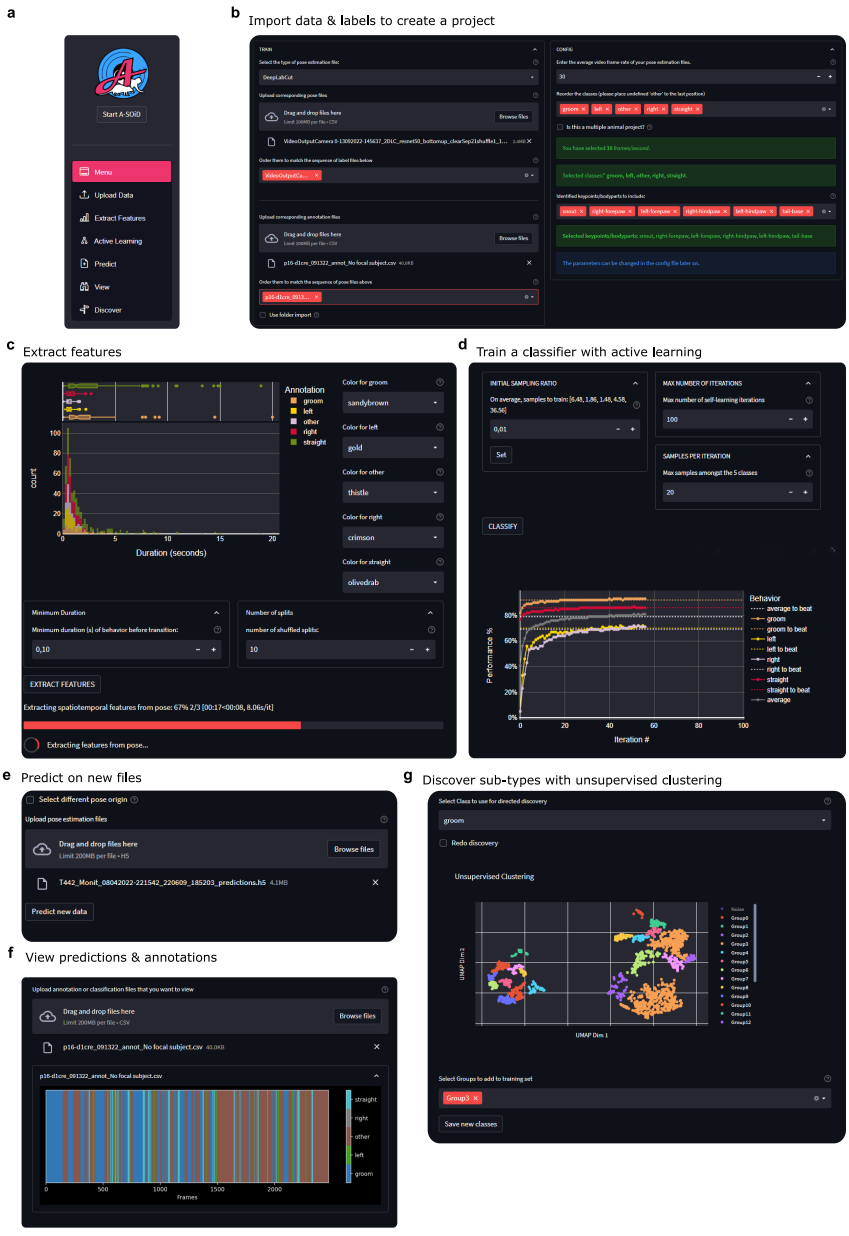

a
Start A-SOID
Menu
Upload Data
Extract Features
Active Learning
Predict
View
Discover
b
Import data & labels to create a project
c
Extract features
Annotation
groom
left
other
right
straight
Color for groom
sandybrown
Color for left
gold
Color for other
thistle
Color for right
crimson
Color for straight
olivedrab
count
Duration (seconds)
Minimum Duration
Minimum duration (s) of behavior before transition:
0,10
Number of splits
number of shuffled splits:
10
EXTRACT FEATURES
Extracting spatiotemporal features from pose: 67% 2/3 [00:17<00:08, 8.06s/it]
Extracting features from pose...
d
Train a classifier with active learning
INITIAL SAMPLING RATIO
0,01
Set
MAX NUMBER OF ITERATIONS
100
SAMPLES PER ITERATION
20
CLASSIFY
Performance %
Iteration #
Behavior
average to beat
groom
groom to beat
left
left to beat
right
right to beat
straight
straight to beat
average
e
Predict on new files
Select different pose origin
Upload pose estimation files
Drag and drop files here
Browse files
T442_Monit_08042022-221542_220609_185203_predictions.h5
Predict new data
f
View predictions & annotations
Upload annotation or classification files that you want to view
p16-d1cre_091322_annot_No focal subject.csv
Frames
g
Discover sub-types with unsupervised clustering
Select Class to use for directed discovery
groom
Redo discovery
Unsupervised Clustering
UMAP Dim 1
UMAP Dim 2
Select Groups to add to training set
Group3
Save new classes

Figure S1: Active learning framework on user-defined data. a) The A-SOiD GUI offers a step-by-step navigation to run A-SOiD on your own data. b) First users can select the origin/type of their pose estimation data (SLEAP, DLC or CALMS21) and uploads their data set including a previously labeled ground truth. Using a user-defined working directory and prefix, previous sessions can be continued at later stages by uploading the corresponding config file (not shown). Right, the user is able to enter basic parameters (framerate, resolution), behavioral categories of interest that are