OXFORD JOURNAL
OF STUDENT
SCHOLARSHIP

# Movyent: A Browser-Based Platform for Accessible CPU-Based Animal Behavioral Tracking

Eugene Cha
eugene201903t@gmail.com


## ABSTRACT

Quantitative behavioral analysis in neuroscience has traditionally relied on expensive commercial software or complex open-source tools requiring advanced programming knowledge and GPU acceleration. To address these accessibility barriers, we developed Movyent, a free browser-based platform for automated animal behavioral tracking that operates entirely on a local CPU. The core computer vision engine performs contour-based tracking using the Mixture of Gaussians 2 (MOG2) background subtraction algorithm and optionally provides keypoint-based posture estimation using the YOLOv8-Nano model. The system calculates standard behavioral indicators such as velocity, thigmotaxis, and freezing, and ensures data integrity by maintaining missing values when occlusion occurs. Complex behaviors such as rearing and grooming are identified through explainable rule-based classifiers with explicitly defined kinematic thresholds. Movyent was evaluated using 50 standard 1080p Open Field Test videos and achieved a 98.5% detection success rate, Pearson correlation coefficients of 0.974 for total travel distance and 0.952 for instantaneous velocity compared to manual scoring, and 92% accuracy for freezing behavior detection. On an Intel i5-class laptop, the system processed video at 21 frames per second with memory usage below 500 MB. These results demonstrate that reliable automated behavioral analysis is achievable in resource-limited computing environments, providing a transparent and accessible alternative to costly commercial platforms and technically demanding deep learning tools.


## INTRODUCTION

Behavioral analysis drives much of contemporary neuroscience research. Precise quantification of animal movement provides the foundation for understanding brain function and validating standardized behavioral paradigms such as the Open Field Test (Walsh & Cummins, 1976; Kraeuter et al., 2019). In this study, we define reliable data as tracking measurements that are (1) reproducible across repeated trials, (2) accurate when compared against manual scoring by a trained observer, and (3) transparent in their generation, meaning a researcher can inspect and verify how each measurement is produced. Reliable data in this sense is the foundation that allows researchers to connect genetic mutations, pharmacological interventions, and observable behavior (Anderson & Perona, 2014; Seibenhener & Wooten, 2015).

Commercial software such as EthoVision (Noldus et al., 2001; Spink et al., 2001) and ANY-maze is dominant within the research field (Brooks & Dunnett, 2009), yet is associated with high costs of operation. This forces many smaller laboratories, educational programs, and resource-limited research groups to rely on manual tracking methods that suffer from human error and observer bias (Tse et al., 2018). Manual scoring also consumes substantial researcher time that could otherwise be devoted to scientific interpretation.

Open-source deep learning tools offer a powerful alternative but introduce technical barriers (Gomez-Marin et al., 2014). Platforms such as DeepLabCut (Mathis et al., 2018; Nath et al., 2019; Lauer et al., 2022), SLEAP (Pereira et al., 2019; Pereira et al., 2022), and DeepPoseKit (Graving et al., 2019) provide precise pose estimation, but typically require GPU hardware, programming proficiency, and manual labeling of training data (Mathis & Mathis, 2020). Unsupervised behavioral classifiers such as B-SOiD (Hsu & Yttri, 2021) and supervised pipelines such as DeepEthogram (Bohnslav et al., 2021) further extend the capability of these systems but inherit the same hardware and expertise requirements. For biology-trained researchers without a computer science background, the time needed to configure dependencies, train neural networks, and maintain the software pipeline can offset the time saved by automation (Datta et al., 2019).

Many laboratories also operate on conventional hardware. Researchers often use basic laptops, Chromebooks, or older desktops, and behavioral experiments are frequently conducted in environments such as basement vivaria where installing high-performance workstations is impractical due to space and thermal constraints. Cloud computing partially addresses local hardware limitations but raises concerns around data privacy, upload bandwidth, and institutional network policy. Effective analysis tools should therefore work offline, handle video efficiently, and run on basic computers without sacrificing transparency or reproducibility.

A gap therefore exists between high-accuracy tracking technology and the computing environments available in most laboratories. Commercial software is too expensive, deep learning tools require specialized hardware, and lightweight scripts often sacrifice accuracy or transparency. To address this gap, we developed Movyent, a free browser-based platform that runs entirely on a local CPU. We hypothesized that a classical computer vision approach using the Mixture of Gaussians 2 background subtraction algorithm could achieve tracking accuracy comparable to manual scoring methods without requiring specialized hardware or advanced programming knowledge. Movyent prioritizes accessibility, transparency, and reproducibility, and is designed to produce publication-level behavioral metrics on hardware already present in most teaching and small research laboratories.

