JAX Animal Behavior System (JABS): A video-based phenotyping platform for the laboratory mouse

Glen Beane[1], Brian Q. Geuther[1], Thomas J. Sproule[1], Anshul Choudhary[1], Jarek Trapszo[1], Leinani Hession[1], Vivek Kohar[1], and Vivek Kumar[1]

[1]The Jackson Laboratory, 600 Main Street, Bar Harbor ME 04609
⋆Correspondence: vivek.kumar@jax.org

February 9, 2023

Abstract

Automated detection of complex animal behavior remains a challenge in neuroscience. Developments in computer-vision have greatly advanced automated behavior detection and allow high-throughput pre-clinical studies. An integrated hardware and software solution is necessary to facilitate the adoption of these advances in the field of behavioral neurogenetics, particularly for non-computational labs. We have published a series of papers using an open field arena to annotate complex behaviors such as grooming, posture, and gait as well as higher level constructs such as frailty. Here, we present an integrated rodent phenotyping platform, JAX Animal Behavior System (JABS) to the community for data acquisition, machine learning based behavior annotation and classification, classifier sharing, and genetic analysis. JABS Data acquisition module enables uniform data collection with its combination of 3D hardware designs and software for real-time monitoring and video data collection. JABS-Active Learning Module allows behavior annotation, classifier training, and validation. We also present a novel graph-based framework (*ethograph*) that enables efficient boutwise comparison of classifiers. JABS-Database Module allows users to share behavior classifiers and finally the JABS-Analysis Module infers a deposited classifier on a library of 600 open field videos consisting of 60 mouse strains, returns frame level and bout level classifier statistics.In summary, this open-source tool is an ecosystem that allows the neuroscience community to build shared resources for behavior analysis.

1 Introduction

Behavioral analysis seeks to link complex and dynamic behaviors with their underlying genetic and neural circuits [1]. The laboratory mouse has been at the forefront of these discoveries. Animal behavior quantification has rapidly advanced in the past few years with the application of machine learning to the problem of behavior annotation[2]. These advances are mainly due to breakthroughs in the statistical learning field [3–8]. Our lab has previously used computer-vision methods to track visually diverse mice under different environmental conditions [9], infer pose for gait analysis and posture analysis [10], and detect complex behaviors like grooming [11]. We have also used computer-vision derived features to predict complex constructs, such as health and frailty [12].

Although major advances have been made in behavior annotation using computer vision, a major challenge exists in the democratization of this technology. Currently, a high level of expertise is needed for efficient use of these methods. For instance, large amounts of training data are needed to train a pose estimation or grooming detection network. Many labs collect data in disparate ways which lack standardized visual appearance. In order to apply existing models for detecting pose or behavior, each individual lab must train their own model at high cost of labeled data. This is a challenge to the field - many useful models exist, but they cannot be directly applied across data originating from different labs. If visual input is standardized across labs, trained models can be adopted across labs, decreasing barriers to entry and increasing reproducibility of data.

With this in mind, we describe our data acquisition system here. We have developed an integrated mouse phenotyping platform called JAX Animal Behavior System (JABS), which consists of video collection hardware and software, a behavior labeling and active learning app, and an online database for sharing classifiers. In the subsequent sections, we outline the features and design of the JABS data acquisition module, along with the release of all design specifications and accompanying software. Additionally, we provide results on the health and safety of the mice in our long-term monitoring arena, which can be used for Institutional Animal Care and Use Committee applications. The process of manually annotating animal behavior and training classifiers is critical for accurately interpreting the data collected by JABS data acquisition module. We detail the process for behavior annotation and classifier training using the JABS active learning GUI app. We then evaluate the performance of the trained classifiers and explore the inter-annotator variability in classification for the same behavior. We utilize both frame based and bout based metric to capture the differences in predictions between annotators, and demonstrate that the bout based comparison allows for a more comprehensive analysis of how the classifiers are performing on a sequence of behaviors rather than just individual frames. In addition to traditional classifier evaluation methods, we have developed a novel approach called *etho-graph* to facilitate inter-annotator comparison at the bout level. This approach enables construction of graph based representation of bouts of behavior, allowing for a detailed visualization and analysis of difference in annotations. Finally, by utilizing the classifiers trained within JABS, we can quickly and accurately analyze multiple behaviors for our large strain survey consisting of 600 videos representing 60 mouse strains with 5 males and 5 females per strain.

Our hardware and software solution collects high quality data for behavior analysis and has been used for several of our papers [9–12]. Adoption of the JABS will allow labs to use the trained machine learning models described in these manuscripts. In addition, we hope this standard system for the laboratory mouse will democratize the automated quantification of animal behavior using machine learning.

2 JAX Animal Behavior System

The process and various components of JABS are illustrated Figure 1A. Briefly, our system comprises of three components encompassing five different processes, namely, i) data acquisition, ii) behavior annotation, iii) classifier training, iv) behavior characterization, and v) data integration. The first component (JABS data acquisition module) is the custom designed standardized data acquisition hardware and software that provides a controlled environment, optimized video storage, and live monitoring capabilities. The second component (JABS active learning module) is a python based GUI active learning app for behavior annotation and training classifiers using the annotated data. One can then use the trained classifiers for predicting whether behavior happens or not in the unlabeled frames. The last component of JABS is a webapp which provides an interactive user interface to browse through the strain survey results from different classifiers, download classifiers and related training data. The app can also be used to classify various behaviors in user submitted videos (pose files) using the classifiers available in the database. One can also submit a classifier trained using the JABS classification and annotation app and submit it to make predictions on our large strain survey. Next, we discuss the individual components of JABS in detail.

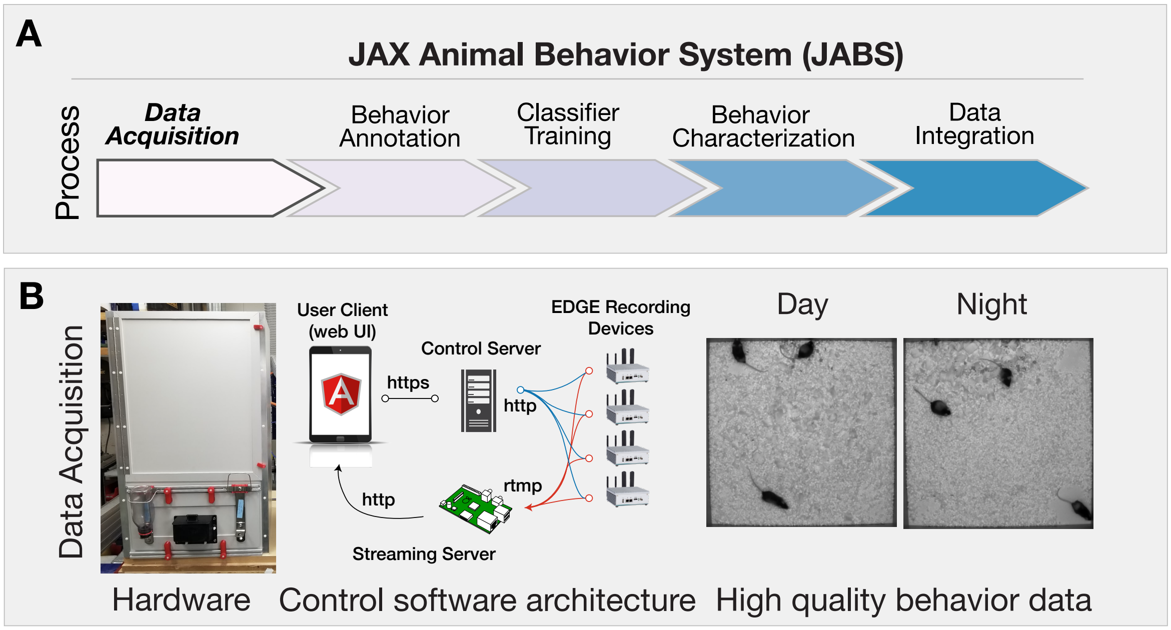


Figure 1: **JABS data acquisition module** (A) JABS pipeline highlighting individual steps towards automated behavioral quantification. (B) Detailed example of JABS data acquisition including a picture of the monitoring hardware, architecture of the real-time monitoring app, and screenshots from videos taken during daytime and nighttime.

2.1 Data acquisition - Hardware and Software

We use a standardized hardware setup for high quality data collection and optimized storage (Figure 1B). The end result is uniform video data across day and night. Complete details of the software and hardware, including 3D designs used for data collection, are available on our Github (`https://github.com/KumarLabJax/JABS-data-pipeline/tree/main`). We also provide a step-by-step assembly guide (`https://github.com/KumarLabJax/JABS-data-pipeline/blob/main/Multi-day%20setup%20PowerPoint%20V3.pptx`).

We have organized the animal habitat design into three groups of specifications. The first group of specifications are requirements necessary for enabling compatibility with our machine learning algorithms. The second group describes components that can be modified as long as they produce data that adheres to the first group. The third group describes components that do not affect compatibility with our machine learning algorithms. While we distinguish between abstract requirements in group 1 and specific hardware in group 2 that meet those requirements, we recommend that users of our algorithms use our specific hardware in group 2 to ensure compatibility.

The design elements that are critical to match specifications in order to re-use machine learning algorithms include (1a) the camera viewpoint, (1b) minimum camera resolution and frame rate, (1c) field of view and imaging distance, (1d) image quality and (1e) the general appearance of the habitat (cage or enclosure). The design elements that are flexible but impact the compatibility are (2a) camera model, (2b) compute interface for capturing frames, (2c) lighting conditions, (2d) strains and ages of mice and (2e) animal bedding contents and quantity. Design elements that have no impact on compatibility are (3a) height of habitat walls to prevent mice from escaping, (3b) animal husbandry concerns, (3c) mounting hardware, (3d) technician ergonomic considerations and (3e) electrical connection hardware and management.

2.1.1 Group 1 specifications

Our system operates on a top-down camera viewpoint. This specification enables flexibility and allows more diverse downstream hardware and ease of construction. The top-down viewpoint enables wider adoption due to construction simplicity and the ability to test more varied assays. While other approaches such as imaging from the bottom through a clear floor are possible and enable better view of animal appendages, they are achieved at the cost of limiting assay duration and construction complexity. For instance, long-term monitoring which requires bedding, and accumulation of feces and urine eventually obstruct bottom up view. We therefore use top-down data acquisition.

Our algorithms are trained using data originating from 800x800 pixel resolution image data and 30 frames per second temporal resolution. This resolution was selected to strike a balance between resolution of the data and size of data produced. While imaging at higher spatial and temporal resolution is possible and sometimes necessary for certain behaviors, these values were selected for general mouse behavior such as grooming, gait, posture, and social interactions. We train and test our developed algorithms against the spatial resolution. We note that these are minimum requirements, and downsampling higher resolution and frame rate data still allows our algorithms to be applied.

Similar to the pixel resolution, we also specify the field of view and imaging distance for the acquired images in real-world coordinates. These are necessary to achieve similar camera perspectives on imaged mice. Cameras must be mounted at a working distance of approximately 100cm above the floor of the arena. Additionally, the field of view of the arena should allow for between $5 - 15\%$ of the pixels to view the walls (FoV between 55cm and 60cm). Having the camera a far distance away from the arena floor reduces the effect of both perspective distortion and barrel distortion. We selected values such that our custom camera calibrations are not necessary, as any error introduced by these distortions are typically less than 1%.

Additionally, image quality is important for meeting valid criteria for enabling the use of machine learning algorithms. Carefully adjusting a variety of parameters of hardware and software values in order to achieve similar sharpness and overall quality of the image is important. While we cannot provide an exact number or metric to meet this quality, users of our algorithms should strive for equal or better quality that exists within our training data. One of the most overlooked aspect of image quality in behavioral recordings is image compression. We recommend against using typical software-default video compression algorithms and instead recommend using either defaults outlined in the software we use or recording uncompressed video data. Using software-defaults will introduce compression artifacts into the video and will affect algorithm performance.

Finally, the general appearance of the cage should be visually similar to the variety of training data used in training the machine learning algorithms. Please read associated documentation on this for each individual algorithm for assessing the limitations [9–12] . While our group strives for the most general visual diversities in mice behavioral assays, we still need to acknowledge that any machine learning algorithms should always be validated on new datasets that they are applied to. Generally our machine learning algorithms earlier in the entire processing pipeline, such as pose estimation, are trained on more diverse datasets than algorithms later in the pipeline, such as pain and frailty predictions.

2.1.2 Group 2 specifications

In order to achieve compliant imaging data for use with our machine learning algorithms, we specify the hardware we use. While the hardware and software mentioned in this section is modifiable, we recommend that careful consideration is taken such that changes still produce complaint video data.

