One Fish, Two Fish, Red Fish, Blue Fish:

Deep Learning Tracks Network Dynamics in a Highly Social Cichlid Fish

A Thesis Presented

by

Erika A. Moore

Submitted to the Department of Natural Sciences

Pitzer College and Scripps College

In Partial Fulfillment of the Requirements for the Degree of

Bachelor of Arts in Neuroscience

December 9, 2024

Abstract

Tracking the identities of organisms accurately is critical for reliable behavioral analysis. Manual approaches are both time consuming and error prone, especially when studying *Astatotilapia burtoni*, a highly social cichlid fish. This species displays complex patterns of interaction in groups, and individuals are difficult to distinguish. In this project, we leverage the deep learning-based framework Social Leap Estimates Animal Poses (SLEAP) to maintain identities across frames. Despite overlaps, occlusions, and relatively little annotated training data (1610 frames), we successfully track individual fish in 32 full-length videos. Each video is 20-minutes in length, containing a single group of either 1, 2, 3, or 4 fish. We apply Centroid and Centered-Instance models to the videos to generate 1,152,000 total frames with annotated identities. From coordinate data, we analyze the impact of group size on social behavior, exploring a variety of metrics related to spatial preferences. We find a nuanced effect of group size on social interaction, with larger groups displaying more intricate and varied behaviors. For example, fish in triads and tetrads traveled significantly farther than individuals or pairs. Revealing these hidden layers of social complexity through computational approaches deepens our understanding of biological systems, such as the elements of social experience that shape development in *A. burtoni* and other organisms alike.

Contents

1 Introduction

Computer vision has advanced rapidly in recent years. For instance, Segment Anything (SA) can 'cut out' any object, in any image, with a single click [10], DALLE can translate written language into contextually accurate images [18], and Vision Transformers (ViT) are showing remarkable promise in medical diagnostic testing [9].

However, there is a disconnect: automation has not yet become widespread in natural science research. Though most laboratories study model organisms, few are able to track them efficiently. Researchers often rely solely on manual methods of tracking. This places unnecessary constraints on science, limiting both the questions that can be asked and the types of studies that can be carried out.

Thus, the aim of this study is twofold:

1. We aim to leverage deep learning to track *A. burtoni*.
2. We aim to reveal hidden layers of social complexity through subsequent spatial analyses.

1.1 Tracking Identity

Tracking the identities of organisms accurately is critical for reliable behavioral analysis. For example, Harmon et al. (2024) [8] rely on a tattoo-ink-based tracking method to track *Astatotilapia burtoni*. Unfortunately, not all ink tags were visible on video, prohibiting them from quantifying individual social experience, social network position, and spatial position in some fish. Nonetheless, high throughput tracking technologies can mitigate such limitations, enabling a more comprehensive investigation into the intricacies underlying individual variation, social environment, and behavior.[13][22] Manual approaches are both time consuming and error prone. This is especially true when studying *A. burtoni*, a highly social cichlid fish known for its behavioral and phenotypic adaptability.[8][21] This species displays complex patterns of interaction, making it difficult to distinguish between individuals when in groups.

1.2 Tracking Frameworks

Machine learning approaches to automated tracking offer a promising solution, enabling us to: improve scale and reliability in neuroscience research, gain deeper insight into behavior across species, and reveal hidden layers of social complexity. Automated tracking frameworks, such as DeepLabCut [11] and Social Leap Estimates Animal Poses (SLEAP) [15], can be used to track multiple animals simultaneously. Intended for laboratory settings, these tools enable researchers to identify individuals more consistently. They also provide precise pose-estimation and frame-by-frame coordinate locations. This allows data to be analyzed in new ways. For instance, automated tracking has also been used to study opioid use disorders (OUDs) in rodents [4], classify eusocial behaviors in naked mole rats [20], and infer foraging strategies in elephants [16].

Here, we leverage the deep learning framework SLEAP to investigate the impact of group size on spatial preferences in *A. burtoni*. With many social partners and complex patterns of interaction, it is especially difficult to keep track of these fish. By manipulating group size, we introduce variation and demonstrate the applications of automated tracking in this species.

1.3 An Overview of SLEAP

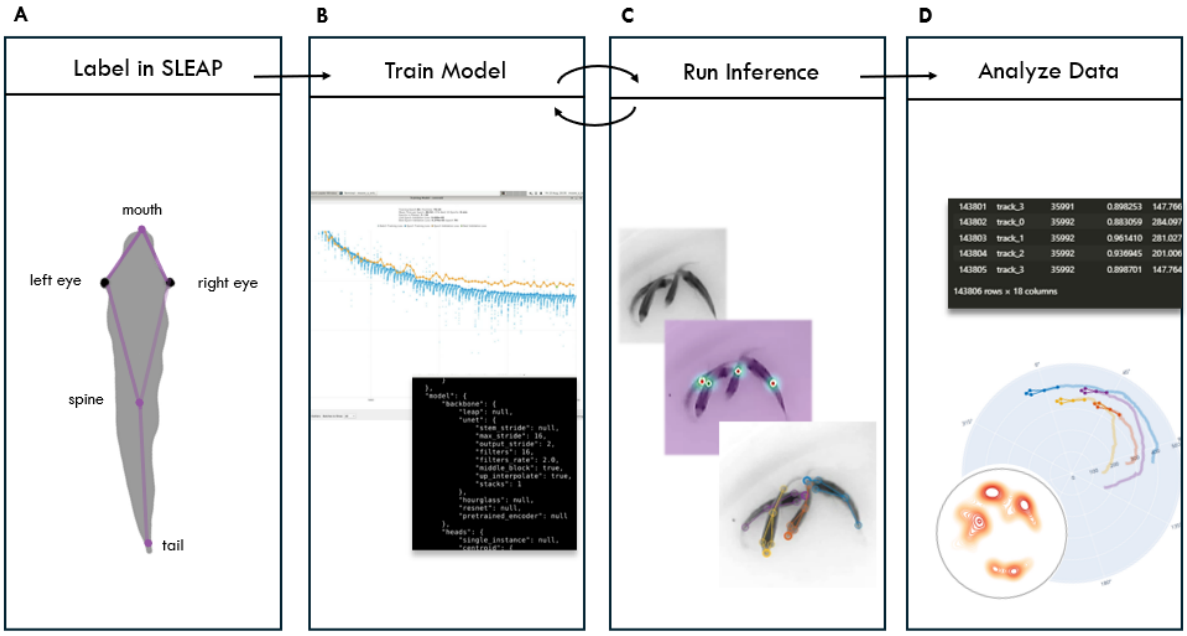


Figure 1. SLEAP tracking framework:
(A) labeling, (B) training, (C) inference, (D) custom analysis of generated coordinate data.

SLEAP is a software package for tracking and pose-estimation. As input, SLEAP processes a video, and as output, SLEAP returns a video with tracked instances and the frame-by-frame coordinate locations of body parts. As shown in Figure 1, working with SLEAP involves labeling (A), training (B), inference (C), and the analysis of spatial-temporal coordinate data (D). Labels for an animal on a given frame are referred to as a 'skeleton'. The skeleton can be thought of as a network of body parts, where nodes represent specific locations on the animal (Figure 1A). The skeleton-based labeling allows for the recovery of individual identities when occlusions and overlaps occur. Examples of occluded fish are shown in Figure 2. SLEAP utilizes human-in-the-loop training. This means users can review and manually correct predictions at various stages during training through a label refinement process.