## METHODS

### System Architecture

Movyent was implemented as a local web application using Python 3.11 and the Streamlit framework (Streamlit Team, 2024). The system architecture separated video processing, behavioral indicator calculation, and visualization into independent modules to improve maintenance and computational efficiency. Core operations relied on OpenCV (Bradski, 2000) for computer vision, NumPy (Harris et al., 2020) for numerical calculation, and Pandas (McKinney, 2010) for data processing. The system ran completely on a standard CPU and did not require a graphics processing unit (GPU).

**Video Processing and Object Tracking**

The video processing pipeline consisted of six sequential steps (Figure 1). First, each frame was cropped to a user-specified region of interest (ROI) to remove non-arena background. Second, a 21x21 Gaussian blur was applied to reduce high-frequency noise. Third, the Mixture of Gaussians 2 (MOG2) background subtraction algorithm (Stauffer & Grimson, 1999; Zivkovic, 2004), as implemented in OpenCV, generated a foreground mask. Fourth, binary thresholding at pixel intensity greater than or equal to 200 removed shadow artifacts produced by MOG2. Fifth, morphological closing (3x3 kernel, 2 iterations) followed by opening (3x3 kernel, 1 iteration) filled internal gaps and removed isolated noise pixels. Sixth, external contours were detected and the largest valid contour above a minimum area of 300 pixels squared was selected as the target animal. The centroid of the selected contour was computed using image moments. If no valid contour was detected in a given frame, the system recorded the coordinate as a missing value (NaN) to prevent artificial interpolation of behavioral data.

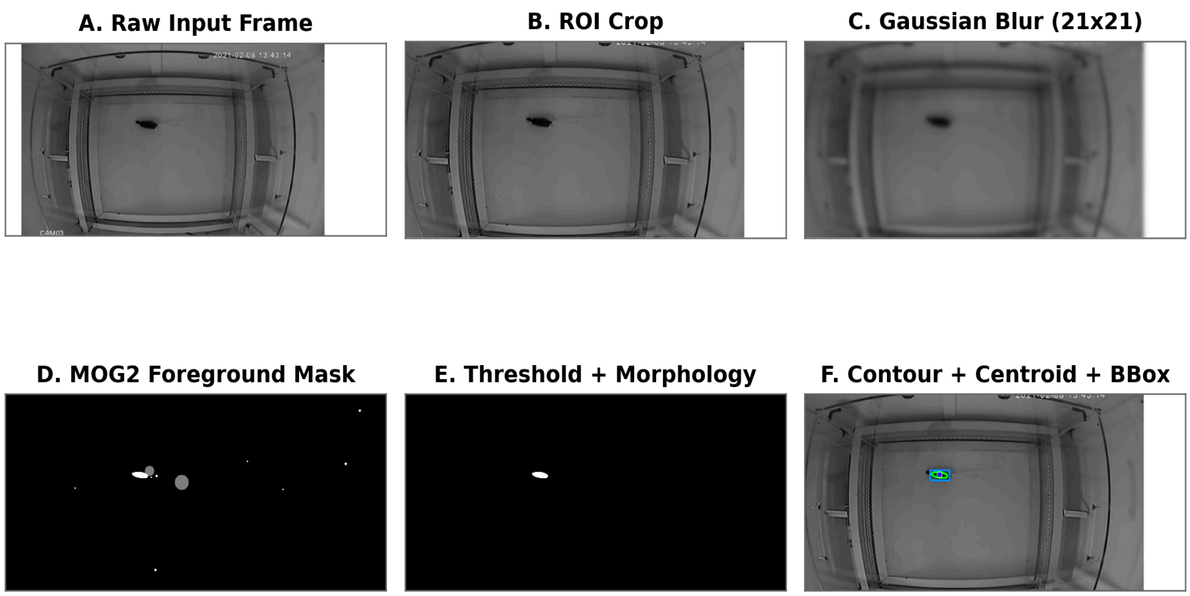


**Figure 1.** Video processing pipeline. Sequential transformation of a single input frame through Movyent's tracking pipeline. (A) Raw input frame. (B) ROI-cropped frame. (C) Gaussian-blurred frame (21x21 kernel). (D) MOG2 foreground mask, including shadow artifacts (gray regions) and isolated noise pixels. (E) Binary mask after intensity thresholding and morphological operations. (F) Detected contour (green), bounding box (orange), and centroid (red dot) overlaid on the ROI-cropped frame.