We modified a standard open field arena that has been used for high-throughput behavioral screens

[13]. The animal environment floor is 52 cm square with 92 cm high walls to prevent animals escaping and to limit environmental effects. The floor was cut from a 6mm sheet of Celtec® (Scranton, PA) Expanded PVC Sheet, Celtec® 700, White, Satin / Smooth, Digital Print Gradesquare and the walls from 6mm thick Celtec® Expanded PVC Sheet, Celtec® 700, Gray, (6 mm x 48 in x 96 in), Satin / Smooth, Digital Print Grade. All non-moving seams were bonded with adhesive from the same manufacturer. We used a Basler (Highland, IL) acA1300-75gm camera with a Tamron (Commack, NY) 12VM412ASIR 1/2" 4-12mm F/1.2 Infrared Manual C-Mount Lens. Additionally, to control for lighting conditions, we mounted a Hoya (Wattana, Bangkok) IR-80 (800nm), 50.8mm Sq., 2.5mm Thick, Colored Glass Longpass Filter in front of the lens using a 3D printed mount. Our cameras are mounted 105 +/- 5 cm above the habitat floor and powered the camera using the power over ethernet (PoE) option with a TRENDnet (Torrance, CA) Gigabit Power Over Ethernet Plus Injector. For IR lighting, we used 6 10 inch segments of LED infrared light strips (LightingWill DC12V SMD5050 300LEDs IR InfraRed 940nm Tri-chip White PCB Flexible LED Strips 60LEDs 14.4W Per Meter) mounted on 16-inch plastic around the camera. We used 940nm LED after testing 850nm LED which produced a marked red hue. The light sections were coupled with the manufactured connectors and powered from an 120vac:12vdc power adapter.

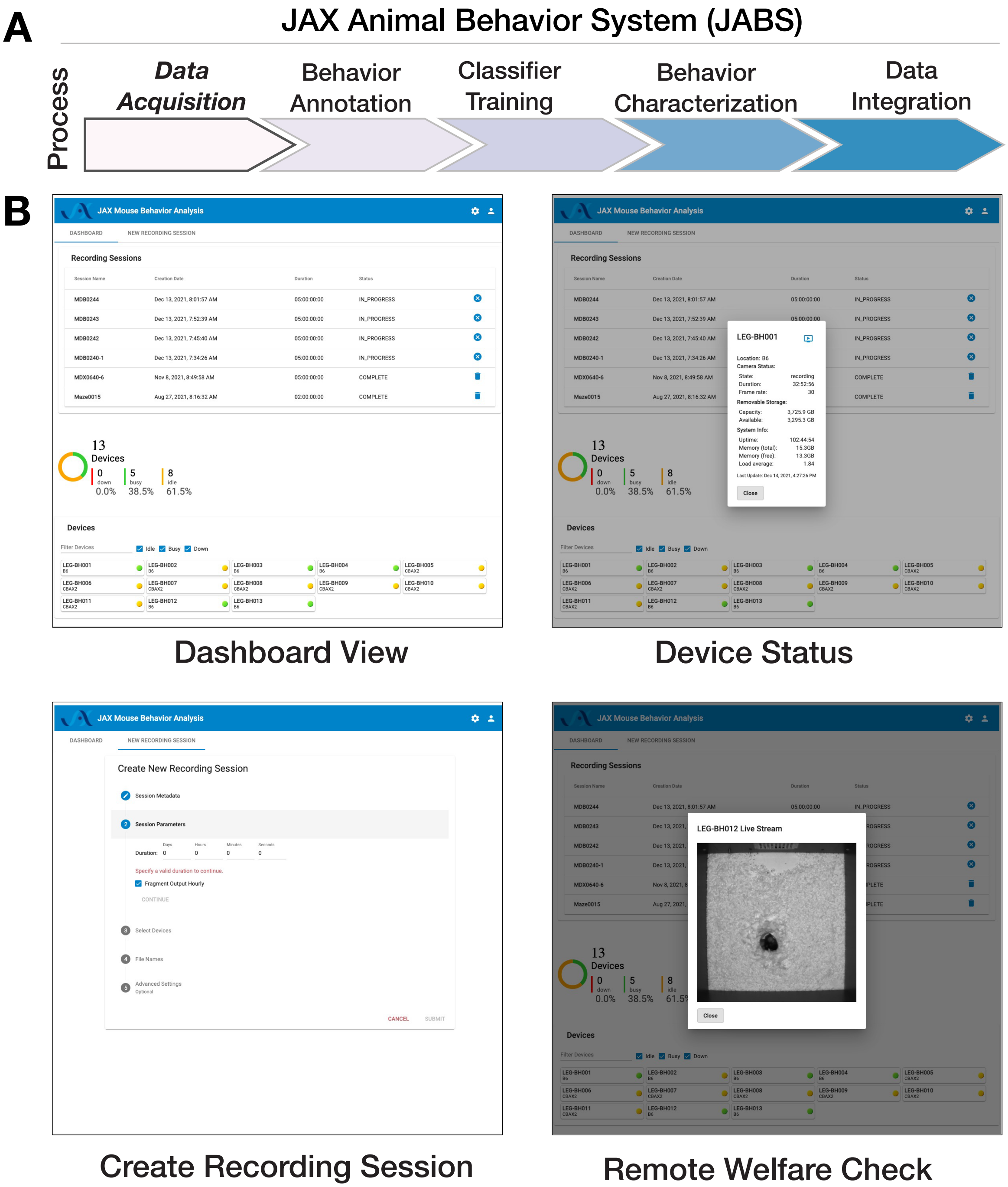
For image capture, we connected the camera to an nVidia (Santa Clara, CA) Jetson AGX Xavier development kit embedded computer. To store the images, we connected a local four-terabyte (4TB) USB connected hard drive (Toshiba (Tokyo, Japan) Canvio Basics 4TB Portable External Hard Drive USB 3.0) to the embedded device. When writing compressed videos to disk, our software both applies optimized de-noising filters as well as selecting low compression settings for the codec. While most other systems rely on the codec for compression, we rely on applying more specific de-noising to remove unwanted information instead of risking visual artifacts in important areas of the image. We utilize the free ffmpeg library for handling this filtering and compression steps with the specific settings available in our shared C++ recording software. Complete parts list and assembly steps are described in (https://github.com/KumarLabJax/JABS-data-pipeline)

2.1.3 Group 3 specifications

Finally, here we present hardware and software that can be modified without risk of affecting video compliance. For natural light, we used a F&V (Netherlands) fully dimmable R-300SE Daylight LED ring light powered by a 120vac:12vdc power adapter. These lights are adjustable to meet the visible lighting needs of specific assays without affecting the visual appearance of the data. To keep the animals nourished, we installed water bottles and a food hopper external to the animal environment. These were placed on the outside of the arena on a removable panel. The panel can be customized as needed for experiments without the need to replace/modify the entire arena. To suspend the camera and lights, we used a wire shelf from our solution for technician ergonomics.

To raise the animal cage to an ergonomic height, we used the 24-inch by 24-inch option of the Metro (Wilkes-Barre, PA) Super Erecta wire shelving system with three shelves. As mentioned in the earlier paragraph, the topmost shelf was used to suspend the camera and lights. We also hinged one wall, turning it into a door, to allow easier animal access. Communication between the electronic devices was interconnected with CAT5 cables and a network switch and a powered USB hub was used between the USB connected hard drive and the nVidia compute device. We used a digital timer for the visible LED light, a 120v power strip to consolidate the power, and a universal power source (battery backup) between the chamber and facility power.

For ease of use and reduction of environmental noise, we also include a software for remote monitoring and welfare check. The software consists of three main components: a recording client implemented in C++, a control server implemented with the Flask Python framework, and a web-based user interface implemented with Angular (Figure 1). The recording client runs locally on each Nvidia Jetson



Recording Sessions

Session Name	Creation Date	Duration	Status
MDB0244	Dec 13, 2021, 8:01:57 AM	05:00:00:00	IN_PROGRESS
MDB0243	Dec 13, 2021, 7:52:39 AM	05:00:00:00	IN_PROGRESS
MDB0242	Dec 13, 2021, 7:45:40 AM	05:00:00:00	IN_PROGRESS
MDB0240-1	Dec 13, 2021, 7:34:26 AM	05:00:00:00	IN_PROGRESS
MDX0640-6	Nov 8, 2021, 8:49:58 AM	05:00:00:00	COMPLETE
Maze0015	Aug 27, 2021, 8:16:32 AM	02:00:00:00	COMPLETE

13 Devices

0 down	5 busy	8 idle
0.0%	38.5%	61.5%

Devices

Filter Devices ☑ Idle ☑ Busy ☑ Down

LEG-BH001 B6 ●	LEG-BH002 B6 ●	LEG-BH003 B6 ●	LEG-BH004 B6 ●	LEG-BH005 CBAX2 ●
LEG-BH006 CBAX2 ●	LEG-BH007 CBAX2 ●	LEG-BH008 CBAX2 ●	LEG-BH009 CBAX2 ●	LEG-BH010 CBAX2 ●
LEG-BH011 CBAX2 ●	LEG-BH012 B6 ●	LEG-BH013 B6 ●		

Dashboard View

LEG-BH001

Location: B6

Camera Status:

State:	recording
Duration:	32:52:56
Frame rate:	30

Removable Storage:

Capacity:	3,725.9 GB
Available:	3,295.3 GB

System Info:

Uptime:	102:44:54
Memory (total):	15.3GB
Memory (free):	13.3GB
Load average:	1.84

Last Update: Dec 14, 2021, 4:27:26 PM

Close

Device Status

Create New Recording Session

1. Session Metadata
2. Session Parameters

Duration: Days 0 Hours 0 Minutes 0 Seconds 0

Specify a valid duration to continue.

☑ Fragment Output Hourly

CONTINUE

3. Select Devices
4. File Names
5. Advanced Settings
   Optional

CANCEL SUBMIT

Create Recording Session

LEG-BH012 Live Stream

Close

Remote Welfare Check

Figure 2: **JABS data acquisition module: Web-based control system for recording and monitoring experiments**. (A) JABS pipeline highlighting individual steps towards automated behavioral quantification. (B) Screenshots from Angular web client that allows monitoring of multiple JABS Acquisition units in multiple physical locations. Dashboard view allows monitoring of all JABS units and their status, Device Status provided detailed data on individual devices, recording session dashboard allows initiation of new experiments, and remote welfare view allows live video to be observed from each unit.

Xavier computer and communicates with the server using the Microsoft C++ REST SDK to provide centralized monitoring and control of distributed recording devices. The recording client captures raw frames from the camera and encodes video using the ffmpeg library. In addition to saving encoded video on the local hard drive, the recording client can optionally send video over the RTMP protocol to a NGINX server configured with the nginx-rtmp plug-in. The web interface communicates with the control server, which relays recording start and stop commands to individual recording devices, enabling the user to remotely control various aspects of recording in addition to viewing the live stream from the NGINX streaming server using the HTTP Live Streaming (HLS) protocol (Figure 2).

2.2 Environment checks

Since the JABS acquisition arena was not previously validated for long-term housing, we carried out a series of experiments to confirm health and welfare of animals over time in these apparatus. These data can be used for Institutional ACUC protocols. We compare our data with established guidelines from the Guide for the Care and Use of Laboratory Animals (the Guide) [14]. Our experiments were performed in one room at The Jackson Laboratory, Bar Harbor, Maine (JAX) with temperature and humidity set to 70-74^{o}F (˜21-23^{o}C) and 40-50%, respectively.

One concern related to use of the JABS arena in long-term experiments was that the 90 cm height of the walls without lower air openings might result in inadequate air flow and buildup of toxic gases. To address this, we compared environmental parameters in JABS arenas with that of a standard JAX housing cage. Two JABS arenas were observed with 12 male C57BL/6J mice 12-16 weeks old in each for a 14-day period. At the same time, one wean cage containing 10 male C57BL/6J age-matched mice was observed on a conventional rack for matching air flow in the same room. We used a #2 Wean Cage (30.80 x 30.80 x 14.29 cm) from Thoren (Hazleton, Pennsylvania) with 727.8 cm^2 floor space, which is a common housing container for mice and is approved at JAX to house 10 animals. This commercial cage has a floor area that is ˜1/4 that of the JABS arena. The ceiling height in the wean cage ranges 5-14 cm due to the sloped metal wire cover that contains food and water. The JABS arena, by contrast, has no ceiling.

Food, water, bedding type and depth and light level were all matched in the arenas and wean cage. Bedding (1:1 ratio of aspen chip/shavings mix (chip from P.J. Murphy Forest Products, Montville, New Jersey and shavings from Northeast Forest Products Corp., Warrenburg, New York) and Alpha-Dri (Shepherd Specialty Papers)) was left unchanged for the full two week period in all pens, as we wish to minimize interaction with long term monitoring mice in JABS arenas as much as possible. To determine if forced air flow was needed for an acceptable arena environment, one of two arenas and the wean cage were exposed to normal room air flow, while the second arena had a 6” quiet electric fan mounted above. The fan was pointed so as to blow air up, potentially drawing air out of the arena while not actively blowing air over the mice. The choice of the fan blowing upwards was based on a previous experiment with the fan blowing down into the arena starting day 7. In that experiment, we observed that mice in the arena with a fan lost more weight than mice in the arena with no fan, suggesting the fan blowing down introduced a confounding negative environmental factor.