SLEAP is a supervised training process, learning only from ground-truth labels. Its strength is being able to use a small number of manually annotated frames to identify instances of animals in subsequent frames. SLEAP leverages deep learning models to as-

Figure 2. Two examples of occlusions. Four fish in total are shown.

sociate skeletal structures to these instances. Manual corrections are made in the SLEAP training process to refine predicted labels. These new labels are then used for further predictions. Various types of models can be used for multi-animal pose tracking. Two popular choices are Centroid and Centered-Instance models. The Centroid model relies upon a top-down processing framework, leveraging the centroid of an animal to then identify corresponding body parts. The Centered-Instance model, however, uses bottom-up processing, localizing body parts prior to identifying the animal that they belong through a probability distribution. These two approaches when used help SLEAP overcome occlusions where parts of an animal are obscured. Traditional motion algorithms are used in cross-frame identity tracking.

As with many machine learning tasks, the inference process is used to predict across previously unseen frames of video. This generates coordinate locations of tracked body parts through pose-estimation. These coordinates can be exported in comma-separated values (CSV) form for subsequent processing. SLEAP demonstrates remarkable potential for animal tracking, yet it is computationally expensive and is best used with a Graphics Processing Unit (GPU). We aim to leverage this framework for automating the tracking of our cichlid fish.

2 Methodology

2.1 Experimental Design

The animals used in this study were young adult *A. burtoni* fish. These fish were bred in the lab and selected from the same tank. To create a size gradient, the fish were individually placed in cups and sorted by relative body size. From these sorted fish, 80 individuals were selected to form the experimental sample. The 80 fish sample was then divided into 32 groups (Table 1): 8 groups of individuals, 8 groups of pairs, 8 groups of triads, and 8 groups tetrads. Within each group of multiple fish, individuals varied in body size (e.g., small, medium, large).[1]

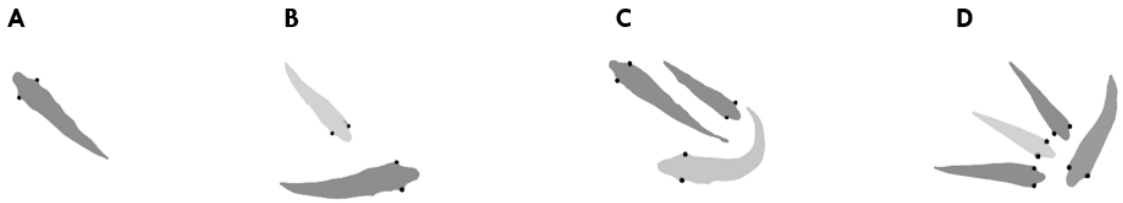


Figure 3. (A) Individuals, (B) pairs, (C) triads, (D) tetrads.

Fish Group Size	**Unique Groups (n=8)**	**Videos (20 min each)**	**Frames**
Individual (1 fish in video)	8	8	288,000
Pair (2 fish in video)	8	8	288,000
Triad (3 fish in video)	8	8	288,000
Tetrad (4 fish in video)	8	8	288,000
Total Frames	**32**	**32**	**1,152,000**

Note:
- The dataset consisted of 80 fish total.
- No individual fish were reused across groups or videos.

Table 1. Summary of dataset: group sizes, replicates, videos, and frames per video.

[1]The modest variation in fish size assisted in the human validation of tracking accuracy. While SLEAP can distinguish animals based on appearance, this was not used in our experiment.

This grouping method ensured variation both within and across groups, allowing us to examine the effects of body size without requiring each fish to be weighed. To allow for group formation, each group was placed in a separate holding tank. During the holding period, tanks were divided such that individuals from different groups could not see one another. This helped to minimize external influences on behavior. Filming occurred either 24 or 48 hours following group formation.

2.2 Video Acquisition Details

Videos were filmed on Raspberry Pi 5 3MP HQ cameras with a 6mm wide-angle lens. At most, 3 groups were filmed at a time because we only had 3 cameras. To film a given group, the fish were transferred to a white 5-gallon food-grade plastic bucket. These specific buckets were chosen to enhance contrast in the videos and to reduce environmental background clutter, such as algae or gravel, which are not optimal for segmentation tasks. Between each filming session, the water in the buckets was changed to ensure that hormonal or neurotransmitter residues from prior groups did not affect subsequent groups. The water level was filled to a consistent depth each time. The 3 Raspberry Pi cameras were mounted on a metal grid above each of the 3 buckets. This ensured that all fish within a given group remained in the frame throughout the video and allowed for a consistent camera placement. Filming began after a 10-minute acclimation period intended to allow the fish to settle after the transfer process. Filming was started and stopped remotely with a Python script that controlled the Raspberry Pi cameras. This minimized human interference during the filming process. Each group was filmed at 30 frames per second at 1080p resolution, and each video was 20 minutes in length (i.e., $36,000$ frames). Following filming, the 32 videos were converted to greyscale. Contrast and exposure were adjusted using Microsoft Clipchamp.

2.3 Computational Environment

Following labeling and training, SLEAP models were used to track individual fish identities. Initially, attempts were made to train a model locally on a Windows machine (HP

Spectre x360, Intel Core i7-1165G7, 16GB RAM). However, SLEAP requires more compute power to be usable – a CUDA-compatible GPU and significant RAM for efficient model training. To achieve this, we used cloud computing. This was a bit like 'renting' space on a server in Utah so that we did not need to buy physical hardware ourselves.

Google Cloud was selected as the cloud hosting platform. A Virtual Machine (VM) was set up and acted as a cloud-hosted server. The VM provided access to necessary compute resources, including multiple CPUs, a CUDA-compatible GPU, memory, and storage. Linux, `Chrome Remote Desktop`, and SLEAP were installed on the VM. Training on the cloud-based VM was substantially faster than training on the local machine

Cloud providers charge for every minute of compute time. To minimize unnecessary expenditures, we used a type of VM called a `spot instance`. A spot instance is approximately 60% cheaper than a standard, on-demand instance but provides access to the same compute resources. The slight caveat is that a spot instance may be interrupted by the cloud provider at any time if said compute resources are requested by a higher priority (i.e., higher-paying) customer. The spot instance pricing model was suitable for our purposes, as occasional interruptions during model training and inference did not substantially impact the workflow.

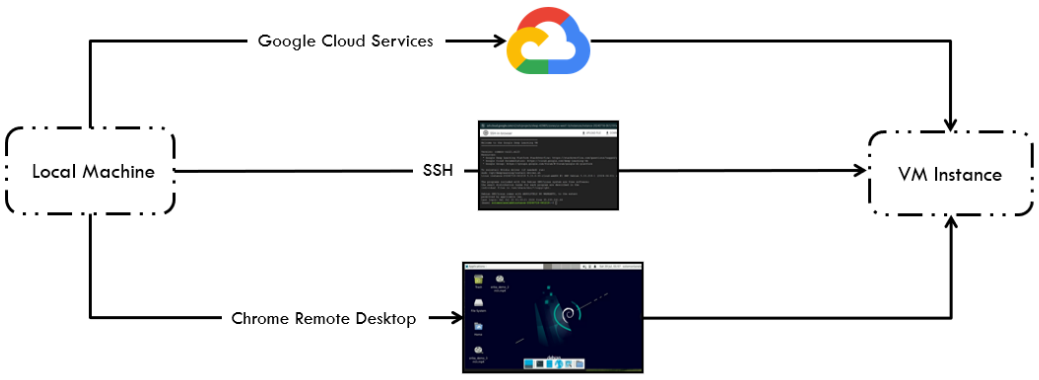


Figure 4. Local control of remote Virtual Machine (VM) using Chrome Desktop and Secure Shell.