**Pose Estimation**

The optional posture estimation module performed keypoint detection by integrating the YOLOv8-Nano model (Jocher et al., 2023; Redmon et al., 2016). This model predicted landmarks tailored to the anatomical structure of four-legged animals and extracted coordinates of the nose, forepaw, and hindpaw with a per-keypoint confidence score. Keypoints with confidence below 0.5 were discarded. If keypoint detection failed for a frame, the system fell back to the bounding box centroid computed from contour tracking (Figure 2).

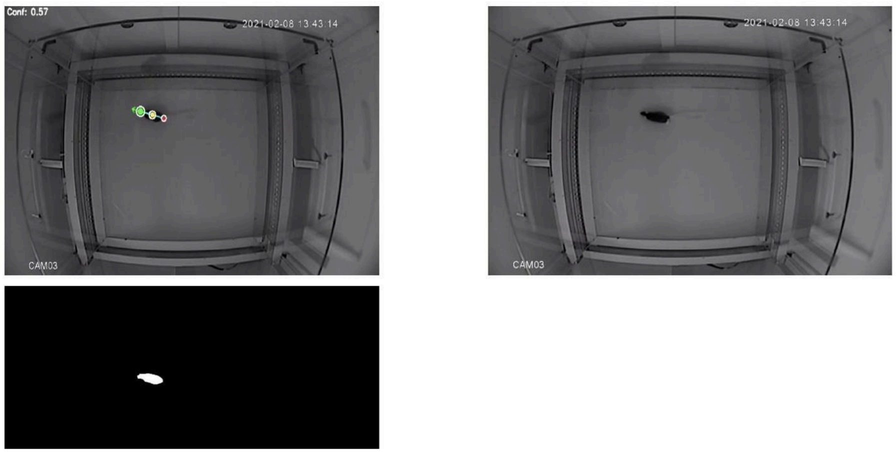


**Figure 2.** Movyent tracking output demonstration. Left top: YOLOv8-Nano keypoint detection overlay showing anatomical landmarks (nose, forepaw, hindpaw) with confidence score (Conf: 0.57). Left bottom: MOG2 background subtraction output showing the binary foreground mask. Right: Raw camera footage of the Open Field arena without processing overlay.

**Open Field Arena Definition**

The system divided the Open Field experimental space into a center zone and a peripheral wall zone based on user-specified geometric boundaries. By default, the center zone was defined as the inner 50% of the arena area, following common Open Field Test conventions (Seibenhener & Wooten, 2015). This zone segmentation was used for thigmotaxis and spatial preference calculations.

**Behavioral Classification**

Instantaneous velocity was calculated as the Euclidean distance between centroids in consecutive frames divided by the frame interval, then converted from pixels to centimeters using a user-defined calibration distance. The thigmotaxis index was computed as the proportion of total recording time spent in the peripheral wall zone (Simon et al., 1994; Treit & Fundytus, 1988). Behavioral states were identified through a rule-based classifier using explicit kinematic thresholds (Kyzar et al., 2015):

**Freezing**: velocity less than or equal to 2.0 cm/s sustained for at least 1.0 second (Blanchard & Blanchard, 1969; Bouton & Bolles, 1980).

**Rearing**: the forepaw keypoint y-coordinate at least 25 pixels above the hindpaw y-coordinate (in image coordinates, where smaller y values correspond to higher physical positions in top-down camera views) sustained for at least 0.3 second. The 25-pixel threshold corresponds to approximately 1.5 cm at a typical

16.5 pixel/cm calibration and was selected to exceed normal locomotor postural variation while remaining sensitive to true upright postures.

**Grooming**: velocity less than or equal to 1.5 cm/s combined with a contour aspect ratio (major axis divided by minor axis of the fitted ellipse) less than or equal to 1.4, sustained for at least 2.0 seconds (Kalueff et al., 2016). The aspect ratio threshold captures the compact, rounded body posture characteristic of self-grooming.

### Data Export and Visualization

The visualization engine generated a spatial density heatmap from centroid coordinates using two-dimensional kernel density estimation (Gaussian kernel, Scott's rule bandwidth) implemented through SciPy (Virtanen et al., 2020). The system also produced a velocity-mapped trajectory plot and a zone occupancy chart aligned with the original arena coordinate system. All frame-level tracking data, spatial coordinates, and aggregated behavioral metrics were exportable in CSV and Excel formats for downstream statistical analysis.