contained in the ground truth data set as well as sub-select individuals/animals and key points/body parts as a basis for feature extraction. c) After input of a temporal reference frame (aka. bout length) for feature extraction using a histogram as shown in Fig. 1 (top), features are extracted and a number of splits is provided to evaluate later classification training. d) In the active learning segment, a classifier is trained as described previously by iterative addition of low-confidence predictions. Here, refinement is directly taken from the remaining ground truth. During each iteration the model's performance is evaluated on a held out test data for multiple splits. This process can be viewed live for each iteration. e) Finally, once the training is complete, users can use the app to upload new unlabeled data and use the previously trained model for classification. f) After classification, the app allows users to go through the results and view a brief report. g) Users are also able to discover conserved patterns in their ground truth data, by selectively clustering annotation classes with unsupervised classification. Sub-types that are of interest can then be exported to create a new training set and be used to train a classifier.

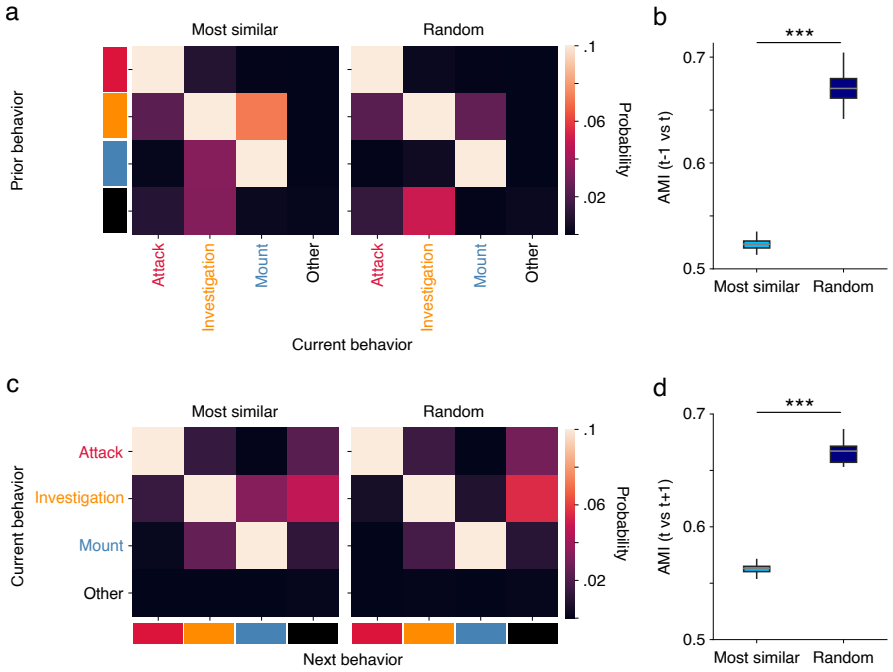


Figure S2: More candidates for active learning refinement appear at behavior transitions. a) Transition matrix for the frame annotation that happens before refinement candidates throughout A-SOiD (left), when compared to a random frame selection of these behaviors (right). b) Adjusted mutual information score as a metric to quantify similarity between prior frame (t-1) and refinement candidate/random selection (t). c) Transition matrix for the frame annotation that happens after refinement candidates throughout A-SOiD (left), in contrast to a random frame selection of these behaviors (right). d) Adjusted mutual information score as a metric to quantify similarity between next frame (t+1) and refinement candidate/random selection. *** = $p < 0.001$.