We checked for CO_2 and ammonia, common gases that can build up in housing [14]. CO_2 was measured using an Amprobe (Everett, Washington) CO_2-100 meter with levels checked daily except weekends and holidays throughout the experiment in both arenas and the wean cage. CO_2 measurements were taken in the middle of the room before and after each arena and wean cage measurement as a control. For higher levels, CO_2 is shown as a range, as it was observed to oscillate between high and low values over reading periods lasting several minutes. Ammonia was tested using Sensidyne (St. Petersburg, Florida) Ammonia Gas Detector Tubes (5-260 ppm) in the arena without a fan and the wean cage on days 0, 2, 4, 7 and 14, with air samples taken near floor level and areas of greatest waste accumulation in both. MadgeTech (Warner, New Hampshire) RHTEMP1000IS temperature and humidity data loggers were placed on the floor in each arena and the wean cage and left for the duration of the experiment. An environment monitor (Hobo, U12-012, temperature/humidity/light monitor, Onset, Bourne, Massachusetts) was mounted on the wall of the room to provide background levels. Body weight measurements were taken daily except weekends and holidays. Grain and water were weighed at the start and end of each experiment to check consumption.

We observed daily room background CO_2 levels of 454 to 502 ppm throughout the 14-day experiment. These are very close to expected outdoors levels and indicative of a high air exchange rate [15]. JABS arena CO_2 levels varied from a low of 511 ppm on day 1 to an oscillating range of 630 to 1565 ppm on day 14. The JAX standard wean cage experienced an oscillating range of 2320 to 2830 ppm on day 0 climbing to an oscillating range of 3650 to 4370 ppm on day 14. The wean cage CO_2 values approximately match those from another published study of maximum grouped mice in similar static housing [16]. Indoor CO_2 is often evaluated as level above background [15]. We observe a maximum JABS arena CO_2 level above background of 1082 ppm. This is 3.8 fold lower than the maximum observed maximum CO_2 level above background observed in the wean cage (4121 ppm) (Figure 3A, arena with fan excluded from graph for clarity).

Ammonia levels in the JABS arena were below the detection threshold of 5 ppm on days 0, 2, 4 and 7, and rose to 18 ppm on day 14. Ammonia levels in the wean cage were <5 ppm on days 0 and 2, rose to 52 ppm on day 4 and reached levels of ˜230ppm on both days 7 and 14. An early concern was that the high walls of the JABS arena may inhibit air flow and lead to a buildup of toxic gases. CO_2 and ammonia levels both indicate that the JABS arena has greater air exchange than standard conventional housing. The National Institute for Occupational Safety and Health (NIOSH) recommended maximum ammonia exposure for humans is 25 ppm over an 8 hour period. There is no mouse standard, but a similar recommended maximum continuous exposure of 25 ppm for mice is sometimes used [14, 17]. Ammonia levels are influenced by several variables, with the most prominent being air changes per hour (ACH) [18, 19]. JAX animal rooms have ˜10 ACH and PIV cages in those rooms have ˜55-65 ACH. We find that ammonia levels remained 10-50 fold lower in the JABS arena than our control standard static wean cage and well within the strictest recommended range (Figure 3B). One consideration in use of JABS arenas for observations will be the impact of ammonia levels on behavior [20]. Currently all mice moved into JABS arena experiments in our facility are coming from PIV housing. Although we did not measure ammonia levels in PIV housing, we anticipate from Ferrecchia et al. [19] that ammonia levels will be similar in the PIV home cage the mice come from and the JABS arena they go into, and this would have minimal impact upon observed behavior.

Temperatures in all locations (room background, two JABS arenas and one wean cage) remained in a range of 22-26^{o}C throughout the experiment. Variance in room background readings suggest temperature fluctuations are more due to innate room conditions (such as environmental controls) than anything else. We find that arena structure does not adversely affect control of the temperature to which mice are exposed (Figure 3C).

The probes which measured temperature also measured humidity. The room probe, mounted on one wall of the 8'x8' room at ˜6' above the floor, measured consistent background humidity of 45+/-5% throughout the experiment (Figure 3D, green line). Housing probes were placed in the bedding on the floor of each chamber - near the center in the JABS arenas and along one wall in the much more space limited wean cage. Humidity in the JABS arenas was 55-60% throughout the 14-day experiment, aside from notable sporadic spikes (Figure 3D, blue and black lines). The spikes do not correlate with background humidity changes and are interpreted as occasions when the mice urinated on or very near to the probe, which evaporated quickly. By contrast, humidity levels within the wean cage started in the same 55-60% range but quickly climbed above 75% (within 12 hours) and did not come back down as humidity in the JABS arenas had done. Wean cage humidity then continued a gradual climb, reaching 97.5% by day 14 (Figure 3D, red line). While both the JABS arenas and the standard wean

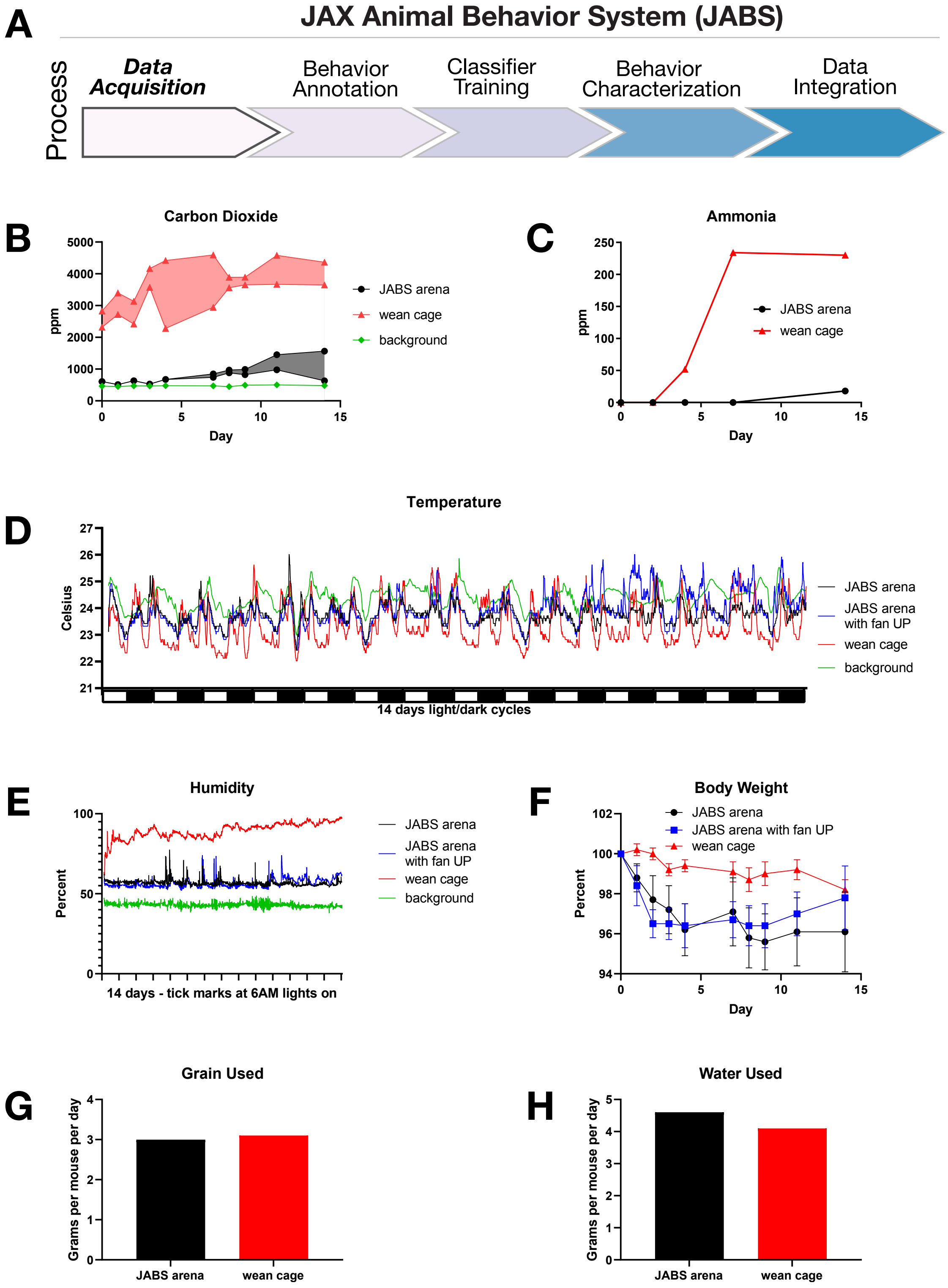


Figure 3: **JABS data acquisition module: Environmental parameters in the arena.** (A) JABS pipeline highlighting individual steps towards automated behavioral quantification. (B) Carbon dioxide concentrations and (C) ammonia concentrations were both much higher in the standard wean cage than in the JABS arena. Carbon dioxide was also compared to room background levels. (D) Temperature and (E) humidity measured at floor level in JABS arenas and a standard wean cage compared to room background across a 14 day period. (F) Average body weight as percent of start weight in each JABS arena and wean cage across the 14 day period. (G) Food and (H) water consumption shown as grams per mouse per day for one JABS arena and one wean cage for a 14 day period.

cage can be anticipated to have higher humidity in their micro-environments compared to the room macro-environment due to the mice contributing moisture via urination and the housing structures somewhat inhibiting air flow (per the Guide [14]), the JABS arenas were found to maintain a much drier environment. This is likely partly due to a greater bedding to mouse ratio (the arena has 1.25 times as many mice but 4 times as much bedding material in this test, so 3.2 times as much bedding per mouse in JABS arena compared to the wean cage) but also due to superior air circulation in the arena compared to the wean cage. (Figure 3D).

Weight is often used as a marker for health though body condition score is used as a more reliable indicator of serious concerns [21, 22]. Mice in arenas lost weight compared to those in the wean cage and this was initially a cause of concern. However, mice in JABS arenas maintained a healthy appearance and normal body condition score throughout the experiment. Other measurements demonstrating normal parameters and other control experiments not shown additionally led us to believe the weight differences are because JABS arena mice are active while wean cage mice, with more limited movement available, are sedentary. Mice started the experiment at 25-33 grams body weight. The lowest average recorded during the experiment was 95.6% of the start value, for mice in the JABS arena without a fan on day 9. The lowest individual recorded was 85.8% of start value at 23.6 grams on day 14, also in the arena without a fan (Figure 3E).

Per mouse grain usage was comparable between the JABS arena and the wean cage and in an expected range [23] (Figure 3F). Per mouse water usage was comparable between the JABS arena and the wean cage and in the expected range [24]. Somewhat higher water use in the arena could be indicative of higher activity requiring more hydration (Figure 3G). Since only one JABS arena and one wean cage were tested, error bars are not available to aid in interpretation.

Three mice from one arena and three from a wean cage were necropsied immediately following 14 days in the JABS arena or control wean cage to determine if any environmental conditions, such as possible low air flow in arenas potentially leading to a buildup of heavy ‘unhealthy’ gases like ammonia or CO_2, were detrimental to mouse health. Nasal cavity, eyes, trachea and lungs were collected from each mouse. They were H&E stained and analyzed by a qualified pathologist. No signs of pathology were observed in any of the tissue samples collected (Figure 10).

Based on these environmental and histological analysis, we conclude that the JABS arena is comparable and in many respects better than a standard wean cage. Lack of holes near the floor do not create a buildup of ammonia or CO_2. Mice ate and drank at normal levels. We observe a slight decrease in body weight initially, which is gained in the next few days. We hypothesize that this could be due to the novel environment and increased space for movement, leading to more active mice.

2.3 JABS Active learning module

In the section, we first present an overview on behavior annotation and classifier training using JABS active learning module which utilizes our python-based, open-source graphical user interface (GUI) application which has been developed to be compatible with Mac, Linux and Windows operating systems. We then evaluate the utility and accuracy of JABS trained classifiers through two complementary approaches. In the first approach, we benchmark the performance of JABS classifiers against a previous neural network based approach [11], providing us a comparison of the performance of the two approaches on the same dataset. In the second approach, we studied how classifiers for the same behavior trained by two different human annotators in the lab compare with each other in terms of behavior identification, allowing us to assess the inherent variability among expert annotators.