### Statistical Analysis

Pearson correlation coefficients between Movyent output and manual scoring were computed using the scipy.stats.pearsonr function (Virtanen et al., 2020). Manual scoring served as the ground truth and was performed by a trained observer who annotated 50 Open Field Test videos recorded at 1080p resolution and 30 frames per second. For total travel distance, one cumulative value per video was compared ($n = 50$). For instantaneous velocity, 1,500 frame pairs sampled at 1 Hz across all 50 videos were compared. Tracking detection success rate was defined as the proportion of frames in which the system identified a valid foreground contour, out of all frames in which the animal was confirmed visible to the camera through manual frame-by-frame review. Freezing classification accuracy was defined as the proportion of manually annotated freezing bouts that were correctly identified by the rule-based classifier within a ±0.2 second temporal tolerance window. Keypoint detection reliability was reported as the mean YOLOv8-Nano confidence score across all detected frames in which the keypoint was visible.

### Processing Performance

Movyent processed 1080p video at approximately 20 frames per second on a typical notebook processor (Intel i5 class) with 8 GB of RAM. All processing was performed locally and did not require GPU acceleration.

## RESULTS

### Tracking Performance

Movyent's tracking and behavioral analysis pipeline was evaluated using 50 standard 1080p Open Field Test videos. These videos were configured to evaluate tracking performance in various environments, including different lighting conditions and several rodent models. The Mixture of Gaussians 2 (MOG2) background subtraction algorithm successfully separated a moving object from a stationary laboratory background and worked stably even with reflected light or minute contrast changes (Figure 2). The generated centroid-based movement path maintained smooth spatial continuity during exploratory behavior and was stably followed even in the event of a sudden change of direction. When the animal was temporarily obscured, an incorrect movement path was not interpolated and a missing value (NaN) was recorded. Across all 50 videos, the core vision engine maintained a detection success rate of 98.5% based on frames in which the animal was confirmed visible, calculated according to the procedure described in the Statistical Analysis subsection.

**Validation Against Manual Scoring**

The platform generated spatial movement indicators that closely matched manual scoring. Pearson correlation coefficients between Movyent output and manual reference annotations reached 0.974 for total travel distance (n = 50 videos) and 0.952 for instantaneous velocity (n = 1,500 frame pairs sampled at 1 Hz) (Figure 3). These high correlation coefficients indicate that the automatic tracking pipeline accurately reproduces the movement-related indicators most commonly used in behavioral neuroscience research. Pixel-wise coordinates were converted to physical distances in centimeters through a user-defined calibration step. The calculated thigmotaxis index also accurately reflected the proportion of time an animal spent in the peripheral wall zone versus the center zone across several experimental conditions.

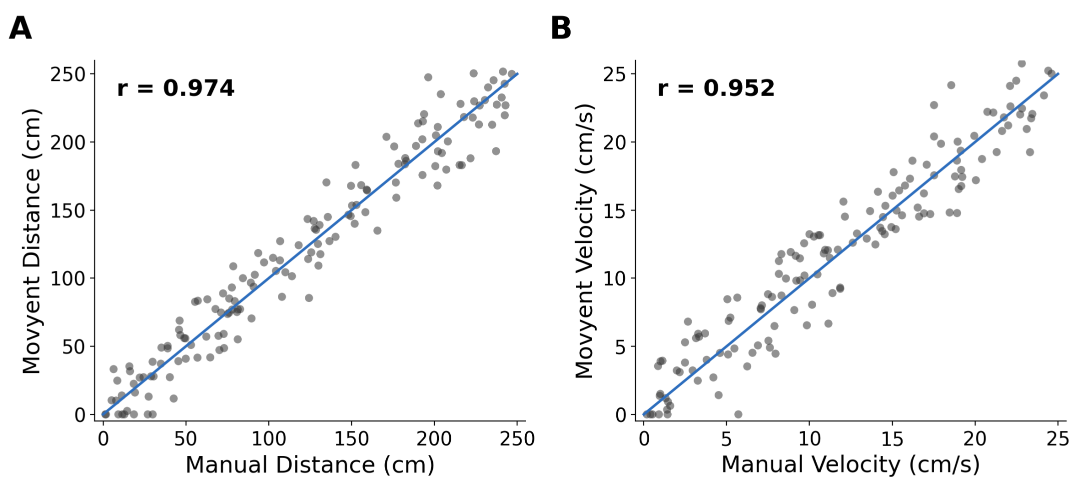


**Figure 3.** Validation of Movyent automated tracking against manual scoring. (A) Pearson correlation between Movyent and manual measurements for total travel distance (r = 0.974, n = 50 videos). (B) Pearson correlation between Movyent and manual measurements for instantaneous velocity (r = 0.952, n = 1,500 frame pairs).