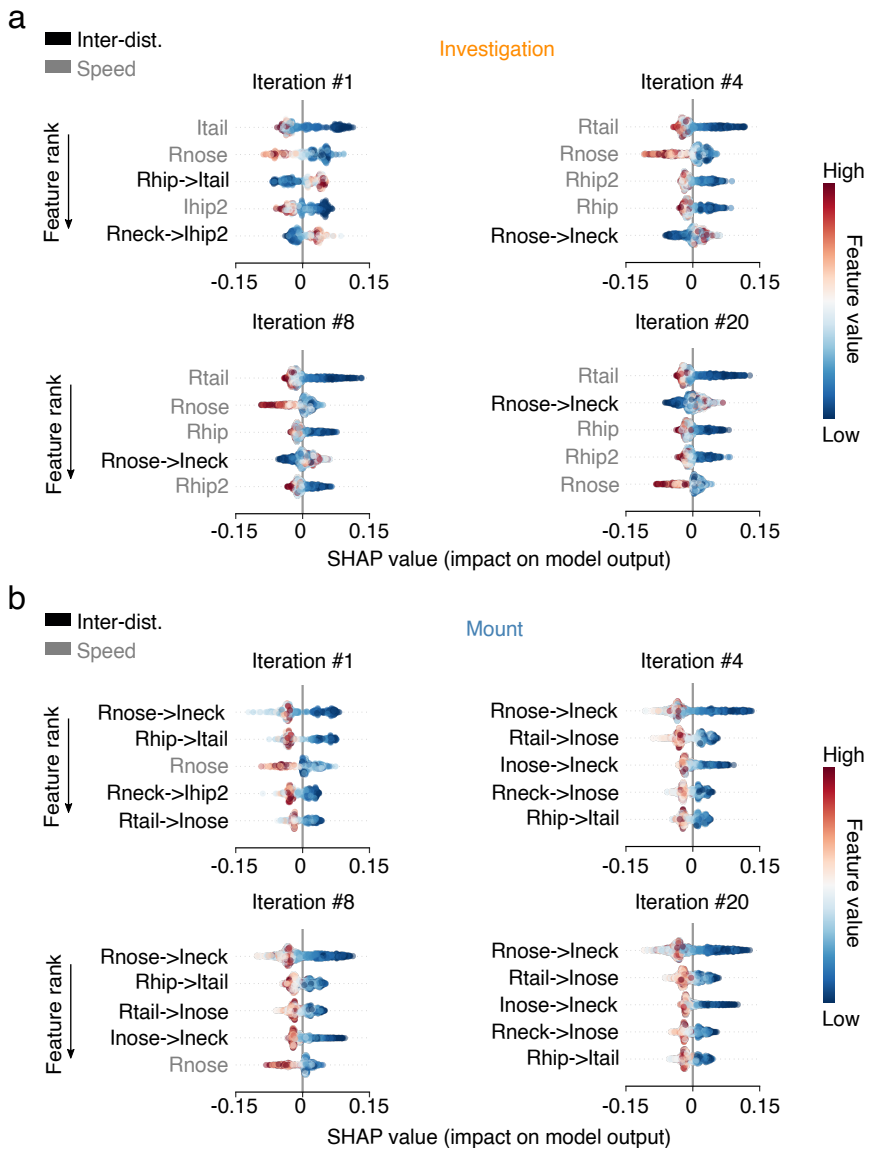