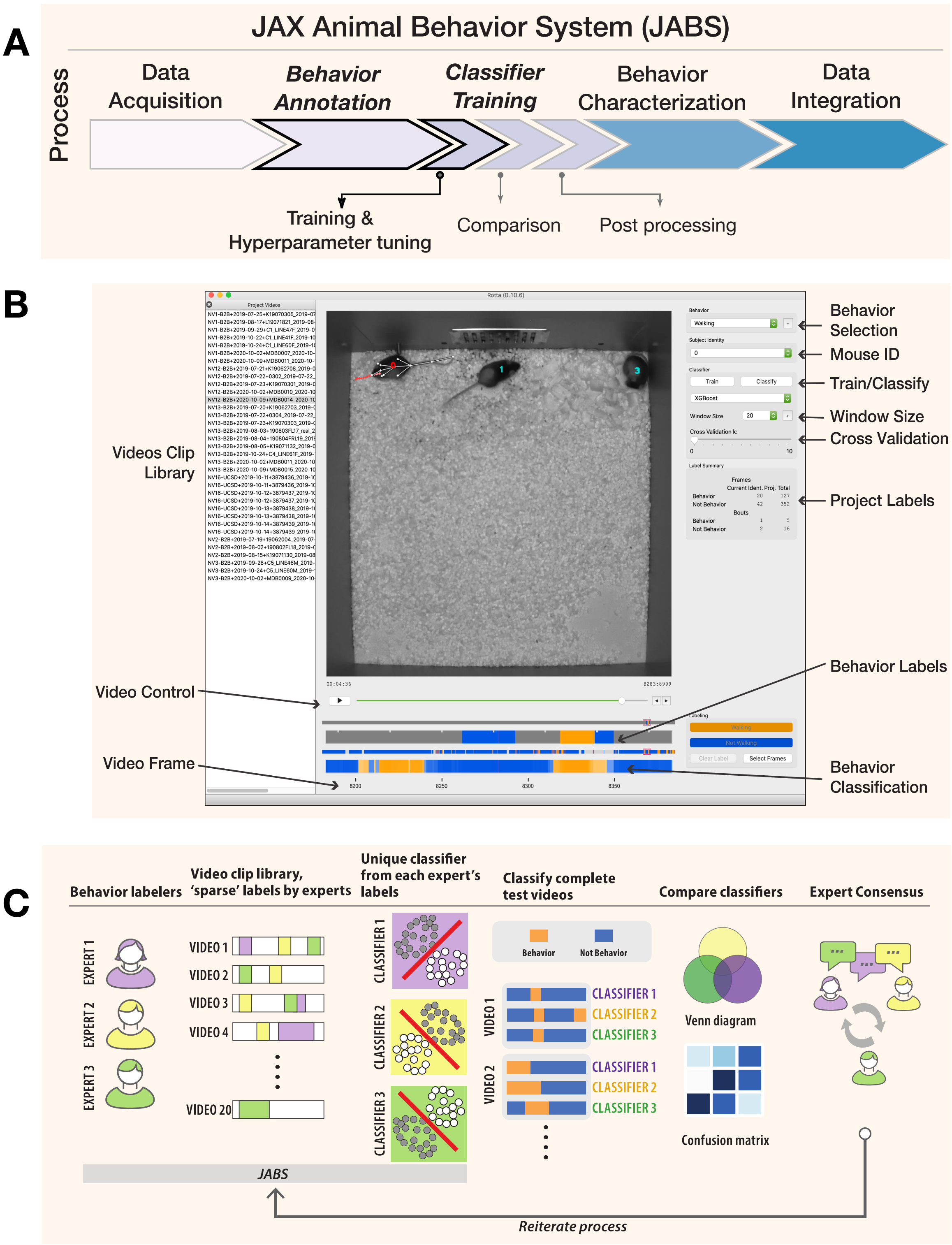
2.3.1 Behavior annotation and classifier training

There are two prominent approaches in the literature for training behavioral classifiers. The first approach trains the classifiers using the raw video files, as previously demonstrated to identify grooming behavior through the use of a deep neural network [11]. The second approach involves first extracting pose keypoints in each frame using deep neural networks, which serves as inputs for machine learning classifiers. Previously, we utilized a deep neural network based classifier to extract poses and used the keypoints to study gait behavior [10]. Pose based approach offers the flexibility to use the identified poses for training classifiers for multiple behaviors and we used this approach for JABS. Additionally, the extracted keypoints can also be used to generate quantifiable and interpretable features that can be used to study various aspects of animal behavior such as gait and posture. In addition to the raw video file, JABS annotation and classification active learning module requires pose files from our previously established neural network for pose estimation as an input to train the classifiers. Note that the raw videos are needed only for annotating behaviors, and one can predict the behaviors using only the pose files.

We have developed an easy to use open source python GUI software to annotate behaviors in videos, as shown in Fig. 4A. This tool allows users to easily annotate behaviors in video recordings through mouse/trackpad or keyboard shortcuts, as well as the option to leave frames unlabeled for ambigious cases. The GUI provides statistics of the total number of frames as well as the number of frames and bouts annotated for a particular behavior. The annotations are displayed below the video as an ethogram (Fig. 4B).The user can annotate multiple behaviors for the same video. Once minimum number of frames (100) and videos (2) have been annotated, the user can train a classifier using either of the tree-based methods such as Random Forest (RF) [25] /Gradient Boost/XGBoost (XGB) [26] and check the accuracy of the classifier by selecting k-fold cross validation. We used our HRNet based pose estimation neural network [10] to estimate location of twelve keypoints in the videos and computed a number of per frame and per window features. We then compute a number of informative features like distance between various keypoints, linear and angular velocity between keypoints, etc. that are used as input for these classifiers. We also incorporate temporal information from the videos by computing window features that include information from w (window size) frames on each side of the current frame. A complete list of base features currently included in JABS is provided in the supplementary information (Table 2). The weights of different features used by the trained classifiers improve the interpretability of the classifiers. Typically, to arrive at an optimal classifier for a behavior, we start by training multiple classifiers using annotated data from different human experts for the same set of videos and then evaluating performance of each classifier against test videos as depicted in Fig. 4C. Further, since there is no ground truth for the test videos, we compare each frame level and bout level predictions from each classifier against each other to evaluate the degree of agreement and consistency. Finally, depending on the expert consensus on the desired level of agreement, a classifier is selected or the whole process is repeated with new or corrected labels. Once the training is completed, the classifier can be exported and be used to predict labels for every unlabelled frame in all the videos in the project directory. One can even use the command line interface of the app for high performance computing environment to train and/or predict using the python scripts included with the software. The detailed user guide along with a video tutorial to install and run JABS active learning app is available online (https://jabs-tutorial.readthedocs.io/en/latest/JABS_user_guide.html).

2.3.2 Benchmarking JABS classifier using grooming behavior

Previously, we trained a neural network for grooming behavior which attained human level accuracy [11]. We re-purpose this large training dataset as a benchmark for estimating learning capacity of pose-based classifiers. Further, we evaluate how the performance of the classifier varies with the choice of machine learning algorithm, window size (w) of the features and the amount of training data.



Project Videos

NV1-B2B+2019-07-25+K19070305_2019-07
NV1-B2B+2019-08-17+L19071821_2019-08-
NV1-B2B+2019-09-29+C1_LINE47F_2019-0
NV1-B2B+2019-10-22+C1_LINE41F_2019-10
NV1-B2B+2019-10-24+C1_LINE60F_2019-10
NV1-B2B+2020-10-02+MDB0007_2020-10-
NV1-B2B+2020-10-09+MDB0011_2020-10-
NV12-B2B+2019-07-21+K19062708_2019-0
NV12-B2B+2019-07-22+0302_2019-07-22_
NV12-B2B+2019-07-23+K19070301_2019-0
NV12-B2B+2020-10-02+MDB0010_2020-10
NV12-B2B+2020-10-09+MDB0014_2020-10
NV13-B2B+2019-07-20+K19062703_2019-0
NV13-B2B+2019-07-22+0304_2019-07-22_
NV13-B2B+2019-07-23+K19070303_2019-0
NV13-B2B+2019-08-03+190803FL17_real_2
NV13-B2B+2019-08-04+190804FRL19_2019
NV13-B2B+2019-08-05+K19071132_2019-0
NV13-B2B+2019-10-24+C4_LINE61F_2019-
NV13-B2B+2020-10-02+MDB0011_2020-10
NV13-B2B+2020-10-09+MDB0015_2020-10
NV16-UCSD+2019-10-11+3879436_2019-10
NV16-UCSD+2019-10-11+3879436_2019-10
NV16-UCSD+2019-10-12+3879437_2019-10
NV16-UCSD+2019-10-12+3879437_2019-10
NV16-UCSD+2019-10-13+3879438_2019-10
NV16-UCSD+2019-10-13+3879438_2019-10
NV16-UCSD+2019-10-14+3879439_2019-10
NV16-UCSD+2019-10-14+3879439_2019-10
NV2-B2B+2019-07-19+19062004_2019-07-
NV2-B2B+2019-08-02+190802FL18_2019-0
NV2-B2B+2019-08-15+K19071130_2019-08
NV3-B2B+2019-09-28+C5_LINE46M_2019-
NV3-B2B+2019-10-24+C5_LINE60M_2019-
NV3-B2B+2020-10-02+MDB0009_2020-10-

Label Summary

	Current Ident.	Proj. Total
Frames		
Behavior	20	127
Not Behavior	42	352
Bouts		
Behavior	1	5
Not Behavior	2	16

Figure 4: **JABS Behavior annotation module: Behavior annotation and classification.** (A) JABS pipeline highlighting individual steps towards automated behavioral quantification. (B) Screenshot of the python based open source GUI application used for annotating multiple videos frame by frame. One can annotate multiple mouse and for multiple behaviors. The labeled data is used for training classifiers using either random forest or gradient boosting methods. Adjustable window size (number of frames on the left and right of the current frame) to include features from a window of frames around the current frame. The labels and predicted labels are displayed at the bottom. (C) A sample workflow for training a typical classifier. Multiple experts can sparsely label videos to train multiple classifiers for the same behavior. These classifiers can be compared and experts can consult to iterate through the training process

For the choice of machine learning algorithm, we utilize two popular tree-based methods, namely Random Forest (RF) and XGBoost (XGB). Briefly, the dataset contains 1,253 video segments, and we held out 153 video clips for validation. This split results in similar distributions of frame-level classifications between training and validation sets. More details of the dataset are available in Table-1. We trained multiple classifiers by varying the amount of annotated data, window size, and machine learning algorithm. Our best accuracy from the neural network based approach for this dataset was 0.937 and the best classifier from JABS using all the annotated data, a window size (w) of 60 frames, and XGB machine learning algorithm achieved a comparable per-frame accuracy score of 0.9364. We noticed that with the same set of features, XGB typically achieved better accuracy than RF method across different window sizes and training data size. The results for these benchmark tests are shown in Figure 5B-D. Our tests with different window sizes show that grooming performance increases as we increase the window size, reaches a maximum (around 60 frames) and then degrades for large window sizes (Fig. 5B). Because grooming typically lasts for few seconds, classifiers using features within nearby frames will perform better as they incorporate optimal temporal information and including features from too few or too many frames will decrease the performance. We also investigate the impact of the amount of labeled data on the performance of JABS classifiers, as it can help to optimize the annotation process, ultimately reducing the time and resources required to train the model. To do this, we trained the XGB and RF classifiers using a subset of the full dataset (about 20 hours) consisting of 10, 20, 50, 100, 500 and 1100 training videos. These correspond to approximately 1.3%, 2.2%, 4.4%, 8.5%, 46.1% and 100% out of a total of 2181790 frames. As expected, the performance of JABS improves as we include more labeled data. However, the results demonstrate that a high degree of accuracy, approaching 85%, can be attained through the utilization of only 10 videos of training data, as evidenced by the corresponding area under the receiver operating characteristic curve (AUROC) of approximately 0.94, as depicted in Figure 5C-E. Additionally, it was found that the true positive rate (TPR) experienced a minimal decrease of about 1% when the training data was reduced from 100% to 50%, while maintaining a false positive rate (FPR) of 5% (Fig. 5F).

In the rapidly evolving field of automated quantification of animal behavior, two predominant methodologies have been established for learning behavior: using raw video data and using a reduced representation of the animal with certain keypoints, from which informed features are calculated [10, 27–29]. To understand the trade-offs and strengths of each approach, we evaluate the performance of different classifiers that employ these methodologies when utilizing varying amounts of training data, as depicted in Fig. 5G. Interestingly, our findings demonstrate that utilizing keypoint-based low dimensional representation of animal behavior, as employed by JABS and JAABA [29] methodologies, leads to superior performance when compared to using high dimensional raw video data as employed by 3D CNNs, particularly when the availability of training data is limited. However, as the quantity of training data increases, the performance of both approaches tend to converge.

Therefore, by distilling the essence of a video into a series of key poses, JABS is able to effectively learn and generalize, even with smaller training sets. It has been shown to have a learning capacity on par with deep neural networks, as demonstrated by per-frame accuracy using the same benchmark data-set. Further, achieving 85% accuracy with just 1.4% of the labeled data, suggests that researchers can strike a balance between labeling efforts and desired accuracy by carefully selecting the amount of training data.