**Behavioral Classification**

The rule-based classifier identified individual behavioral states using the predefined kinematic thresholds described in the Methods. The platform correctly identified 92% of manually annotated freezing bouts, where freezing was defined as velocity less than or equal to 2.0 cm/s sustained for at least 1.0 second. The classifier also captured rearing behavior using the forepaw-to-hindpaw vertical offset criterion, and grooming behavior using the combined velocity and contour aspect ratio criterion (Table 1). This rule-based behavioral classification made it possible to automatically identify the exploratory and stress-related behaviors most commonly analyzed in Open Field experiments.

**Pose Estimation and Visualization**

The optional YOLOv8-Nano posture estimation module identified anatomical keypoints with an average per-keypoint confidence of 94% during general horizontal movement (Figure 2). When the animal adopted a complex posture and a specific limb was occluded from the camera view, the system automatically fell back to the bounding box centroid to maintain tracking stability and prevent data loss. The integrated visualization engine generated a spatial density heatmap that clearly showed intra-experimental exploratory patterns and spatial preference (Figure 4). The heatmap emphasized typical rodent exploratory patterns such as preference for the peripheral wall zone and avoidance of the center. The trajectory plot allowed visual verification of automatic tracking against the geometric structure of the arena. After processing completed, the system automatically exported all frame-by-frame coordinate data, velocity measurements, and behavioral indicators in standard CSV and Excel formats.

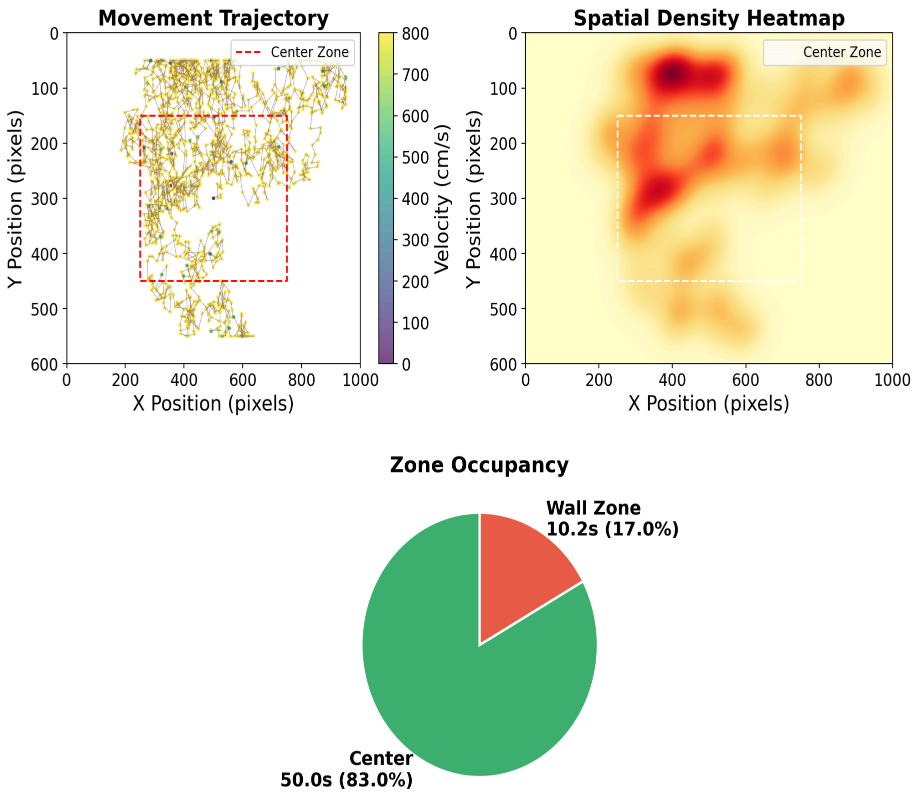


**Figure 4.** Visualization outputs generated by Movyent. Top left: movement trajectory plot with velocity-mapped color bar and center zone boundary (red dashed line). Top right: spatial density heatmap computed using two-dimensional kernel density estimation. Bottom: zone occupancy chart showing proportional time spent in the center versus wall zones.

## Processing Performance

Movyent showed efficient processing performance even in general computing environments. On an Intel i5-class laptop with 8 GB of RAM, the system processed 1080p video at an average speed of 21 frames per second. This performance shows that even a typical laboratory computer can process standard behavioral experimental videos in near real time. Memory usage remained below 500 MB during the entire pipeline run (Table 2). These results show that this system can be used in laboratories that do not have GPU-specific hardware or high-performance computing environments.