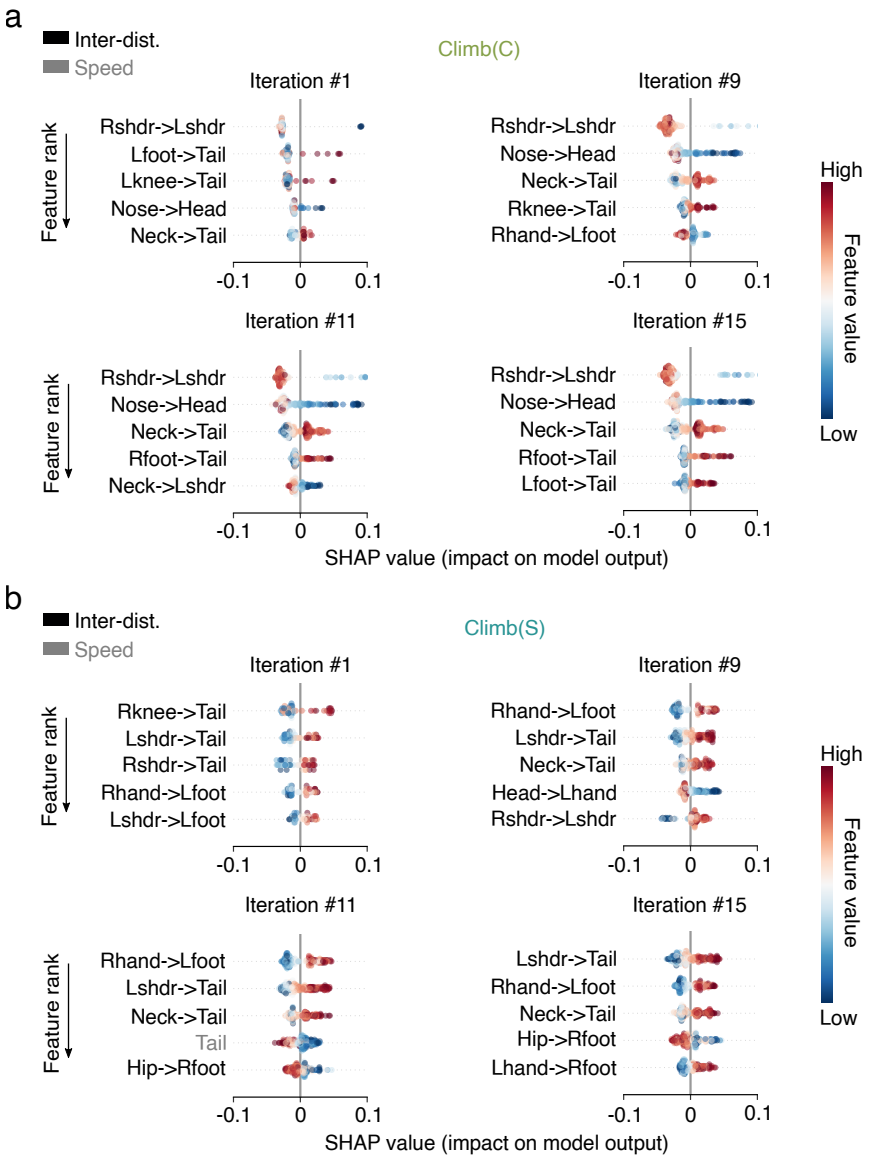