A graphical interface for the VM was provided by installing the lightweight `xfce4` desktop environment (Figure 4). `Chrome Remote Desktop` was installed for remote access, and user privileges were updated accordingly. A separate `Conda` environment was created to reduce dependency conflicts in the SLEAP installation process.

The VM was configured with the following specifications:

Component	**Details**
Operating System	Debian 11 image with CUDA preinstalled. Debian 11 is a stable version of Linux, and having CUDA preinstalled ensured GPU compatibility.
vCPUs	4 high-memory virtual CPUs (vCPUs).
Disk Size	100GB boot disk.
GPU	NVIDIA T4 GPU
Kernel	Deep-learning optimized kernel for better performance.

Table 2. VM configuration specifications.

2.4 Applying SLEAP to Our Dataset

2.4.1 Training Process

After recording the videos, the poses of all fish were tracked across frames using SLEAP. The Centroid and Centered-Instance models were trained incrementally using a total of 1610 manually labeled frames. Training began with model weight initialization using 53 annotated frames from a representative video. This was followed by baseline training on 1210 frames and further refined with additional annotations (Table 3). These trained models were subsequently applied to unseen videos in the dataset for pose tracking.

Data Source	**Purpose**	**Annotated Frames**
A representative video (not in dataset)	Model weight initialization	53 (not counted)
3 groups of 4 fish from the dataset	Baseline training with transfer learning	1,210
Additional annotations on 1 group of 4 fish	Model refinement	333
Additional annotations on 1 larger fish	Model refinement	67
	Total	**1,610**

Table 3. Sources of annotated frames used in training from the 32-video dataset.

Inference was run on our 32-video dataset to detect the body parts of fish through pose-estimation. Thus, inference was run across $\sim 1,152,000$ frames in total.[2] Following inference, predictions were manually reviewed to verify accuracy in identity tracking and pose-estimation.

SLEAP supports various types of models for multi-animal pose tracking. We utilized a Centroid model and a Centered Instance Model, as shown in Table 4. These two models were chosen for their complementary strengths: the Centroid Model efficiently identifies animal locations, and the Centered Instance Model precisely estimates their poses. Together, they create a robust and scalable solution for multi-animal pose tracking in SLEAP. This approach ensures accuracy and efficiency, even in challenging scenarios with overlapping or densely packed animals.

Our model specifications can be summarized as follows:

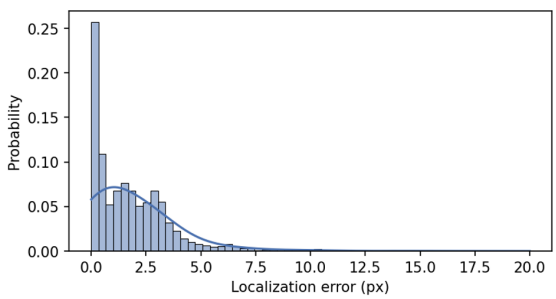
Feature	**Centroid Model**	**Centered Instance Model**
Architecture	UNet (16 filters, output stride 2)	UNet (24 filters, output stride 4)
Purpose	Predict centroids	Predict key body parts
Training Data	1610 labeled frames	1610 labeled frames
Input Scaling	0.4	1.0
Output Stride	2	4
Instance Cropping	None	Cropping around keypoints, crop size: 224
Augmentation	Rotation (-180° to 180°)	Rotation (-180° to 180°)
Batch Size	4	4
Inference Data	1,152,000 frames (32 videos)	1,152,000 frames (32 videos)

Table 4. Centroid and Centered Instance model specifications

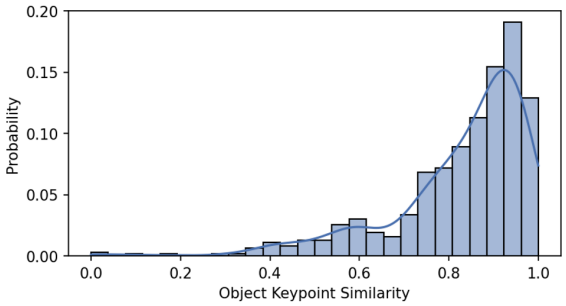
[2]Out of the 1,152,000 frames that we ran inference on, only 1,610 of these frames (0.14%) were seen in the iterative training process. A small fraction of the total number of frames in the dataset were seen during training.

2.4.2 Validating Model Performance

In addition to visually verifying tracking accuracy, we reviewed metrics for localization error and Object Keypoint Similarity (OKS) (Figure 5). Localization error measures how far off the predicted body parts were from the ground truth. OKS provides a more holistic assessment, taking into account factors such as landmark visibility and object size. While these metrics offered valuable insight, we found that human visual inspection tended to provide a more representative estimate of overall model performance.



Percentiles of the localization error distribution: *How far off the model's predictions were from the ground truth locations. Lower values indicate better accuracy.*



Object Keypoint Similarity (OKS): ***A*** *more holistic metric that considers visibility, object size, and keypoint difficulty. Higher OKS scores indicate better overall pose estimation accuracy.*

Figure 5. Centered Instance Model Evaluation Metrics

3 Quantitative Behavioral Analysis

3.1 Measures of Central Tendency

3.1.1 Calculating the Center of the Tank

Calculating the geometric center of each tank provides a stable reference point for consistent spatial comparisons across videos.[3] To identify this center reliably, we explored two methods: the *midpoint method* and the *least squares method*.

Midpoint Method

The Midpoint Method (Figure 6) estimates the center of a circular region based upon the farthest-apart fish in each frame. Consider a circular region with multiple points in it. The midpoint between two points that are far apart, or even the farthest apart, can provide a reasonable estimate of the geometric center of the circular region. In our case, these farthest-apart "points" happen to be fish at opposite sides of the circular region. Thus, the midpoint of their coordinates should be near the center.

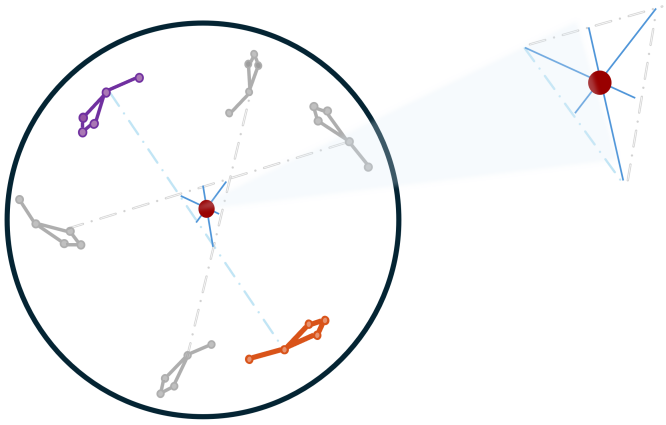


Figure 6. The midpoint method uses the centroid of a triangle to find the center.

[3]SLEAP provides the locations of individual fish and not other objects in the frame. We leverage the coordinates of individuals to allow reliable comparisons without further image processing.

In each frame, we:

1. Found the pair of fish that were furthest apart,
2. Calculated the midpoint of the line segment connecting the pair, and
3. Averaged these midpoints across all frames

The use of the farthest distances approximates a diameter. The average midpoint of these distances is a fairly robust estimate of the center of the bucket. What is nice about this technique is it relies only on the location of the fish and no special image processing is needed to establish a frame of reference.

Least Squares Method

The second method involved an optimization technique known as *least squares* (Figure 7). We begin by taking an initial guess of the center. This initial guess is simply the average of the x and y coordinates for all of the fish. Then, we use the optimization process, `least_squares`, to adjust this estimated center.

```
result = least_squares(circle_fit_error, initial_guess, args=(points,))
```

The goal is to minimize the sum of squared differences between each fish's distance to the estimated center and the average distance. By iteratively reducing these differences, the center becomes balanced with respect to all fish, similar to how all radii are equidistant from the center of a perfect circle.