2.4 JABS Analysis module

In supervised machine learning, the accuracy and reliability of a trained classifier depends heavily on the quality of labeled data. Further, it has been observed that labeling of the same behavior by different human experts introduces variability among annotations due to variety of factors, including personal biases, subjectivity, and individual differences in understanding what constitutes a behavior [30, 31].

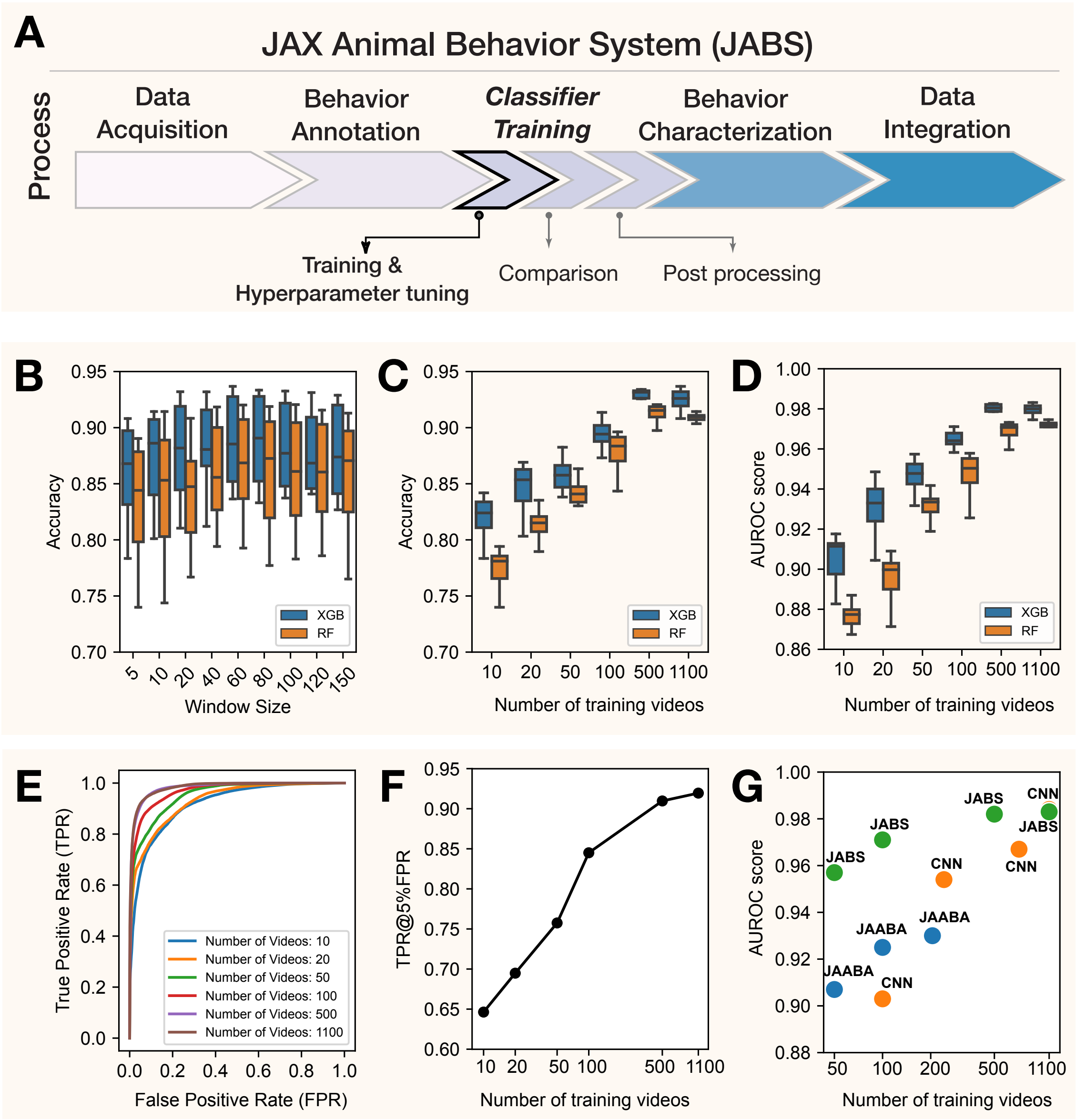


Figure 5: **JABS classifier training module: Selecting hyper-parameters and benchmarking JABS classifiers using grooming dataset.** (A) JABS pipeline highlighting individual steps towards automated behavioral quantification. Using feature window size, type of classification algorithm and the number of training videos as our benchmarking parameters: (B) Accuracy of JABS classifiers trained using different window size features. Each boxplot shows the range of accuracy values for different number of training videos and type of classification algorithms. (C, D) The effect of increasing the training data size on Accuracy and AUROC score of the JABS classifiers. (E) ROC curves for the JABS classifier trained with the window size of 60, XGB algorithm and varying training data size. (F) True positive rate at 5% false positive rate corresponding to the JABS classifier from panel (E) as the amount of training data is changed. (G) Comparing the performance of JABS based classifiers with a 3D Convolutional neural network (CNN) and JAABA based classifiers for different training data sizes.

Therefore, it is critical to accurately capture the inter-annotator variability before selecting classifiers for downstream predictions. To capture this variability, we employ both frame based and bout based comparison and demonstrate that bout-based comparison gives a better estimate of inter-annotator agreement.

2.4.1 Frame and bout-wise classifier comparison of inter-annotator variability

In order to test inter-annotator variability, we use generated a set of single mouse behavior classifiers for two simple behaviors, left and right turn. We inferred behavior from all four classifies on a large set of videos and compared the two pairs of classifiers from each annotator (Figures 6, 7). The classifiers for all behaviors achieved good accuracy and F1 scores (Table 3). Further, the classifiers for the same behavior trained with different human annotations resulted in inter-annotator variability in predictions. This inter-annotator variability can be associated with (a) subjective differences of behavior definition among human labelers (b) varying level of annotator's expertise, and (c) training with-in and across labs. We investigated the source of this variability and sought to determine the best method to mediate its effects. To capture this effect, we first visualized the predictions made by two classifiers trained for the same behaviors (left and right turn) but with different human annotators: annotator-1 (A1) & annotator-2 (A2). Figure 6B,C shows two sample ethograms corresponding to the predictions made by A1 & A2 for the left turn behavior. These ethograms show high level of concordance between the two annotators. However, upon closer examination, we observed that the percentage of left or right turn behavior predicted (for all the videos) by A2 was higher than A1 (see Figure 6D,G). The confusion matrix (shown in Figure 6E,H) quantifies the level of agreement between predictions made by annotators A1 and A2 for left and right turn behavior. However, since this behavioral task is heavily class-imbalanced (the number of frames with no-behavior is much more than that of behavior), accuracy can be misleading, as the classifier can achieve high accuracy by simply predicting majority class (not behavior) for all the frames. To address this imbalance, we calculate Cohen's kappa (κ) metric [32] which is a commonly used measure of inter-annotator agreement accounting for the class imbalance. Mathematically, it is defined as $\kappa = \frac{p_o - p_e}{1 - p_e}$, where p_o is the observed agreement between annotators and p_e is the expected agreement due to random chance. A κ score of 0 indicates that the agreement is no better than chance, and a score of 1 indicates perfect agreement, regardless of high/low accuracy. Finally, we visualize the frame-wise comparison of the two annotators showcasing the percentage of frames where the annotators agree and disagree on the occurence of a behavior as shown in Figure 6F,I. The venn diagram clearly highlights the discrepancy between high accuracy resulting from class imbalance (Figure 6E,H) and significant mismatch between % of predicted behavior (Figure 6D, G), with annotator A2 account for majority of discrepancy by predicting more frames as turning behavior compared to annotator A1.

We observed in the ethogram (Figure 6B,C) that although many of the same bouts are captured by both A1 and A2, most of the frame discrepancies seem to be in the beginnings and ends of the bout. A2 seems to predict longer bouts than A1 (Figure 6D). Between two humans labeling the same behavior, there are unavoidable and sometimes substantial discrepancies in the exact frames of the behavior labeled even when trained in the same lab [27, 31]. To most behaviorists, detecting the same bouts of behavior is more important than the exact starting and ending frame of these bouts– as again, there are human-level discrepancies in this as well. Therefore, we used a bout-based comparison rather than a frame-based comparison to evaluate the performance of the classifiers.

For the bout-based comparison, we looked at how much overlap there was between the bouts of a behavior predicted by annotators A1 and A2, taking inspiration from the machine learning image-recognition and action-detection fields, where an overlap of pixels of the bounding box and ground truth label box called the intersection over union (IoU) [33, 34]. We developed a graph-based approach called an *ethograph* to represent the bouts of behavior recorded in the ethograms of annotators A1 and

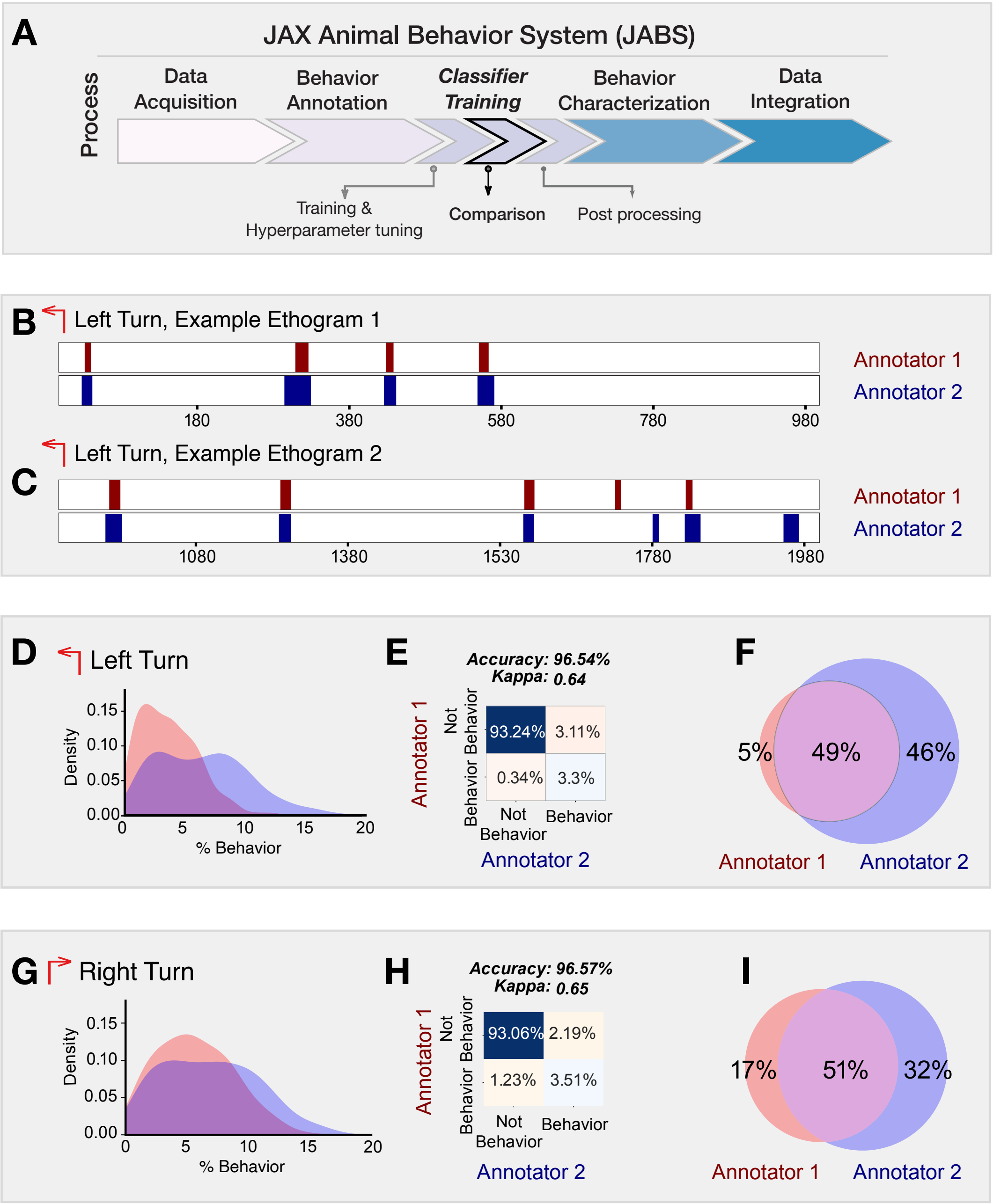


Figure 6: **JABS classifier training module: Frame based comparison of classifiers from different annotators but trained for the same behavior**. (A) JABS pipeline highlighting individual steps towards automated behavioral quantification. (B, C) Two sample ethograms for the left turn behavior showing variation in behavior inference for two different annotators. (D, G) Kernel density estimate (KDE) of the percentage of frames predicted to be a left turn and a right turn respectively, by each annotator across all the videos.