Figure S3: SHAP analysis to replicate model predictions for the CALMS21 social mice data set. a) Ranked order of the top five features (descending order) across iterations for the "investigation" class, including individual feature impact (x-axis) separated by relative feature value (High: red, Low: blue). b) Ranked order of the top five features for the "mount" class. Features (inter-animal distance: black, speed: gray) are denoted by their corresponding animal (R: resident, I: intruder) and body part (e.g., nose).


a
Inter-dist.
Speed
Climb(C)
Iteration #1
Rshdr->Lshdr
Lfoot->Tail
Lknee->Tail
Nose->Head
Neck->Tail
Iteration #9
Rshdr->Lshdr
Nose->Head
Neck->Tail
Rknee->Tail
Rhand->Lfoot
Iteration #11
Rshdr->Lshdr
Nose->Head
Neck->Tail
Rfoot->Tail
Neck->Lshdr
Iteration #15
Rshdr->Lshdr
Nose->Head
Neck->Tail
Rfoot->Tail
Lfoot->Tail
Feature rank
-0.1
0
0.1
High
Feature value
Low
SHAP value (impact on model output)
b
Inter-dist.
Speed
Climb(S)
Iteration #1
Rknee->Tail
Lshdr->Tail
Rshdr->Tail
Rhand->Lfoot
Lshdr->Lfoot
Iteration #9
Rhand->Lfoot
Lshdr->Tail
Neck->Tail
Head->Lhand
Rshdr->Lshdr
Iteration #11
Rhand->Lfoot
Lshdr->Tail
Neck->Tail
Tail
Hip->Rfoot
Iteration #15
Lshdr->Tail
Rhand->Lfoot
Neck->Tail
Hip->Rfoot
Lhand->Rfoot
Feature rank
-0.1
0
0.1
High
Feature value
Low
SHAP value (impact on model output)

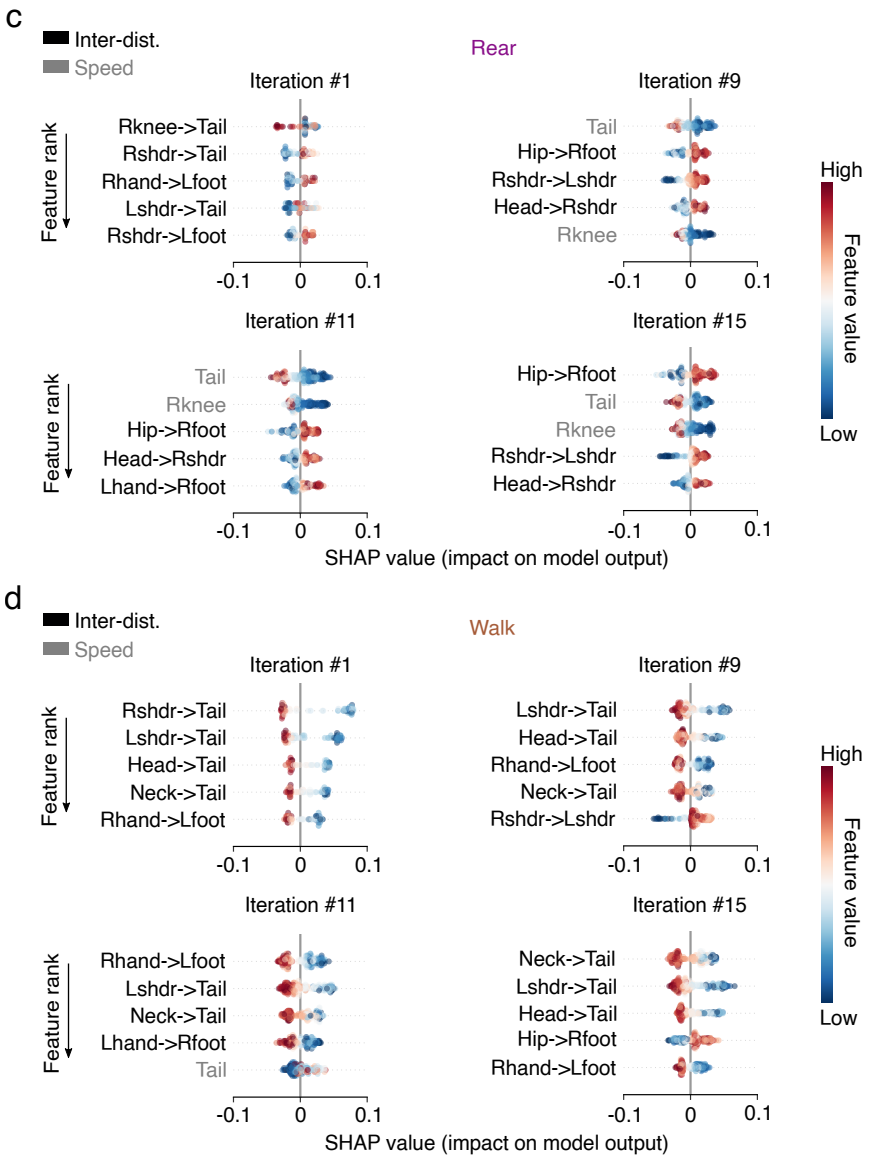


Figure S4: SHAP analysis to replicate model predictions for the single housed monkey data set. a) Ranked order of the top five features (descending order) across iterations for the "ceiling climb" class, including individual feature impact (x-axis) separated by relative feature value (High: red, Low: blue). b) Ranked order of the top five features for the "sidewall climb" class. c) Ranked order of the top five features for the "rear" class, including individual feature impact (x-axis) separated by relative feature value (High: red, Low: blue). d) Ranked order of the top five features for the "walk" class. Features (inter-animal distance: black, speed: gray) are denoted by body part (e.g., tail).