Figure 7. Finding the geometric center through least squares optimization.

One advantage of this approach is that it takes into account the positions of all individuals, rather than just the extremes. However, if all of the fish cluster in one area consistently, the center estimate may be biased towards that cluster. It is also more computationally expensive than the midpoint method.

Nonetheless, both approaches provided reliable estimates for the center of the bucket. We verified the accuracy of our estimated centers by extracting the frame index where a particular fish ID was closest to the computed center. We then visually inspected the position of the fish in that frame by referencing the video.

Thus, the midpoint method was a computationally efficient way to find the geometric center, while the least squares method provided a more refined estimate. By considering these methods together, we can obtain reliable estimates for the center of each bucket, a crucial step for downstream spatial analyses.[4]

[4]And, considering the circular nature of buckets, it is pretty satisfying that we could find a 'least square' way to do so.

3.1.2 Average Distance to Center

Centering Coordinates

Once we had calculated the center of the tank, we could explore measures of 'central tendency.' Specifically, we investigated the average distance of each fish to the center of the bucket. Since SLEAP provides Cartesian (x, y) coordinates with $(0, 0)$ at the top left corner of the video frame, we used our computed center coordinates to recenter the coordinates such that $(0, 0)$ represented the center of the tank.

Converting Coordinates to Polar Form

Once our origin was centered appropriately, we converted the Cartesian coordinates into polar coordinates. Polar coordinates, described in terms of (r, θ), are more suitable for describing circular regions and radial distances. Treating the spine of each fish as its centroid, we used the radial component to represent the distance of each fish to the center. From this, we computed the mean radius for each tracked fish. This provided us with the average distance of each fish to the center over the course of the video (Figure 8).

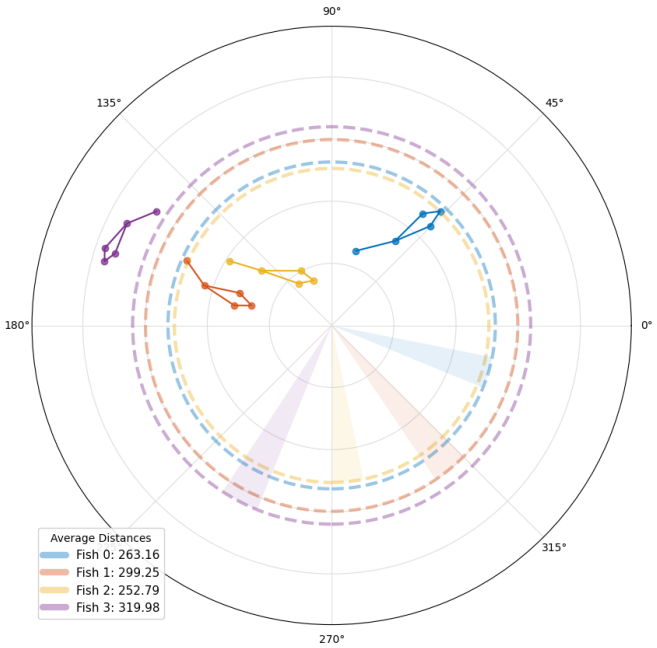


Figure 8. Reconstructed image of individual fish in a frame from coordinate data. Lines are average distances to center per fish across all frames.

Biological Relevance

The average distance to center is biologically relevant, as this metric reflects spatial preferences. In other species, researchers have observed that central tendencies may reflect dominant behaviors.[5][14] In contrast, preferences for edges may suggest anxiety-like behaviors. For instance, the open field test, first described by Hall [7], involves placing an animal, typically a rodent, in an unfamiliar walled arena and observing its behavior. Increased time spent in the center and reduced latency to enter the center suggest lower anxiety levels, or anxiolysis [17]. In contrast, more anxious rodents tend to stay closer to the walls. This behavior is referred to as thigmotaxis. Such studies are often used to examine the effects of anxiolytic drugs.

For *A. burtoni*, the average distance of each fish to the center may reflect individual variations in stress responses, and given this species' dynamic social structure, such insights may be valuable.

3.2 Velocity Analysis with CWT

Motivation for Spatial-Temporal Analysis

Time series data is inherently quite messy, especially for quickly moving animals with sudden 'bursts' of energy. This makes computing metrics like velocity slightly less straightforward than one may initially suppose. We used Savitzky-Golay filtering to calculate the smoothed velocities of the fish. We found that a degree 3 polynomial and a sliding window size of 25 frames seemed to provide a reasonable ratio of smoothness to noise. This was consistent with a Jupyter notebook provided by the developers of SLEAP, and initially, we thought this would be satisfactory for calculating the average velocity of the fish. Then, we realized that such a calculation would have a critical logical flaw and was unlikely to be informative. In fact, it would be a bit like Simpson's Paradox [3], a phenomenon in which taking 'crude' averages misrepresents underlying patterns or groupings that may exist within the data. In this case, it would seem unreasonable to take the average velocity for each fish, as many of the groups tend to alternate between idle and active periods. This means that, if we were to take the average velocity values for, say, a given fish, this would not actually tell us much about that fish's behavioral dynamics.

As a concrete example, suppose we have one fish that is stationary for half a video and then speeds up to twice the speed of another fish. If the other fish were to swim at a constant speed throughout the duration of the video, the two fish would have the same average yet completely different patterns of behavior.

Wavelet Analysis

Thus, in order to represent average velocities more appropriately, we needed a different method, one localized in both time, frequency, and appropriate for small, aquatic species — wavelets. The term wavelet means 'little wave' and is often thought of as the lesser-known counterpart of the Fourier transform.[16] Nonetheless, the two types of functions are actually quite different. Wavelets are localized in both the time and frequency domain,

whereas Fourier transforms are localized only in the frequency domain.[5] Wavelets are more like an orchestra score or melody, showing the frequency with respect to time. For the fish, we actually want to be able to distinguish between idle periods and active periods, and wavelets enable us to 'extract' these higher activity periods specifically by scaling and translating a 'mother wavelet' along our signal. When we have a better fit of our wavelet to the signal at some given interval of time, the energy of the wavelet coefficients is higher at that time.

We used continuous wavelet transform to extract the velocities at various intervals in time. Then, we applied KMeans to cluster the energy levels of the wavelet coefficients into discrete categories (Figure 9). This way, if we so desired, we could calculate the average velocities of the fish without these values being weighted down by idle periods.