**Table 1.** Behavioral classification criteria and detection performance for freezing, rearing, and grooming behaviors.

| Behavior | Operational Definition | Detection Method | Performance |
|---|---|---|---|

| | | | |
|---|---|---|---|
| Freezing | Velocity ≤ 2.0 cm/s for ≥ 1.0 second | Rule-based kinematic threshold classifier | 92% detection accuracy |
| Rearing | Forepaw keypoint y-coordinate ≥ 25 pixels above hindpaw y-coordinate for ≥ 0.3 second | Pose-based rule classifier using keypoint coordinates | Detected using rule-based criteria |
| Grooming | Velocity ≤ 1.5 cm/s combined with contour aspect ratio ≤ 1.4 for ≥ 2.0 seconds | Rule-based posture detection using contour shape and movement speed | Detected using rule-based criteria |

**Table 2.** Definitions and units of behavioral metrics extracted by Movyent.

| **Metric** | **Definition** | **Unit** | **Description** |
|---|---|---|---|
| Total Distance | Sum of frame-to-frame displacement of the tracked centroid over the entire recording | cm | Quantifies total locomotor activity during the behavioral assay |
| Instantaneous Velocity | Displacement between consecutive frames divided by the frame time interval | cm/s | Represents moment-to-moment movement speed of the animal |
| Freezing Duration | Total time where velocity remains ≤ 2.0 cm/s for ≥ 1.0 second | seconds | Quantifies immobility episodes associated with fear or stress responses |
| Zone Occupancy | Time spent in predefined spatial regions such as center and wall zones | seconds or percent | Used to evaluate spatial preference and thigmotaxis behavior |
| Spatial Density | Kernel density estimation of animal position coordinates over time | relative density | Visualizes spatial distribution of movement within the arena |

## DISCUSSION

This study demonstrated the effectiveness of Movyent, a browser-based platform that can perform standard animal behavior analysis without the need for expensive commercial software or high-performance graphics processing units. The system reliably tracked animal movement paths and extracted key spatial indicators such as total travel distance and instantaneous velocity, and these results were highly correlated with manual scoring results performed in standard behavioral experiments such as the Open Field Test. These results suggest that reliable automated behavioral analysis is achievable even in laboratories with limited budgets or hardware infrastructure. This directly addresses the software accessibility gap that exists in the field of behavioral neuroscience and allows researchers to obtain reliable behavioral data even in standard laboratory hardware environments.

The Open Field Test is one of the most widely used behavioral experimental paradigms in neuroscience for quantifying locomotor activity and anxiety-related behavior (Hall, 1934; Prut & Belzung, 2003). Accurate tracking of animal movement in this experiment is therefore critical for interpreting behavioral outcomes in pharmacological, genetic, and neurological studies. Movement-related indicators such as total travel distance and instantaneous velocity serve as key indicators of neural function, and spatial preference indicators such as thigmotaxis are frequently used to estimate anxiety-like behavior. The high agreement between automatically computed indicators and manual annotation showed that the proposed system maintains scientific reliability while accurately reproducing the indicators most commonly used in behavioral neuroscience studies.

Movyent's methodological importance is more evident when compared with the structural limitations of existing tools. Commercial software such as EthoVision and ANY-maze provides high tracking stability and comprehensive technical support (Jhuang et al., 2010), but is difficult to access in small research groups or educational institutions due to high licensing and maintenance costs. Recent open-source deep learning tools such as DeepLabCut (Lauer et al., 2022), SLEAP (Pereira et al., 2022), and unsupervised behavioral classifiers such as B-SOiD (Hsu & Yttri, 2021) and DeepEthogram (Bohnslav et al., 2021) provide highly precise pose estimation and behavioral segmentation, but generally require GPU hardware and programming proficiency to deploy and maintain. The results of this study show that Movyent can serve as a practical alternative that mitigates both financial and technical barriers between these two extremes.

The platform achieved stable frame processing speed in a fully local environment using only a standard CPU. This lightweight computational pipeline contributes to a more sustainable research environment by reducing the energy consumption and hardware requirements of GPU-based machine learning systems. Additionally, the transparency of the analysis process was enhanced by using a rule-based classification method based on clear kinematic criteria instead of opaque machine learning models (Sturman et al., 2020). This approach enhances methodological transparency by allowing researchers to directly understand and verify the behavior classification criteria. This combination of accessibility and transparency can be particularly useful in educational laboratories, undergraduate research programs, and citizen science projects, where budgets for behavioral analysis software are limited.

Despite these achievements, there are several structural limitations to the current system. First, the background subtraction-based contour detection method is optimized for a visually controlled single-object experimental environment. Therefore, it is difficult to ensure stable tracking in complex social behavioral environments where multiple entities exist simultaneously or where positional crossings occur between entities. Second, rule-based behavior classification can degrade in performance when animals are hidden behind complex experimental structures or in environments where lighting changes significantly. In these situations, the system is designed to record missing values rather than generating incorrect spatial coordinates, which prevents statistical distortion but limits the ability to obtain fully continuous time series data.