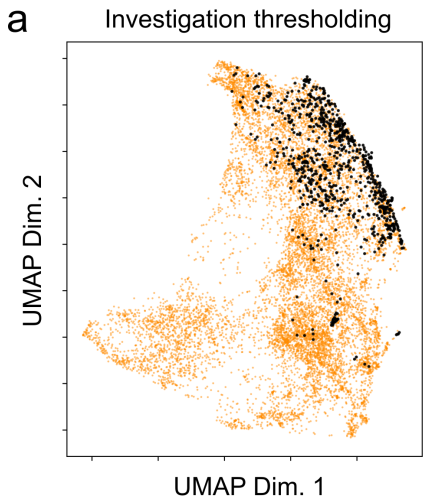


Figure S5: Unsupervised embedding of the investigation class can be explored using heuristic approach a) Annotation of all data points within the investigation class in which the distance between the resident's snout and the intruder's tail base was lower than a manually set threshold (15 pixels). A comparison revealed an extensive overlap between the unsupervised clusters of sub-class 2 and sub-class 5 overlap with the top-down, manually selected feature space (compare with Fig. 3a).

Table 1: Detailed description of the extracted features

Type	Feature	Body part(s)	Description
Intra-animal	Distance	all	*distance in pixels between two body parts of the same animal, e.g., resident snout and resident tail base*
	Angular change	all	*angular change within a* ***bout*** *of two body parts of the same animal, e.g., snout and tail base*
	Speed	all	*speed in pixels/second of an animal measured within a* ***bout***
Inter-animal	Distance	all	*distance in pixels between two body parts of different animals, e.g., resident snout and intruder snout*
	Angular change	all	*angular change within a* ***bout*** *of two body parts of different animals, e.g., resident snout and intruder tail base*

Table 2: Average number of labels during active learning iterations for CalMS21 in Fig. 2b. Averaged across evaluations (20-fold). Total number of available labels (n = 15866); per class: Attack = 1188, Investigation = 12300, Mount = 2378.

iteration	attack	investigation	mount	total
1	91.4	165.2	97.55	354.15
2	112.85	316.5	127.65	557.
3	180.35	395.65	181.	757.
4	230.85	470.4	255.75	957.
5	290.55	566.7	299.75	1157.
6	353.15	646.75	357.1	1357.
7	410.45	727.	400.55	1538.
8	462.25	734.65	435.1	1632.
9	490.75	740.85	457.4	1689.
10	509.85	770.1	481.05	1761.
11	525.45	798.	504.55	1828.
12	537.3	813.7	515.	1866.
13	544.65	824.35	524.	1893.
14	552.75	832.1	527.15	1912.
15	557.9	840.9	531.2	1930.
16	561.1	849.75	532.15	1943.
17	567.25	856.3	534.45	1958.
18	569.35	860.	537.65	1967.
19	570.5	874.6	539.9	1985.
20	576.7	875.	541.3	1993.

Table 3: Average number of labels during active learning iterations for single housed monkey in Fig. 4b. b) per class. Averaged across evaluations (20-fold). Total number of available labels (n = 1181); per class: Climb (C) = 64, Climb (S) = 177, Jump = 50, Rear = 214, Walk = 676.