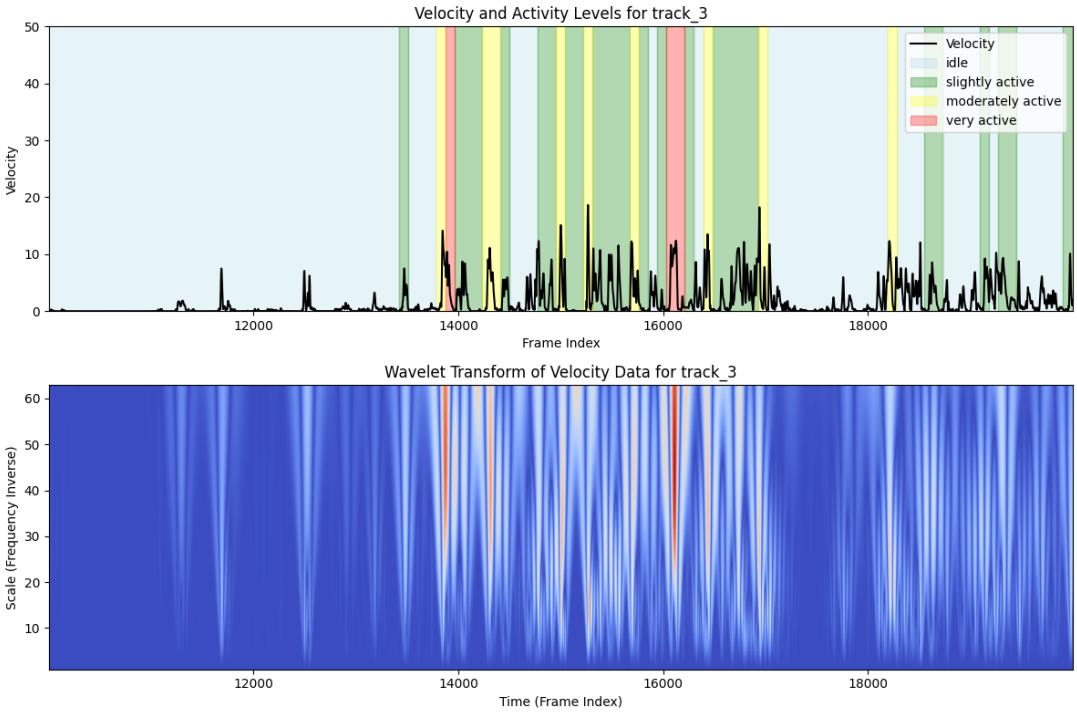


Figure 9. Example wavelet analysis for a single fish. Top: Identification of different periods of activity level. Bottom: Visualization of changes in velocity over time.

[5]As a bit of a side note, the most classic example of a Fourier transform is with music. While we may be able to decompose a given sound, or signal, into its component frequencies (component 'functions'), this actually does not tell us any information about which sounds are being played at a given time. This is due to the uncertainty principle, as there must always be a tradeoff between time and frequency domains. Wavelets represent a combination of frequency and time (i.e., windows of activities).

3.3 Kernel Density Estimation (KDE)

Estimating Spatial Preferences

We used the Seaborn Kernel Density Estimation (KDE) plot to provide a visual representation of spatial preferences. The KDE provides a smooth, continuous probability density estimate of data in one or more dimensions. In our case, the data consisted of the two-dimensional coordinate locations (x, y) of each fish. KDE holds similarities to a probability density function (PDF), except a KDE is based upon sample data rather than a predefined distribution. KDE offers a non-parametric way to estimate the regions where a fish may be more or less likely to reside, and given that the tanks did not have predefined zones, this provided for a fairly interesting way to see some broader trends in the data (Figure 10). We adjusted the bandwidth to strike the right balance of smoothness to detail, and we applied thresholding to exclude low-density areas. This enabled us to focus on regions of interest with greater clarity. This took a bit of trial and error and seemed to be in part dependent upon the total video duration.

Assumptions of KDE

Interestingly, KDE does have assumptions associated with it. For instance, KDE assumes each position is independent, and this is not quite true in our case, as each (x, y) coordinate is from a time series and is, thus, dependent. KDE also assumes that data can extend beyond physical boundaries. This is because the smoothing algorithm uses a Gaussian kernel, so the estimated density may extend to values that are not necessarily part of a particular dataset. We attempted to address this visually by overlaying the limits of the tank itself, plotting a circular boundary given that we had predetermined using the radius of the tank.

We then used contour levels to show density gradients. This highlighted areas of higher densities (i.e., more common regions), much like how the height of a PDF would represent the density associated with a particular interval. We then compared spatial preferences amongst individuals. KDE provides biological insight into the preferred areas of the tank

based upon higher-density regions. Similar to our earlier analysis of average distance to center, this metric provides an alternative way in which we can compare spatial densities. Rather than rely on a precise number, an average or expected value, we instead rely upon densities. With no physical boundaries other than the tank itself, KDE helps to reveal patterns within the underlying distribution of the data. Just as probability densities show likely outcomes, high-density KDE regions reveal preferred areas, indicating possible spatial or social preferences.

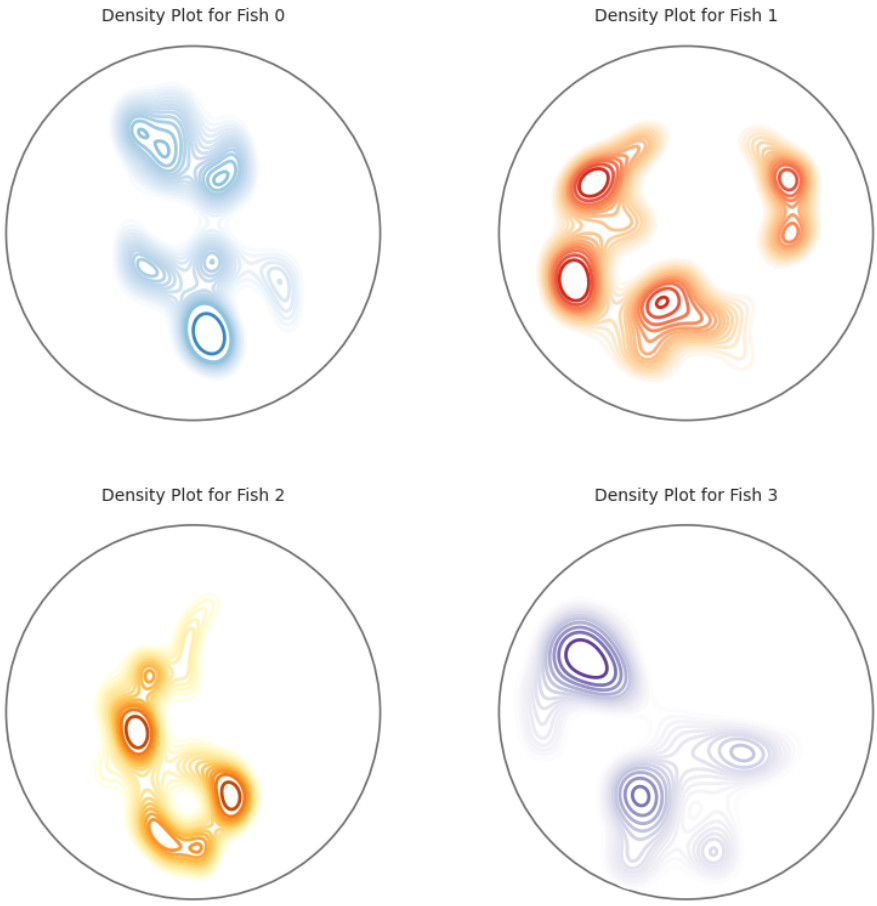


Figure 10. Sample KDE plots for a group of four fish.

3.4 Size Estimation

To estimate the relative size of each fish, we used the keypoint coordinates exported from SLEAP. Since SLEAP tracks each fish with a skeleton connecting specific keypoints, we reasoned that calculating the length of each skeleton would be a reasonable proxy for fish size. Each keypoint represented a specific body part and provided an (x, y) coordinate for each frame of the video.[6]

Skeleton Length Calculation

We began by preprocessing the coordinate data, linearly interpolating any missing values. Using the keypoints, we reconstructed each SLEAP skeleton by calculating the Euclidean distances between connected nodes. The keypoints were treated as nodes, and the connections between them were considered to be edges. These distances were used to approximate the lengths of the skeleton.

After determining the edge lengths, we calculated the total skeletal length for each fish within each video frame by summing all of the edge lengths (Figure 11). Since estimated length varied slightly across frames due to changes in position and depth, we iterated across all frames to ensure reliable estimates.