The major discrepancy between the two annotators is that A-2 systematically predicts larger number of frames as behavior compared to A-1. (E, H) Confusion matrix showing the agreement between predictions of two classifiers over all the videos in the strain survey for left and right turn behavior. (F, I) Venn diagram capturing the frame-wise behavior agreement between the two annotators for left and right turn behavior.

A2. Concretely, we define the ethograph for two annotators as a bipartite graph $\mathscr{G} = (U, V, E)$, where U, V are two disjoint sets corresponding to bouts predicted by each annotator and E represents the edges that connect each element in set U to an element in set V capturing the overlap in time between the bouts. Further, the vertices of an ethograph represents bouts with vertex color encoding for the annotator and vertex size proportional to the duration of the bout. Further, the edges (E) of the graph ($\mathscr{G}$) represents the temporal overlap between the bouts (corresponding to different annotators) with the thickness of the edge proportional to the amount of bout overlap. Figure 7B,C shows the ethograms and their associated ethograph for the left turn behavior as predicted by annotators A1 and A2. In contrast to traditional frame-based ethograms, which simply displays the sequential list of frames in which a behavior is observed, the ethograph allows for a more intuitive and visual representation of the temporal overlap between the bouts corresponding to different annotators (or even behaviors). This can be especially useful in identifying patterns and trends that may not be immediately apparent from comparing ethograms. By coloring the vertices and edges based on the annotator, it becomes easy to see which behaviors are consistently identified by both the annotators and which are more subjective and open to interpretation. Moreover, we can easily compute the bout-based agreement between the two annotators as the fraction of edges having thickness greater than some fixed threshold (see figure 7F for mathematical definition) which essentially means the fraction of bouts having overlap greater than a chosen overlap threshold. The bout agreement between two annotators for the left and right turn at a threshold of 0.5 is shown as a Venn diagram in Figure 7D,E along with the density distribution of bout length. The agreement between two annotators with bout-based measure was certainly much better than that with frame-based comparison (see figure 6F,I).

The predictions coming out of a classifier contained many short bouts (1-3 frames) of behavior that signal false positive bouts as they are much shorter than a typical bout of annotated behavior. Moreover, certain bouts of behavior were split by very short bouts (1-3 frames) of not-behavior signalling the presence of false negative bouts that results in fragmentation of a bout of behavior (see figure 7). To address this issue, we proposed a stitching and filtering step on the predictions coming out of classifier. First, we stitched those bouts whose distance to the neighboring bout is less than certain fixed threshold. This stitched the fragmented bouts as illustrated in Figure 7G. We then applied bout filtering which removed bouts of a length below a fixed threshold. To decide the optimal values of stitching and bout filtering thresholds, a hyper-parameter scan was performed for each behavior. Figure 7H,I presents the results from hyper-parameter scan over stitching and bout filtering thresholds when the value of percentage bout overlap is fixed at 25%, 50% and 75% for left (H) and right turn (I). Figure 7J captures the effect of applying bout filtering and stitching to a portion of an ethogram corresponding to the predictions made by A1 & A2 for the left turn behavior. The effect was clearly discernible when looking at the changes in ethograph, particularly with bouts (nodes) having multiple overlaps (edges) reducing to single overlap (edge) per bout.

In summary, when comparing classifiers, it's important to consider the inherent variability of human annotators. Frame-wise comparison penalizes this natural variability, making it a sub-optimal measure of agreement. On the other hand, bout-wise comparison takes this variability into account, making it a more biologically meaningful measure of agreement between classifiers. In addition to using bout-wise comparison, applying techniques like stitching and filtering can further improve agreement by reducing false and fragmented bouts in classifier predictions. By considering these factors, we can better understand the inter-annotator variability and design more effective guidelines for behavior annotation.

2.4.2 Strain Survey of Multiple Behaviors

One of the advantages of a standardized data acquisition system such as JABS is that data can be repurposed. For instance, a classifier trained by another lab could be inferred on videos generated

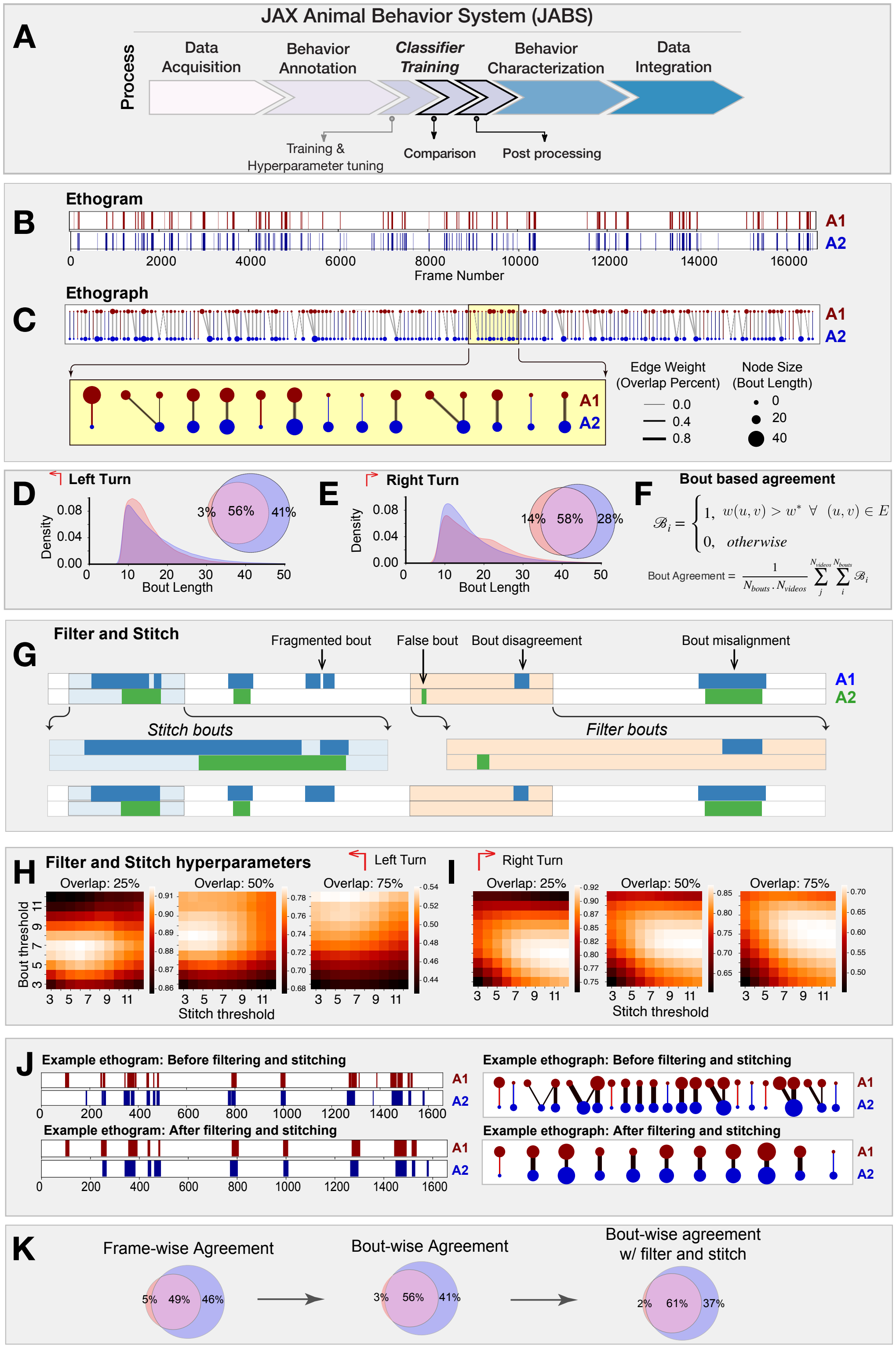


Figure 7: **JABS classifier training module: Bout based comparison of classifier predictions from different annotators but trained for the same behavior**. (A) JABS pipeline highlighting individual steps towards automated behavioral quantification. (B) Ethogram depicting frame-wise left turn predictions for annotators A1 (red) and A2 (blue). (C) Ethograph corresponding to the ethogram in panel (B) capturing the bout level information as a bipartite network. The nodes represent bouts with node size & color proportional to the bout length & annotator respectively. Edge weights captures the fraction of bout overlap between two bouts predicted by different annotators for the same behavior. Edge weight and node size with zero value indicate missed bouts by an annotator. These have been given a small positive value for visualization purposes only. (D-E) Bout length distribution of annotators A1 & A2 for left and right turn behavior. (F) The mathematical definition of the average bout agreement between two annotators, where $w(u,v)$ represents weight between nodes u and v ($u \subset U$, $v \subset V$) in the ethograph $\mathscr{G}(U,V,E)$ and w^* is the bout overlap threshold (0.5 fixed for our study). (G) overview of the workflow for stitching and filtering at the bout level. (H, I) Hyper-parameters tuning to find optimal filtering and stitching thresholds. (J) Sample ethogram and its corresponding ethograph before and after applying stitching and filtering. (K) Inter-annotator agreement in frame wise predictions underestimates the agreement whereas the bout wise comparison post filtering and stitching captures the overall agreement in a more biologically meaningful way.

by another lab. We trained a set of behavior classifiers using JABS active learning system and then inferred them on a previously published strain survey dataset [9]. The training dataset was composed of multiple human-annotated short videos (around 10 minutes each), we trained classifiers for left turn, right turn, grooming, rearing supported, rearing unsupported, scratch and escape as examples. These can easily be extended to other behaviors. To capture the effect of genotype on the behavior, we subsampled the original strain survey data set to 600 one-hour open field videos representing 60 different strains with 5 female and 5 male for each strain and make predictions using the trained classifiers. Further, we define 3 aggregate phenotype associated with each behavior namely the total duration of the behavior (in minutes) for the first 5, 20 and 55 minutes of the one-hour video [11], to capture the dynamic changes in behavior over time. The results are shown in Figure 8B, where the heatmap shows the Z-scores for the total duration of the behavior in 5, 20 and 55 minutes ($|$Z-score$| > 1$ thresholding is applied for easier visualization). The red and blue colored entries for a particular phenotype represents strains exhibiting the behavior that is more than one standard deviation above and below the mean of the phenotype respectively. Such data can have multiple utility. First, any user of JABS can conduct a rich analysis with little effort to yield biological insight. Such data can be used to refine classifiers by adding edge cases to training data. In addition, downstream genetic analysis suchs as heritability quantification and GWAS analysis are possible with this data [10, 11]. In our analysis, we observed a high number of escape attempts in C58/J mice. This strain has been shown previously to have high number of repetitive behaviors, perhaps even a strain for the study of autism features [35, 36] (Figure 8Bottom panel). We find that other strains such as I/LnJ, C57/L, and MOLF/EiJ show increased levels of escape behaviors, thus increasing potential strains that could be used to model this behavior.

In addition to phenotypic diversity due to genotype, we explored sexual dimorphism in our dataset with these new classifiers. We examined the impact of sex on the aggregated phenotype in various strains using a univariate approach. To test for the statistical significance of the effect of sex, we utilized a nonparametric rank test and correcting for multiple testing using false discovery rate (fdr) using Benajamini-Hochberg method. The LOD scores and effect sizes are presented in Figure 9B, with the left panel showing the strength of evidence against the null hypothesis of the non-sex effect. The right panel presents a representation of the direction and magnitude of the effect size with the color and size of circle represents the direction and magnitude of the effect, respectively. The strains highlighted in pink exhibit a significant sex effect for at least one of the aggregated phenotypes. It is important to note that we are generally underpowered with five animals of each sex. However, we find that a high proportion of phenotypes show a sex effect.

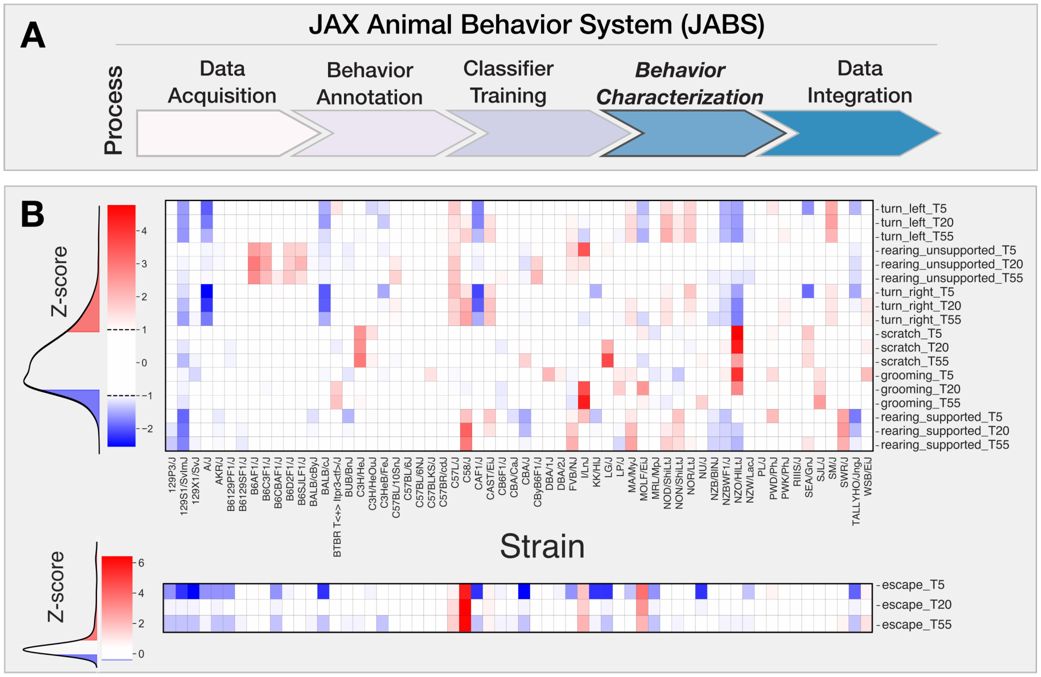


Figure 8: **JABS behavior characterization module: Aggregated phenotypes for behaviors using our large strain survey**. (A) JABS pipeline highlighting individual steps towards automated behavioral quantification. (B) Z-transformed scores for the total duration of behavior (at 5, 20, 55 mins) for each aggregate phenotype ($|z - \text{score}| > 1$; thresholding is applied for all the behaviors except escape).