iteration	Climb (C)	Climb (S)	Jump	Rear	Walk	total
1	5.9	13.85	3.05	14.55	33.65	71
2	9.05	19	4.45	18.35	35.15	86
3	11.7	24.1	6.45	21.45	37.3	101
4	14.45	27.8	8.8	24.55	40.4	116
5	15.9	31.7	11.4	28.8	43.2	131
6	17.5	35.55	14.35	31.95	46.65	146
7	19.4	38.6	16.5	35.1	51.4	161
8	20.75	41.35	19.3	39.35	55.25	176
9	22.2	45.45	22.15	41.6	59.6	191
10	24	48.2	25.65	44.45	63.7	206
11	25.35	50.55	29.15	47.7	68.25	221
12	26.15	53.25	32.15	50.3	73.8	235.65
13	27	55.1	35.05	52.15	79.55	248.85
14	27.35	57.2	36.3	54.35	83.3	258.5
15	27.8	58.55	37.1	55.6	85.35	264.4

Supplementary Movie 1: Video examples of two sub-classes segmented from investigation that reflect anogenital investigation. On the left, one mouse directly approaches the anogenital area of another mouse, irrespective of incoming angle ("anogenital approach"). On the right, one mouse investigates the anogential area of another mouse while already being in close proximity to begin with ("anogenital investigation").

References

1. Luxem, K. *et al.* Identifying Behavioral Structure from Deep Variational Embeddings of Animal Motion. *bioRxiv,* 2020.05.14.095430. `https://www.biorxiv.org/content/10.1101/2020.05.14.095430v3%20https://www.biorxiv.org/content/10.1101/2020.05.14.095430v3.abstract` (Jan. 2022).
2. Hsu, A. I. & Yttri, E. A. B-SOiD, an open-source unsupervised algorithm for identification and fast prediction of behaviors. *Nature Communications 2021 12:1* **12,** 1–13. ISSN: 2041-1723. `https://www.nature.com/articles/s41467-021-25420-x` (Aug. 2021).

3. Graving, J. M. *et al.* DeepPoseKit, a software toolkit for fast and robust animal pose estimation using deep learning. *eLife* **8.** ISSN: 2050084X. /pmc/articles/PMC6897514/%20/pmc/articles/PMC6897514/?report=abstract%20https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6897514/ (Oct. 2019).

4. Berman, G. J., Choi, D. M., Bialek, W. & Shaevitz, J. W. Mapping the stereotyped behaviour of freely moving fruit flies. *Journal of the Royal Society, Interface* **11.** ISSN: 1742-5662. https://pubmed.ncbi.nlm.nih.gov/25142523/ (Oct. 2014).

5. Marshall, J. D. *et al.* Continuous Whole-Body 3D Kinematic Recordings across the Rodent Behavioral Repertoire. *Neuron* **109,** 420–437. ISSN: 0896-6273 (Feb. 2021).

6. Wiltschko, A. B. *et al.* Mapping Sub-Second Structure in Mouse Behavior. *Neuron* **88,** 1121–1135. ISSN: 1097-4199. https://pubmed.ncbi.nlm.nih.gov/26687221/ (2015).

7. Lundberg, S. M., Allen, P. G. & Lee, S.-I. A Unified Approach to Interpreting Model Predictions. *Advances in Neural Information Processing Systems* **30.** https://github.com/slundberg/shap (2017).

8. Lundberg, S. M. *et al.* From local explanations to global understanding with explainable AI for trees. *Nature Machine Intelligence 2020 2:1* **2,** 56–67. ISSN: 2522-5839. https://www.nature.com/articles/s42256-019-0138-9 (Jan. 2020).

9. Stringer, C. *et al.* Spontaneous behaviors drive multidimensional, brainwide activity. *Science* **364.** ISSN: 10959203. https://www.science.org/doi/10.1126/science.aav7893 (Apr. 2019).

10. Goodwin, N. L., Nilsson, S. R., Choong, J. J. & Golden, S. A. Toward the explainability, transparency, and universality of machine learning for behavioral classification in neuroscience. *Current Opinion in Neurobiology* **73,** 102544. ISSN: 0959-4388 (Apr. 2022).

11. Friard, O. & Gamba, M. BORIS: a free, versatile open-source event-logging software for video/audio coding and live observations. *Methods in Ecology and Evolution* **7,** 1325–1330. ISSN: 2041-210X. https://onlinelibrary.wiley.com/doi/full/10.1111/2041-210X.12584%20https://onlinelibrary.wiley.com/doi/abs/10.1111/2041-210X.12584%20https://besjournals.onlinelibrary.wiley.com/doi/10.1111/2041-210X.12584 (Nov. 2016).