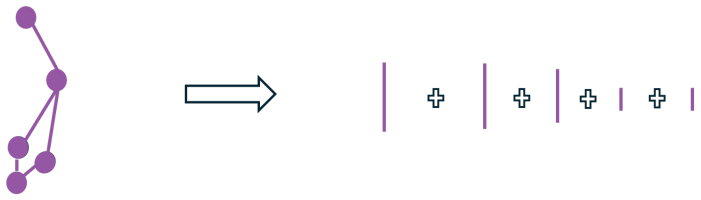


Figure 11. Estimating fish size by summing edge lengths.

[6]While tools like ImageJ [19] are, perhaps, more commonly used in size estimation tasks, our approach enabled us to leverage tracking data from 36,000 frames per fish and integrate the analysis into our existing workflow.

Averaging and Biological Relevance

We then averaged the measurements across all frames for each fish to obtain a stable size estimate. We also considered the standard deviation for the estimated size of each fish to gauge variability across frames.

This method allowed us to compare relative sizes of different fish and calculate summary statistics. Fish size is biologically relevant, as it often correlates with social status, particularly in *A. burtoni*. Fish size also influences behaviors and hierarchies. Thus, by having a proxy for relative fish size, we can gain insights into social dynamics, dominance, and individual roles within a group.

3.5 Total Distance Traveled

To calculate the total distance each fish traveled, we began by preprocessing the data as per usual, linearly interpolating any missing data values to ensure consistency in time steps. We filtered the dataframe for each track, and calculated the differences Δx, Δy between consecutive frames with NumPy, using the spine as our centroid for each fish. We then applied the distance formula iteratively to evaluate the total distance traveled between consecutive frames and summed across all frames for each track. Given that this metric was arguably more computationally efficient to compute than other measures, than say, KDE, we created a main processing loop to carry out the operation across the 80 fish in our 32-video dataset, looping through all paths. This provided us with the total distance traveled, in pixels, by each fish in our dataset within a few minutes.

```
distances = np.sqrt(np.sum(np.diff(coordinates, axis=0) ** 2, axis=1))
total_distance = distances.sum()
```

Following this, we sought out to determine if particular group sizes had a tendency to travel farther than others. Moreover, we were curious about how the total distances traveled by members of the same group may compare. The results section focuses upon this data specifically.

4 Results

ANOVA Results

Total distance traveled varied with group size, with groups of size 4 displaying the greatest mean total distance traveled (Figure 12 and Appendix). An ANOVA indicated a significant effect of group size on total distance traveled ($F(3, 76) = 26.67$, $p < 0.001$, $\eta^2 = 0.51$). Post-hoc Tukey tests (Table 5) showed that individuals in larger groups (sizes 3 and 4) traveled significantly farther than those in smaller groups (sizes 1 and 2). Specifically, groups of 4 fish traveled significantly farther than all other group sizes, while groups of 3 fish traveled significantly farther than groups of sizes 1 and 2. However, there was no significant difference between the distance traveled by groups of sizes 1 and 2.

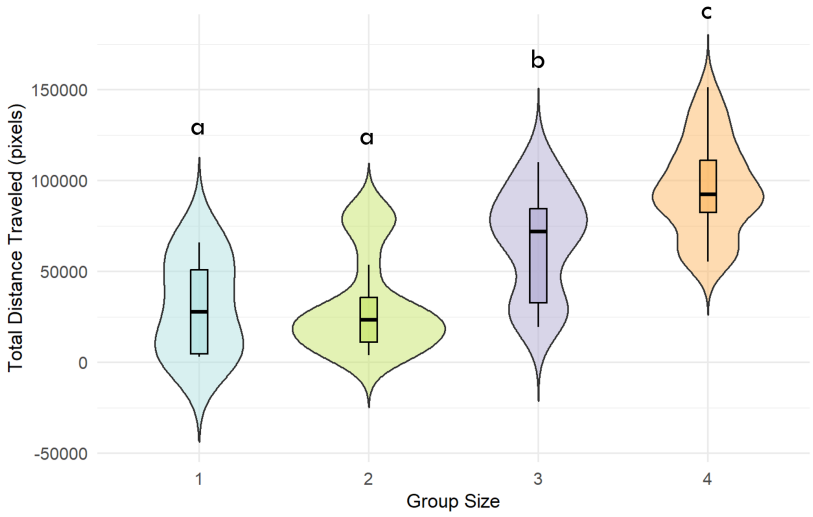


Figure 12. Distribution of total distance traveled by group size. Different letters denote signficant differences between group sizes.

Comparison	***p*-value**
3 vs. 4	0.00067
2 vs. 4	< 0.00001
1 vs. 4	< 0.00001
2 vs. 3	0.00068
1 vs. 3	0.00859
1 vs. 2	0.99992

Table 5. Tukey post-hoc test results for group size comparisons.

Intraclass Correlation Coefficient Results

To further explore the influence of group composition on total distance traveled, we computed the Intraclass Correlation Coefficient (ICC) for the groups of size 2, 3, and 4. For a given group size, the ICC reflects the proportion of variation in distance traveled that is explained by variation between groups, rather than within a specific group.

For groups of size 2, 3, and 4, the ICC values were 0.927, 0.846, and 0.748, respectively. These high ICC values indicate that group identity (i.e., the unique composition of a group) has a substantial impact on the total distance traveled. In other words, when holding group size constant, individuals within a particular group tend to travel more similar distances than individuals across different groups. The results for groups of size 4 are shown in Figure 13.

We note that as group size increases, the ICC decreases. This indicates that individuals within larger groups are more varied in the total distances they travel, perhaps reflecting more complex social dynamics and/or reduced group cohesion.

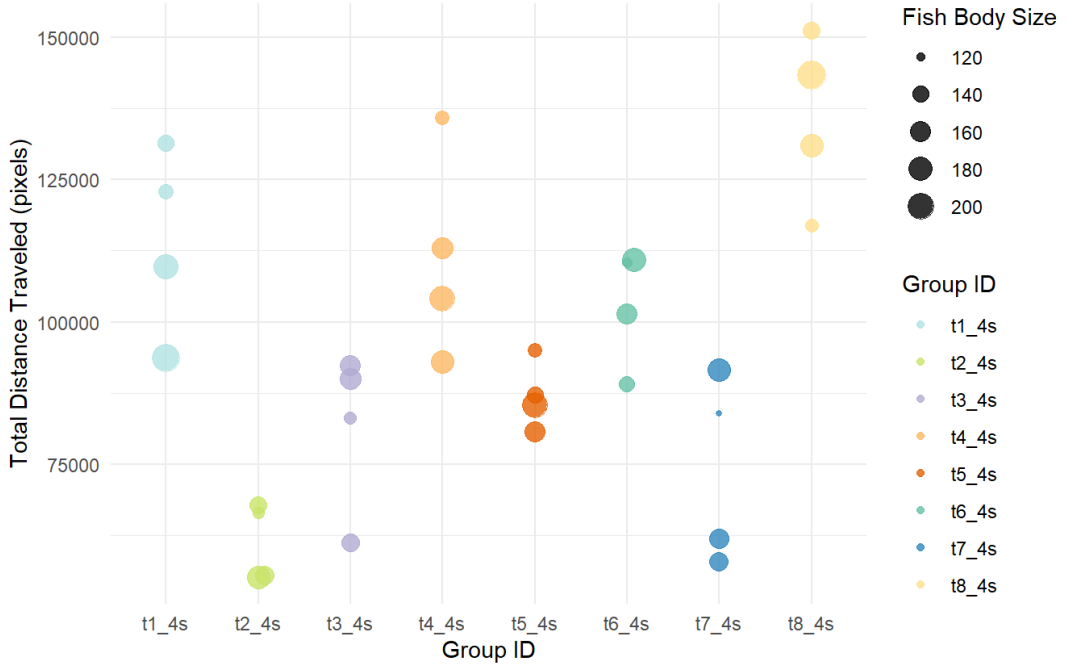


Figure 13. Total distance traveled by group ID for groups of size 4, with ICC = 0.748.