3 Discussion

Recently developed methods of automated behavior detection have great potential for use in behavioral genetics research by allowing researchers to analyze large datasets in a relatively short amount of time [11, 27, 37]. This can be particularly useful for identification of genetic and environmental factors that influence behavior. While these methods have shown promising results in automated quantification of animal behavior, they require a high level of technical expertise. Moreover, these methods often require large amounts of training data, which can be difficult to obtain for researchers working with limited resources. Community value and adoption of these methods requires a commitment to develop and support a new ecosystem of behavior analysis, especially for use by behavioral researchers without computational expertise and resources. Another challenge to this approach is the lack of data standardization, without which it is difficult to compare and share the classifiers across different studies and labs. Therefore, a standardized data collection apparatus would benefit the community by easing the sharing of techniques, data, and classifiers which facilitates comparison of results across different studies and build on previous work. Here we have shown that our apparatus provides robust high quality video data under differing experimental conditions, mice, and lengths of time. In addition to the standardized data acquisition, we have developed a user-friendly GUI tool that provides a visual interface for annotating the videos, train a decision tree based classifiers for different behaviors. This tool also allows researchers to download the existing trained classifier, which can be shared and used by other researchers.

Even when data acquisition is standardized, another fundamental source of variability can enter the system when different human experts within or across different labs, annotate the same videos for the same behavior. This type of variability can arise due to variety of factors, including differences in training, personal biases, and individual interpretation of behavior. For instance, one annotator may view a particular behavior as aggressive, while another may view it as playful. In section 2.4.1, we

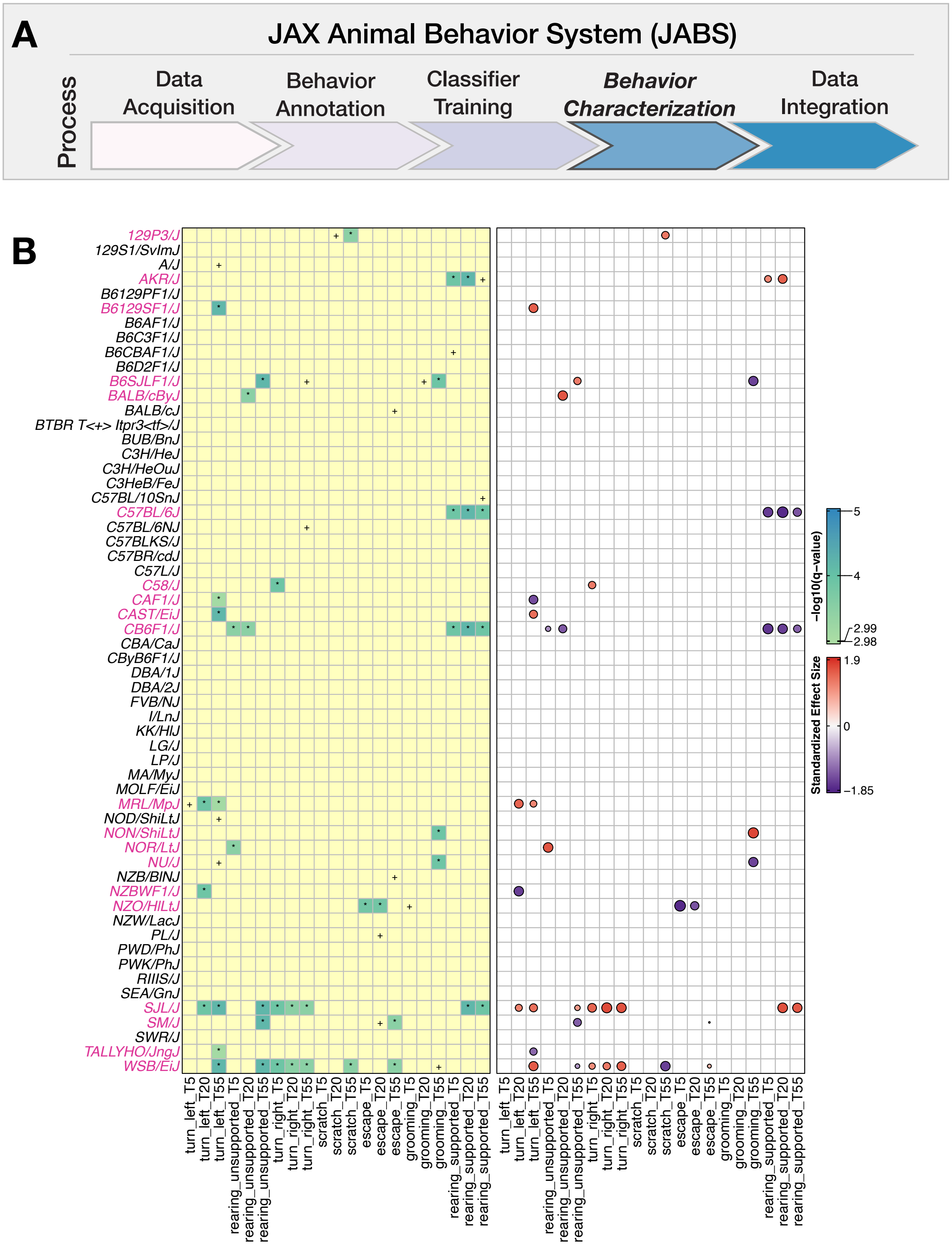


Figure 9: **JABS behavior characterization module: Univariate analysis captures the combined effect of sex and strain on the aggregate phenotype**: (A) JABS pipeline highlighting individual steps towards automated behavioral quantification. (B) The LOD scores $(-log_{10}(q_{value}))$ and effect sizes are shown at left and right panels, respectively. In the left panel, the number of *s represents the strength of evidence against the null hypothesis of no sex effect, while + represents a suggestive effect. In the right panel, the color (red for female and blue for male) and area of the circle (area being proportional to the size of the effect) represent the direction and magnitude of the effect size. Strains with a sex difference in at least one of the aggregated phenotypes are colored pink.

have demonstrated that even for simple behaviors like left and right turn, there is a significant amount of disagreement between predictions coming from classifiers trained by two expert annotators within the lab for the same behavior. One of the most commonly used statistical measure to quantify the inter-annotator variability is Cohen's kappa which assesses the level of agreement between the annotators taking into account the possibility of agreement by chance. The Cohen's kappa statistic work well for frame-wise comparison but is ill-defined for bout-wise comparison as unlike frames, bouts are not conserved. In order to overcome this limitation, we have introduced a new approach based on graph theory, called the *ethograph*. This network approach allows us to define measures that quantify the agreement between two annotators when comparing bouts of behavior among different annotators. By comparing the entire sequences of frames, the ethograph reduces subjectivity and allows for a more holistic and consistent interpretation of behaviors. This makes it well-suited for bout-wise comparison, and may provide a more accurate estimate of inter-annotator agreement than the frame-based kappa statistic.

3.1 Future directions and challenges

One potential approach to understanding the source of inter-annotator variability is to compare SHAP [38] scores for top features associated with classifier. Briefly, SHAP (SHapley Additive exPlanations) is a game-theoretic approach to explain the predictions made by a machine learning model [39]. SHAP can be used to assess the reliability of human labelers by comparing the SHAP values for the same feature across different labelers. If the SHAP values for a particular feature are consistently high across different labelers, it indicates that this feature is a strong predictor of the classification decision and that the labelers are consistently applying the same criteria. On the other hand, if the SHAP values for a particular feature vary widely across different labelers, it indicates that there is inconsistency in the application of the classification criteria and that the reliability of the human labelers may be questionable. For instance, in case of left turn behavior annotations, if one annotator consistently places higher importance on the head orientation feature than the other annotator, this would imply that this annotator has a different understanding of what constitutes the start of a left turn behavior compared to the other annotator. Therefore, depending on the quality of reliability, the researcher may decide to provide additional training to the annotators on how to interpret certain features in the pose estimate data.

Furthermore, since the the classifiers are trained on few densely labeled short video recordings and then further make predictions on a large strain survey consisting of multiple strains of mice, there is some variability in predictions purely due to out-of-distribution strains in the strain survey. Therefore, the inter-annotator variability in predictions on the new set of strains of mice can be attributed to both the variability in the human labeling and genetic variability in the strain survey. Calculating the heritability scores might help in this scenario by providing us a quantitative measure of the extent to which the inter-annotator variability is due to genetic factors versus interpretation by the human labelers.

4 Acknowledgments

We thank members of the Kumar Lab for helpful advice, Michelle Foskett of JAX PQC for the loan of Sensidyne and MadgeTech equipment, Rosalinda Doty for histological analysis. This work was funded by The Jackson Laboratory Directors Innovation Fund, National Institute of Health DA041668 (NIDA) and DA048634 (NIDA). All code and training data will be available at Kumarlab.org and Kumar Lab Github (https://github.com/KumarLabJax/JABS-data-pipeline).

References

1. Gomez-Marin, A., Paton, J. J., Kampff, A. R., Costa, R. M. & Mainen, Z. F. Big behavioral data: psychology, ethology and the foundations of neuroscience. *Nature neuroscience* **17,** 1455–1462 (2014).
2. Datta, S. R., Anderson, D. J., Branson, K., Perona, P. & Leifer, A. Computational neuroethology: a call to action. *Neuron* **104,** 11–24 (2019).
3. LeCun, Y., Bottou, L., Bengio, Y. & Haffner, P. Gradient-based learning applied to document recognition. *Proceedings of the IEEE* **86,** 2278–2324 (1998).
4. Krizhevsky, A., Sutskever, I. & Hinton, G. E. Imagenet classification with deep convolutional neural networks. *Communications of the ACM* **60,** 84–90 (2017).
5. LeCun, Y., Bengio, Y. & Hinton, G. Deep learning. *nature* **521,** 436–444 (2015).
6. He, K., Zhang, X., Ren, S. & Sun, J. *Deep residual learning for image recognition* in *Proceedings of the IEEE conference on computer vision and pattern recognition* (2016), 770–778.
7. Schmidhuber, J. Deep learning in neural networks: An overview. *Neural networks* **61,** 85–117 (2015).
8. Ziegler, L., Sturman, O. & Bohacek, J. Big behavior: challenges and opportunities in a new era of deep behavior profiling. *Neuropsychopharmacology* **46** (June 2020).
9. Geuther, B. Q. *et al.* Robust mouse tracking in complex environments using neural networks. *Communications biology* **2,** 124 (2019).
10. Sheppard, K. *et al.* Stride-level analysis of mouse open field behavior using deep-learning-based pose estimation. *Cell Reports* **38,** 110231 (2022).
11. Geuther, B. Q. *et al.* Action detection using a neural network elucidates the genetics of mouse grooming behavior. *Elife* **10,** e63207 (2021).
12. Hession, L. E., Sabnis, G., Churchill, G. A. & Kumar, V. A machine vision based frailty index for mice. *bioRxiv* (2021).
13. Kumar, V. *et al.* Second-generation high-throughput forward genetic screen in mice to isolate subtle behavioral mutants. *Proceedings of the National Academy of Sciences* **108,** 15557–15564. ISSN: 0027-8424 (2011).
14. Council, N. R. *et al.* Guide for the care and use of laboratory animals (2011).
15. Myatt, T. A. *et al.* A study of indoor carbon dioxide levels and sick leave among office workers. *Environmental Health* **1,** 1–10 (2002).
16. Mexas, A. M., Brice, A. K., Caro, A. C., Hillanbrand, T. S. & Gaertner, D. J. Nasal histopathology and intracage ammonia levels in female groups and breeding mice housed in static isolation cages. *Journal of the American Association for Laboratory Animal Science* **54,** 478–486 (2015).
17. Fawcett, A. & Rose, M. Guidelines for the housing of mice in scientific institutions. *Animal Welfare Unit, NSW Department of Primary Industries, West Pennant Hills. Anim Res Rev Panel* **1,** 1–43 (2012).
18. Gamble, M. & Clough, G. Ammonia build-up in animal boxes and its effect on rat tracheal epithelium. *Laboratory Animals* **10,** 93–104 (1976).
19. Ferrecchia, C. E., Jensen, K. & Van Andel, R. Intracage ammonia levels in static and individually ventilated cages housing C57BL/6 mice on 4 bedding substrates. *Journal of the American Association for Laboratory Animal Science* **53,** 146–151 (2014).
20. Tepper, J. S., Weiss, B. & Wood, R. W. Alterations in behavior produced by inhaled ozone or ammonia. *Fundamental and Applied Toxicology* **5,** 1110–1118 (1985).
21. Easterly, M. E., Foltz, J. & Paulus, M. J. Body condition scoring: comparing newly trained scorers and micro-computed tomography imaging. *LAB ANIMAL-NEW YORK-* **30,** 46–49 (2001).
22. Hickman, D. L. & Swan, M. Use of a body condition score technique to assess health status in a rat model of polycystic kidney disease. *Journal of the American Association for Laboratory Animal Science* **49,** 155–159 (2010).