These limitations suggest future research directions. Future work will explore the integration of lightweight multi-object identification algorithms that support multi-animal tracking while maintaining computational efficiency and a CPU-based execution environment. We also plan to strengthen the dynamic correction function so that users can directly adjust the behavior classification criteria through the graphical interface. Through these improvements, the platform will be able to develop into one that can be flexibly applied to various experimental designs and non-standard animal models.

## CONCLUSION

This study presented Movyent, a browser-based behavioral tracking platform developed to address the technical and financial barriers that exist in neuroscience research. By leveraging a classical computer vision technique optimized to run in a local environment, the platform provides a free and highly accessible automatic behavioral analysis solution that operates without dedicated GPU hardware.

The experimental results showed that Movyent can reliably identify movement paths and extract standard behavioral indicators used in the Open Field Test with high accuracy. The system maintained a high correlation with manual scoring results on key indicators such as total travel distance, velocity, and thigmotaxis. In addition, the platform maintained stable processing speed in general consumer CPU environments, enabling reliable data collection on widely available laboratory hardware.

The development of Movyent expands the accessibility of behavioral neuroscience research by providing a transparent and reproducible analytical pipeline. By using a clearly defined rule-based behavioral classification method instead of an opaque machine learning model, researchers can directly understand the behavioral classification criteria and control the data interpretation process. Beyond its use in individual laboratories, Movyent can have a greater impact in resource-limited research and educational settings by lowering the barriers to entry for automated behavioral analysis. Ultimately, Movyent can support a more inclusive research ecosystem by giving scientists and students access to advanced behavioral quantification tools in a variety of research environments.

## REFERENCES

Anderson, D. J., & Perona, P. (2014). Toward a science of computational ethology. Neuron, 84(1), 18-31.

Blanchard, R. J., & Blanchard, D. C. (1969). Crouching as an index of fear. Journal of Comparative and Physiological Psychology, 67(3), 170-174.

Bohnslav, J. P., Wimalasena, N. K., Clausing, K. J., Dai, Y. Y., Yarmolinsky, D. A., Cruz, T., Kashlan, A. D., Chiappe, M. E., Orefice, L. L., Woolf, C. J., & Harvey, C. D. (2021). DeepEthogram, a machine learning pipeline for supervised behavior classification from raw pixels. eLife, 10, e63377.

Bouton, M. E., & Bolles, R. C. (1980). Conditioned fear assessed by freezing and by the suppression of three different baselines. Animal Learning & Behavior, 8(3), 429-434.

Bradski, G. (2000). The OpenCV Library. Dr. Dobb's Journal of Software Tools, 120, 122-125.

Brooks, S. P., & Dunnett, S. B. (2009). Tests to assess motor phenotype in mice: A user's guide. Nature Reviews Neuroscience, 10(7), 519-529.

Datta, S. R., Anderson, D. J., Branson, K., Perona, P., & Leifer, A. (2019). Computational neuroethology: A call to action. Neuron, 104(1), 11-24.

Gomez-Marin, A., Paton, J. J., Kampff, A. R., Brody, C. D., & Mainen, Z. F. (2014). Big behavioral data: Psychology, ethology and the foundations of neuroscience. Nature Neuroscience, 17(11), 1455-1462.

Graving, J. M., Chae, D., Naik, H., Li, L., Koger, B., Costelloe, B. R., & Couzin, I. D. (2019). DeepPoseKit, a software toolkit for fast and robust animal pose estimation using deep learning. eLife, 8, e47994.

Hall, C. S. (1934). Emotional behavior in the rat. I. Defecation and urination as measures of individual differences in emotionality. Journal of Comparative Psychology, 18(3), 385-403.

Harris, C. R., Millman, K. J., van der Walt, S. J., et al. (2020). Array programming with NumPy. Nature, 585(7825), 357-362.

Hsu, A. I., & Yttri, E. A. (2021). B-SOiD, an open-source unsupervised algorithm for identification and fast prediction of behaviors. Nature Communications, 12(1), 5188.

Jhuang, H., Garrote, E., Mutch, J., Yu, X., Khilnani, V., Poggio, T., & Serre, T. (2010). Automated home-cage behavioural phenotyping of mice. Nature Communications, 1(1), 1-10.

Jocher, G., Chaurasia, A., & Qiu, J. (2023). YOLO by Ultralytics (Version 8.0.0) [Computer software].