5 Discussion

Using the full 80 fish dataset, we calculated the total distance traveled by each fish. Results varied by group size, with individuals in larger groups (triads, tetrads) traveling significantly farther than those in smaller groups (individuals, pairs). Intraclass Correlation Coefficient (ICC) values were used to compare between-group and within-group variation. While all groups had high ICC scores (more variability between groups than within a given group), pairs had the highest. This indicates that pairs displayed the greatest within-group consistency in total distances traveled.

5.1 Contextualizing Findings

Other studies provide context for our findings on total distance traveled. Fu [6] examined the effect of group size on the synchronization of swimming speeds and angles in two cyprinid species. Fu found that as group size increased, synchronous swimming activity decreased in Chinese bream but remained constant in the qingbo, indicating species-specific effects. Furthermore, Fu reports that as group size increased, so too did swimming speed in both species. An interpretation of this is that members within larger groups may display increased boldness, decreased vigilance, and more varied movement. Our results are consistent with Fu, as faster moving individuals must cover greater distances overall.

Researchers have also observed that polarization, or the degree of alignment between individuals, is negatively correlated with group size [12]. This suggests that larger groups display less coordinated activity, greater variation in movement, and more nuanced patterns of social interaction.

5.2 Automated Tracking Works

SLEAP's skeleton based labeling system enables it to perform remarkably well on relatively little training data. SLEAP prioritizes specificity over generalizability, and models are trained on and for a specific species. While large datasets exist for traditional segmentation tasks, such labeled datasets do not exist for cichlids. This is where SLEAP excels. In our case, we were able to train our models on 1610 annotated frames, which were sampled from longer videos. SLEAP is distinct from other machine learning algorithms, as there is an expectation of a human-in-the-loop throughout the process. This human annotator provides feedback on errors made during inference, allowing for iterative improvements to model performance. With this training framework, we tracked cichlids across our 1.15 million frame dataset.

5.3 Limitations

Specificity

SLEAP thrives on narrowly distributed datasets. This is ideal for controlled environments, such as scientific laboratories. Yet, this specificity renders models sensitive to variation across datasets. Models need not only be species specific, but they also need to be trained on representative members of that species. In challenging frames with many occlusions and overlaps, it is necessary that a human stays in the loop to correct any identity swaps. An identity swap is when the labels for two fish swap due to overlaps or occlusions. Fortunately, SLEAP provides confidence scores that help guide the corrective process. In our study, scalability is limited because errors propagate once an identity swap has occurred.

Distinguishing the Indistinguishable

When identity swaps did occur, closer examination would often reveal that one fish was partially, if not entirely, occluded by another. Without additional temporal context (i.e.,

prior frames), we found that we, as human annotators, tended to struggle on the very same frames that were challenging for the model. This suggests that, on a frame-by-frame basis, the model was able to achieve near-human accuracy. Yet, it also demonstrates the complexities of human perception and the importance of temporal context for reliable sensory integration.

One might assume that the solution to identity swaps is to use some sort of 'flow' tracking across frames, perhaps focusing on cosine similarities or trajectory comparison across frames, yet the rapid dynamics of fish movement complicate matters. For instance, in a video shot at 30 fps, a fish is capable of making a 180-degree twist within 2 frames.

Barreiros (2021) [1] propose using Kalman filters and YOLOv2 to reconnect fragmented zebrafish trajectories but note that a frame rate of 30fps is slow for a fish, reducing the reliability of trajectory-dependent approaches. Other approaches rely upon visual distinguishability between subjects, yet cichlids are not visually distinguishable enough to train models based upon individual differences in appearance. In our case, slight variations in size helped validate model performance, but this will not always be the case. This is where manual approaches to tracking, such as tattoo-ink based injections, may complement automated ones in reducing error propagation.

Temporal Sensitivity

We found that when working with SLEAP, both the selection of labeled frames and the timing of their introduction to the model were critical. In early trials, model weights were initialized on less complex frames, those with only two fish and minimal overlap. This led to overtraining on simplistic scenarios. While the model can learn what fish look like under ideal conditions quite easily, it tends to struggle in more complex scenarios—those that involve overlaps, occlusions, and many fish. When training was restricted to only the more complex and challenging frames (e.g., those with four overlapping or occluded fish), we observed a substantial improvement in model performance. This restriction reduced the frequency with which identity swaps occurred. Nonetheless, model loss had a tendency to plateau when arbitrarily more annotated frames were introduced to the model.

5.4 Next Steps

Our next steps can be summarized as follows:

1. Extend our analysis to the full dataset.
2. Manually score behaviors to demonstrate the utility of automated tracking.
3. Explore additional behavioral metrics.

One such metric would be trajectory alignment, as this may reflect shoaling behavior and provide us with a deeper understanding of group cohesion. Trajectory alignment could be implemented using dynamic time warping, an algorithm with the capability to align paths over time.[2]

Finally, we would like to extend our investigation to other automated tracking frameworks, such as SAM2. SAM2 leverages vision transformers for segmentation and is capable of retaining memory of occluded subjects across many frames of video.[23]

6 Conclusion

This study demonstrates the promise of automated tracking in *A. burtoni* using a deep learning framework. With relatively little annotated training data, SLEAP was successful in identifying individuals and tracking their movement across frames. From the tracked identities, we analyzed tabular data of pose-estimation data with custom Python code. We reconstructed fish skeletons using the coordinates of tracked body parts, explored techniques like KDE and wavelets, and computed a variety of metrics related to spatial preferences. Our results demonstrate the promise of automated tracking for behavioral analysis in revealing layers of social complexity.

7 Acknowledgements

We thank the Department of Natural Sciences and the National Science Foundation (IOS-2341006) for funding to TKSL and NSF summer funding to EAM. Thank you to Andrew Yuan and Veronica Britton for the filming assistance and to the members of the Solomon-Lane Lab for the help in handling *real, live* fish.

8 Appendix

8.1 Supplemental Figures

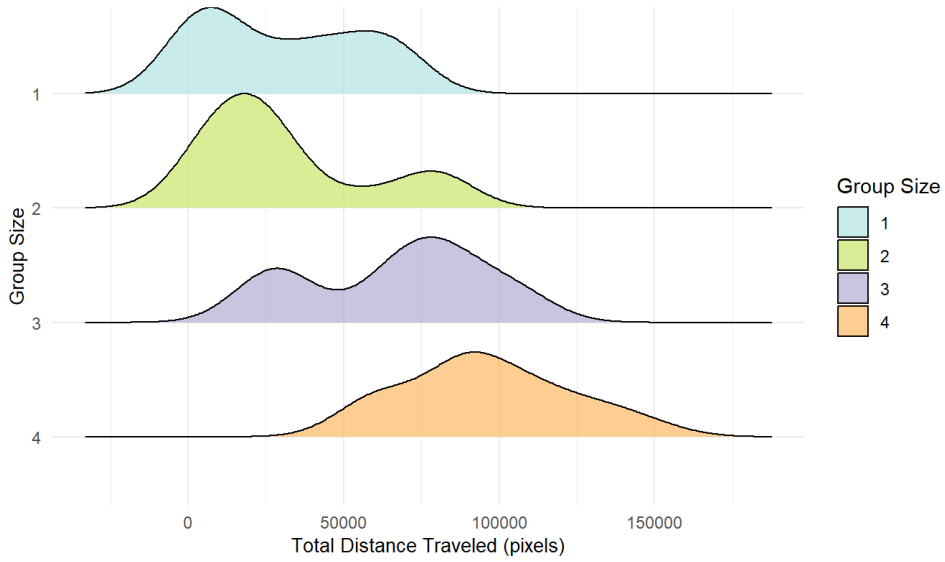


Figure 14. Ridgeline plot of distance traveled by group size

8.2 Data Availability

Source code available at https://github.com/erikamoore/onefishtwofish or upon request.