23. Lovasz, R. M., Marks, D. L., Chan, B. K. & Saunders, K. E. Effects on Mouse Food Consumption After Exposure to Bedding from Sick Mice or Healthy Mice. *Journal of the American Association for Laboratory Animal Science* **59,** 687–694 (2020).
24. Green, E. L. Biology of the laboratory mouse (1966).
25. Ho, T. K. *Random decision forests* in *Proceedings of 3rd international conference on document analysis and recognition* **1** (1995), 278–282.
26. Chen, T. & Guestrin, C. *Xgboost: A scalable tree boosting system* in *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining* (2016), 785–794.
27. Segalin, C. *et al.* The Mouse Action Recognition System (MARS) software pipeline for automated analysis of social behaviors in mice. *Elife* **10** (2021).
28. Mathis, A. *et al.* DeepLabCut: markerless pose estimation of user-defined body parts with deep learning. *Nature Neuroscience* **21** (Sept. 2018).
29. Kabra, M., Robie, A. A., Rivera-Alba, M., Branson, S. & Branson, K. JAABA: interactive machine learning for automatic annotation of animal behavior. *Nature methods* **10,** 64 (2013).
30. Kaufman, A. B. & Rosenthal, R. Can you believe my eyes? The importance of interobserver reliability statistics in observations of animal behaviour. *Animal Behaviour* **78,** 1487–1491 (2009).
31. Tjandrasuwita, M., Sun, J. J., Kennedy, A., Chaudhuri, S. & Yue, Y. Interpreting expert annotation differences in animal behavior. *arXiv preprint arXiv:2106.06114* (2021).
32. McHugh, M. L. Interrater reliability: the kappa statistic. *Biochemia medica* **22,** 276–282 (2012).
33. Feichtenhofer, C., Fan, H., Malik, J. & He, K. *Slowfast networks for video recognition* in *Proceedings of the IEEE International Conference on Computer Vision* (2019), 6202–6211.
34. Kalogeiton, V., Weinzaepfel, P., Ferrari, V. & Schmid, C. *Action tubelet detector for spatio-temporal action localization* in *Proceedings of the IEEE International Conference on Computer Vision* (2017), 4405–4413.
35. Ryan, B. C., Young, N. B., Crawley, J. N., Bodfish, J. W. & Moy, S. S. Social deficits, stereotypy and early emergence of repetitive behavior in the C58/J inbred mouse strain. *Behavioural Brain Research* **208,** 178–188. ISSN: 0166-4328. https://www.sciencedirect.com/science/article/pii/S0166432809007086 (2010).
36. Blick, M. G. *et al.* Novel object exploration in the C58/J mouse model of autistic-like behavior. *Behavioural Brain Research* **282,** 54–60. ISSN: 0166-4328. https://www.sciencedirect.com/science/article/pii/S0166432814008249 (2015).
37. Nilsson, S. R. *et al.* Simple Behavioral Analysis (SimBA)–an open source toolkit for computer classification of complex social behaviors in experimental animals. *BioRxiv* (2020).
38. 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 (2022).
39. Lundberg, S. M. & Lee, S.-I. A unified approach to interpreting model predictions. *Advances in neural information processing systems* **30** (2017).

5 Appendix

	Annotated Frames	
	Behavior	Not Behavior
10	11778	18201
20	19540	28777
50	43022	51466
100	76109	104889
500	399155	595798
1100 (All)	863394	1318396

Table 1: Data used for grooming benchmark

5.1 List of features for JABS

Table 2: List of JABS features

No.	Features	Type
1	ANGLE NOSE-BASE_NECK-LEFT_FRONT_PAW	base
2	ANGLE RIGHT_FRONT_PAW-BASE_NECK-CENTER_SPINE	base
3	ANGLE LEFT_FRONT_PAW-BASE_NECK-CENTER_SPINE	base
4	ANGLE BASE_NECK-CENTER_SPINE-BASE_TAIL	base
5	ANGLE RIGHT_REAR_PAW-BASE_TAIL-CENTER_SPINE	base
6	ANGLE LEFT_REAR_PAW-BASE_TAIL-CENTER_SPINE	base
7	ANGLE RIGHT_REAR_PAW-BASE_TAIL-MID_TAIL	base
8	ANGLE LEFT_REAR_PAW-BASE_TAIL-MID_TAIL	base
9	ANGLE CENTER_SPINE-BASE_TAIL-MID_TAIL	base
10	ANGLE BASE_TAIL-MID_TAIL-TIP_TAIL	base
11	ANGULAR_VELOCITY	base
12	BASE TAIL VELOCITY DIRECTION	base
13	BASE TAIL VELOCITY MAGNITUDE	base
14	CENTROID_VELOCITY_DIR	base
15	CENTROID_VELOCITY_MAG	base
16	LEFT FRONT PAW VELOCITY DIRECTION	base
17	LEFT FRONT PAW VELOCITY MAGNITUDE	base
18	NOSE VELOCITY DIRECTION	base
19	NOSE VELOCITY MAGNITUDE	base
20	NOSE-LEFT_EAR	base
21	NOSE-RIGHT_EAR	base
22	NOSE-BASE_NECK	base
23	NOSE-LEFT_FRONT_PAW	base
24	NOSE-RIGHT_FRONT_PAW	base
25	NOSE-CENTER_SPINE	base
26	NOSE-LEFT_REAR_PAW	base
27	NOSE-RIGHT_REAR_PAW	base
28	NOSE-BASE_TAIL	base
29	NOSE-MID_TAIL	base

Table 2: List of JABS features

No.	Features	Type
30	NOSE-TIP_TAIL	base
31	LEFT_EAR-RIGHT_EAR	base
32	LEFT_EAR-BASE_NECK	base
33	LEFT_EAR-LEFT_FRONT_PAW	base
34	LEFT_EAR-RIGHT_FRONT_PAW	base
35	LEFT_EAR-CENTER_SPINE	base
36	LEFT_EAR-LEFT_REAR_PAW	base
37	LEFT_EAR-RIGHT_REAR_PAW	base
38	LEFT_EAR-BASE_TAIL	base
39	LEFT_EAR-MID_TAIL	base
40	LEFT_EAR-TIP_TAIL	base
41	RIGHT_EAR-BASE_NECK	base
42	RIGHT_EAR-LEFT_FRONT_PAW	base
43	RIGHT_EAR-RIGHT_FRONT_PAW	base
44	RIGHT_EAR-CENTER_SPINE	base
45	RIGHT_EAR-LEFT_REAR_PAW	base
46	RIGHT_EAR-RIGHT_REAR_PAW	base
47	RIGHT_EAR-BASE_TAIL	base
48	RIGHT_EAR-MID_TAIL	base
49	RIGHT_EAR-TIP_TAIL	base
50	BASE_NECK-LEFT_FRONT_PAW	base
51	BASE_NECK-RIGHT_FRONT_PAW	base
52	BASE_NECK-CENTER_SPINE	base
53	BASE_NECK-LEFT_REAR_PAW	base
54	BASE_NECK-RIGHT_REAR_PAW	base
55	BASE_NECK-BASE_TAIL	base
56	BASE_NECK-MID_TAIL	base
57	BASE_NECK-TIP_TAIL	base
58	LEFT_FRONT_PAW-RIGHT_FRONT_PAW	base
59	LEFT_FRONT_PAW-CENTER_SPINE	base
60	LEFT_FRONT_PAW-LEFT_REAR_PAW	base
61	LEFT_FRONT_PAW-RIGHT_REAR_PAW	base
62	LEFT_FRONT_PAW-BASE_TAIL	base
63	LEFT_FRONT_PAW-MID_TAIL	base
64	LEFT_FRONT_PAW-TIP_TAIL	base
65	RIGHT_FRONT_PAW-CENTER_SPINE	base
66	RIGHT_FRONT_PAW-LEFT_REAR_PAW	base
67	RIGHT_FRONT_PAW-RIGHT_REAR_PAW	base
68	RIGHT_FRONT_PAW-BASE_TAIL	base
69	RIGHT_FRONT_PAW-MID_TAIL	base
70	RIGHT_FRONT_PAW-TIP_TAIL	base
71	CENTER_SPINE-LEFT_REAR_PAW	base
72	CENTER_SPINE-RIGHT_REAR_PAW	base
73	CENTER_SPINE-BASE_TAIL	base
74	CENTER_SPINE-MID_TAIL	base
75	CENTER_SPINE-TIP_TAIL	base

Table 2: List of JABS features

No.	Features	Type
76	LEFT_REAR_PAW-RIGHT_REAR_PAW	base
77	LEFT_REAR_PAW-BASE_TAIL	base
78	LEFT_REAR_PAW-MID_TAIL	base
79	LEFT_REAR_PAW-TIP_TAIL	base
80	RIGHT_REAR_PAW-BASE_TAIL	base
81	RIGHT_REAR_PAW-MID_TAIL	base
82	RIGHT_REAR_PAW-TIP_TAIL	base
83	BASE_TAIL-MID_TAIL	base
84	BASE_TAIL-TIP_TAIL	base
85	MID_TAIL-TIP_TAIL	base
86	NOSE POINT MASK	base
87	LEFT_EAR POINT MASK	base
88	RIGHT_EAR POINT MASK	base
89	BASE_NECK POINT MASK	base
90	LEFT_FRONT_PAW POINT MASK	base
91	RIGHT_FRONT_PAW POINT MASK	base
92	CENTER_SPINE POINT MASK	base
93	LEFT_REAR_PAW POINT MASK	base
94	RIGHT_REAR_PAW POINT MASK	base
95	BASE_TAIL POINT MASK	base
96	MID_TAIL POINT MASK	base
97	TIP_TAIL POINT MASK	base
98	NOSE SPEED	base
99	LEFT_EAR SPEED	base
100	RIGHT_EAR SPEED	base
101	BASE_NECK SPEED	base
102	LEFT_FRONT_PAW SPEED	base
103	RIGHT_FRONT_PAW SPEED	base
104	CENTER_SPINE SPEED	base
105	LEFT_REAR_PAW SPEED	base
106	RIGHT_REAR_PAW SPEED	base
107	BASE_TAIL SPEED	base
108	MID_TAIL SPEED	base
109	TIP_TAIL SPEED	base
110	RIGHT FRONT PAW VELOCITY DIRECTION	base
111	RIGHT FRONT PAW VELOCITY MAGNITUDE	base

Classifier	Window Size	F1 score
Escape	2	0.858
Grooming	60	
Rearing supported	5	0.91
Rearing unsupported	10	0.73
Scratch	5	0.798
Turn Left (Annotator-1)	5	0.73
Turn Left (Annotator-2)	5	0.95
Turn Right (Annotator-1)	20	0.89
Turn Right (Annotator-2)	5	0.88

Table 3: Classifiers trained by JABS with their respective window sizes and F1 scores

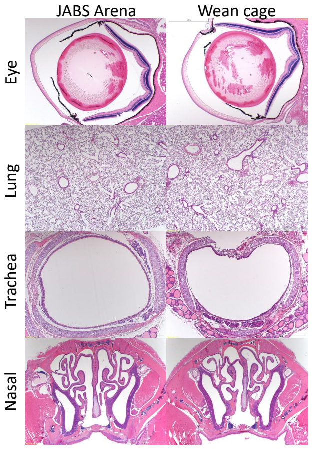


Figure 10: Representative hematoxylin and eosin (H&E) stained tissue sections from mice after spending 14 days in the JABS arena or control wean cage. Tissues selected for examination (eye, lung, trachea and nasal passages) are those expected to be most affected if the mice lived in a space with inadequate air flow. All tissues appeared normal.