Kalueff, A. V., Stewart, A. M., Song, C., Berridge, K. C., Graybiel, A. M., & Fentress, J. C. (2016). Neurobiology of rodent self-grooming and its value for translational neuroscience. Nature Reviews Neuroscience, 17(1), 45-59.

Kraeuter, A. K., Guest, P. C., & Sarnyai, Z. (2019). The Open Field Test for measuring locomotor activity and anxiety-like behavior. Methods in Molecular Biology, 1916, 99-103.

Kyzar, E. J., Stewart, A. M., Meyer, N. N., & Kalueff, A. V. (2015). Recent advances in analyzing mouse open field tracking datasets. Methods in Molecular Biology, 1201, 107-119.

Lauer, J., Zhou, M., Ye, S., Menegas, W., Schneider, S., Nath, T., Rahman, M. M., Di Santo, V., Soberanes, D., Feng, G., Murthy, V. N., Lauder, G., Dulac, C., Mathis, M. W., & Mathis, A. (2022). Multi-animal pose estimation, identification and tracking with DeepLabCut. Nature Methods, 19(4), 496-504.

Mathis, A., Mamidanna, P., Cury, K. M., Abe, T., Murthy, V. N., Mathis, M. W., & Bethge, M. (2018). DeepLabCut: Markerless pose estimation of user-defined body parts with deep learning. Nature Neuroscience, 21(9), 1281-1289.

Mathis, M. W., & Mathis, A. (2020). Deep learning for deep behavioral phenotyping. Current Opinion in Neurobiology, 60, 1-11.

McKinney, W. (2010). Data structures for statistical computing in Python. Proceedings of the 9th Python in Science Conference, 445, 51-56.

Nath, T., Mathis, A., Chen, A. C., Patel, A., Bethge, M., & Mathis, M. W. (2019). Using DeepLabCut for 3D markerless pose estimation across species and behaviors. Nature Protocols, 14(7), 2152-2176.

Noldus, L. P., Spink, A. J., & Tegelenbosch, R. A. (2001). EthoVision: A versatile video tracking system for automation of behavioral experiments. Behavior Research Methods, Instruments, & Computers, 33(3), 398-414.

Pereira, T. D., Aldarondo, D. E., Willmore, L., et al. (2019). Fast animal pose estimation using deep neural networks. Nature Methods, 16(1), 117-125.

Pereira, T. D., Tabris, N., Matsliah, A., et al. (2022). SLEAP: A deep learning system for multi-animal pose tracking. Nature Methods, 19(4), 486-495.

Prut, L., & Belzung, C. (2003). The open field as a paradigm to measure the effects of drugs on anxiety-like behaviors: A review. European Journal of Pharmacology, 463(1-3), 3-33.

Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 779-788.

Seibenhener, M. L., & Wooten, M. C. (2015). Use of the Open Field Maze to measure locomotor and anxiety-like behavior in mice. Journal of Visualized Experiments, (96), e52434.

Simon, P., Dupuis, R., & Costentin, J. (1994). Thigmotaxis as an index of anxiety in mice. Behavioural Brain Research, 61(1), 59-64.

Spink, A. J., Tegelenbosch, R. A., Buma, M. O., & Noldus, L. P. (2001). The EthoVision video tracking system: A tool for behavioral zoology. Forensic Science International, 118(2-3), 225-241.

Stauffer, C., & Grimson, W. E. L. (1999). Adaptive background mixture models for real-time tracking. Proceedings, 1999 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), 2, 246-252.

Streamlit Team. (2024). Streamlit: The fastest way to build and share data apps [Computer software].

Sturman, O., Zarco, I. G., Landon, M., et al. (2020). Deep learning-based behavioral analysis reaches human accuracy and is capable of outperforming commercial solutions. Neuropsychopharmacology, 45(11), 1942-1952.

Treit, D., & Fundytus, M. (1988). Thigmotaxis as a test for anxiolytic activity in rats. Pharmacology Biochemistry and Behavior, 31(4), 959-962.

Tse, D. Y., Chung, S. K., & Roberts, J. (2018). High-throughput automated behavioral assessment. Neurobiology of Learning and Memory, 150, 148-154.

Virtanen, P., Gommers, R., Oliphant, T. E., et al. (2020). SciPy 1.0: Fundamental algorithms for scientific computing in Python. Nature Methods, 17(3), 261-272.

Walsh, R. N., & Cummins, R. A. (1976). The Open-Field Test: A critical review. Psychological Bulletin, 83(3), 482-504.

Zivkovic, Z. (2004). Improved adaptive Gaussian mixture model for background subtraction. Proceedings of the 17th International Conference on Pattern Recognition (ICPR), 2, 28-31.