References

[1] Marta de Oliveira Barreiros, Diego de Oliveira Dantas, Luís Claudio de Oliveira Silva, Sidarta Ribeiro, and Allan Kardec Barros. Zebrafish tracking using YOLOv2 and Kalman filter. *Scientific Reports*, 11(1):3219, February 2021.

[2] Donald J. Berndt and James Clifford. Using dynamic time warping to find patterns in time series. In *Proceedings of the 3rd international conference on knowledge discovery and data mining*, pages 359–370, 1994.

[3] Stefanos Bonovas and Daniele Piovani. Simpson's Paradox in Clinical Research: A Cautionary Tale. *Journal of Clinical Medicine*, 12(4):1633, February 2023.

[4] Jacob R. Bumgarner, Darius D. Becker-Krail, Rhett C. White, and Randy J. Nelson. Machine learning and deep learning frameworks for the automated analysis of pain and opioid withdrawal behaviors. *Frontiers in Neuroscience*, 16, September 2022. Publisher: Frontiers.

[5] J. L. Conrad, K. L. Weinersmith, T. Brodin, J. B. Saltz, and A. Sih. Behavioural syndromes in fishes: a review with implications for ecology and fisheries management. *Journal of Fish Biology*, 78(2):395–435, February 2011.

[6] Sj Fu. Effects of group size on schooling behavior in two cyprinid fish species. *Aquatic Biology*, 25:165–172, December 2016.

[7] C. S. Hall. Emotional behavior in the rat. I. Defecation and urination as measures of individual differences in emotionality. *Journal of Comparative Psychology*, 18(3):385–403, 1934. Place: US Publisher: Williams & Wilkins Company.

[8] Isabela P. Harmon, Emily A. McCabe, Madeleine R. Vergun, Julia Weinstein, Hannah L. Graves, Clare M. Boldt, Deijah D. Bradley, June Lee, Jessica M. Maurice, and Tessa K. Solomon-Lane. Multiple behavioral mechanisms shape development in a highly social cichlid fish. *Physiology & Behavior*, 278:114520, May 2024.

[9] Emerald U Henry, Onyeka Emebo, and Conrad Asotie Omonhinmin. Vision Transformers in Medical Imaging: A Review.

[10] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross Girshick. Segment Anything. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 3992–4003, Paris, France, October 2023. IEEE.

[11] Jessy Lauer, Mu Zhou, Shaokai Ye, William Menegas, Steffen Schneider, Tanmay Nath, Mohammed Mostafizur Rahman, Valentina Di Santo, Daniel Soberanes, Guoping Feng, Venkatesh N. Murthy, George Lauder, Catherine Dulac, Mackenzie Weygandt Mathis, and Alexander Mathis. Multi-animal pose estimation, identification and tracking with DeepLabCut. *Nature Methods*, 19(4):496–504, April 2022. Publisher: Nature Publishing Group.

[12] Noam Miller and Robert Gerlai. From Schooling to Shoaling: Patterns of Collective Motion in Zebrafish (Danio rerio). *PLoS ONE*, 7(11):e48865, November 2012.

[13] Ran Nathan, Christopher T. Monk, Robert Arlinghaus, Timo Adam, Josep Alós, Michael Assaf, Henrik Baktoft, Christine E. Beardsworth, Michael G. Bertram, Allert I. Bijleveld, Tomas Brodin, Jill L. Brooks, Andrea Campos-Candela, Steven J. Cooke, Karl Ø. Gjelland, Pratik R. Gupte, Roi Harel, Gustav Hellström, Florian Jeltsch, Shaun S. Killen, Thomas Klefoth, Roland Langrock, Robert J. Lennox, Emmanuel Lourie, Joah R. Madden, Yotam Orchan, Ine S. Pauwels, Milan Říha, Manuel Roeleke, Ulrike E. Schlägel, David Shohami, Johannes Signer, Sivan Toledo, Ohad Vilk, Samuel Westrelin, Mark A. Whiteside, and Ivan Jarić. Big-data approaches lead to an increased understanding of the ecology of animal movement. *Science*, 375(6582):eabg1780, February 2022. Publisher: American Association for the Advancement of Science.

[14] Heike Neumeister, Mila Adelman, William Gallagher, Jiangtao Gou, Karin Merrins, Melissa Perkowski, Stephanie Shih, Beth Terranova, and Thomas Preuss. Socially induced plasticity in sensorimotor gating in the African cichlid fish Astatotilapia burtoni. *Behavioural Brain Research*, 332:32–39, August 2017.

[15] Talmo D. Pereira, Nathaniel Tabris, Arie Matsliah, David M. Turner, Junyu Li, Shruthi Ravindranath, Eleni S. Papadoyannis, Edna Normand, David S. Deutsch,

Z. Yan Wang, Grace C. McKenzie-Smith, Catalin C. Mitelut, Marielisa Diez Castro, John D'Uva, Mikhail Kislin, Dan H. Sanes, Sarah D. Kocher, Samuel S.-H. Wang, Annegret L. Falkner, Joshua W. Shaevitz, and Mala Murthy. SLEAP: A deep learning system for multi-animal pose tracking. *Nature Methods*, 19(4):486–495, April 2022.

[16] Leo Polansky, Iain Douglas-Hamilton, and George Wittemyer. Using diel movement behavior to infer foraging strategies related to ecological and social factors in elephants. *Movement Ecology*, 1(1):13, December 2013.

[17] Laetitia Prut and Catherine Belzung. 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, February 2003.

[18] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical Text-Conditional Image Generation with CLIP Latents, April 2022. arXiv:2204.06125 [cs].

[19] Caroline A. Schneider, Wayne S. Rasband, and Kevin W. Eliceiri. NIH Image to ImageJ: 25 years of image analysis. *Nature Methods*, 9(7):671–675, July 2012. Publisher: Nature Publishing Group.

[20] Ryan Schwark, Simon Ogundare, Caleb Weinreb, Preston Sheng, William Foster, Andre Toussaint, Phalaen Chang, Yu-Young Wesley Tsai, Maxmillian Comfere, Amanda Arnold, Antonella Guadagnino, Daniel McCloskey, Evan Schaffer, Kanaka Rajan, Sandeep Robert Datta, and Ishmail Abdus-Saboor. Behavioral fingerprinting of the naked mole-rat uncovers signatures of eusociality and social touch, February 2024. Pages: 2024.02.21.581483 Section: New Results.

[21] Tessa K. Solomon-Lane and Hans A. Hofmann. Early-life social environment alters juvenile behavior and neuroendocrine function in a highly social cichlid fish. *Hormones and Behavior*, 115, 2019. Place: Netherlands Publisher: Elsevier Science.

[22] Rose Trappes. How tracking technology is transforming animal ecology: epistemic values, interdisciplinarity, and technology-driven scientific change. *Synthese*, 201(4):128, March 2023.

[23] Jovana Videnovic, Alan Lukezic, and Matej Kristan. A Distractor-Aware Memory for Visual Object Tracking with SAM2, December 2024. arXiv:2411.17576 [cs].