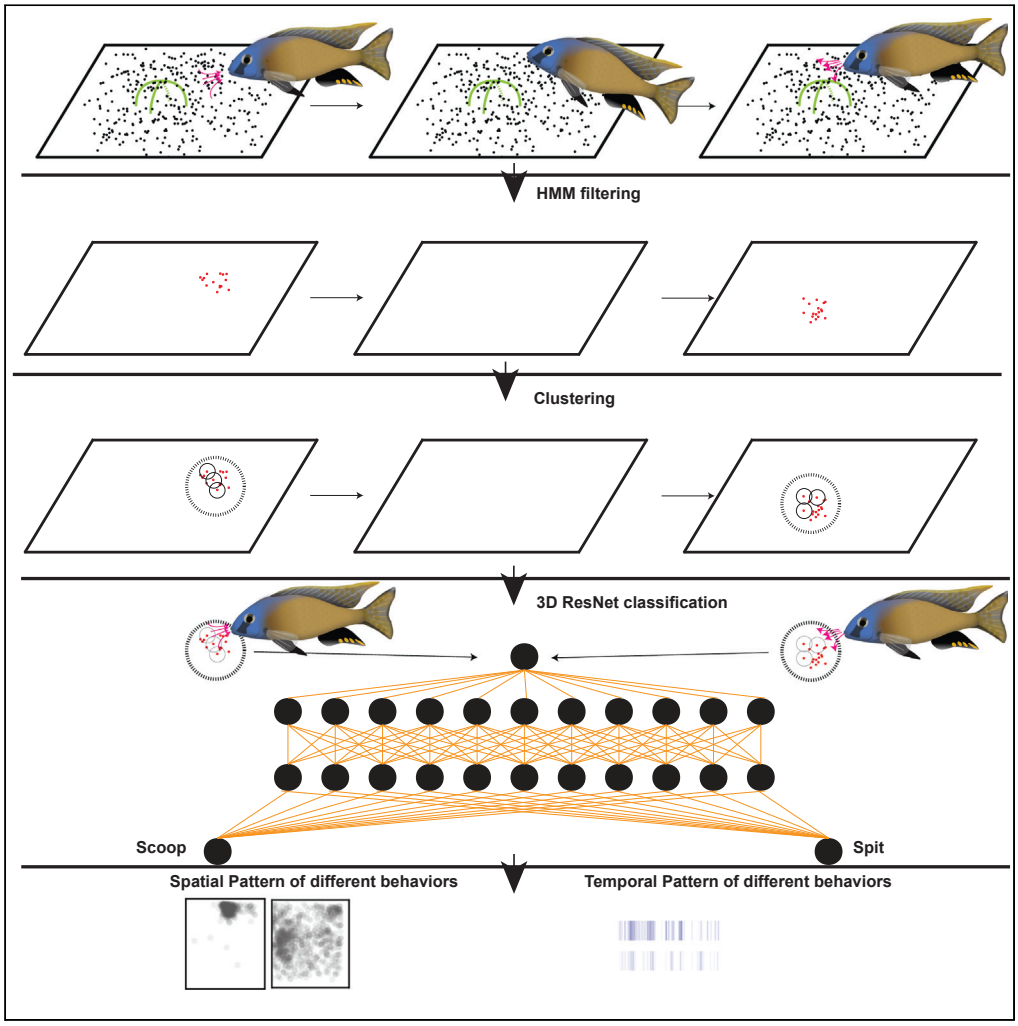
iScience

CellPress
OPEN ACCESS

Article

Automatic Classification of Cichlid Behaviors Using 3D Convolutional Residual Networks



Lijiang Long, Zachary V. Johnson, Junyu Li, Tucker J. Lancaster, Vineeth Aljapur, Jeffrey T. Streelman, Patrick T. McGrath

todd.streelman@biology.gatech.edu (J.T.S.)
patrick.mcgrath@biology.gatech.edu (P.T.M.)

HIGHLIGHTS

A dataset of more than 14,000 annotated animal behavior videos was created

3D residual networks can be used to classify animal behavior

Different intents of similar behavioral actions can be distinguished

A working solution to study long-term behaviors was established

Long et al., iScience *23*, 101591
October 23, 2020 © 2020
https://doi.org/10.1016/j.isci.2020.101591

iScience

CellPress
OPEN ACCESS

Article

Automatic Classification of Cichlid Behaviors Using 3D Convolutional Residual Networks

Lijiang Long,[1,2,5] Zachary V. Johnson,[1,5] Junyu Li,[1] Tucker J. Lancaster,[1,2] Vineeth Aljapur,[1] Jeffrey T. Streelman,[1,3,*] and Patrick T. McGrath[1,3,4,6,*]

SUMMARY

Many behaviors that are critical for survival and reproduction are expressed over extended time periods. The ability to inexpensively record and store large volumes of video data creates new opportunities to understand the biological basis of these behaviors and simultaneously creates a need for tools that can automatically quantify behaviors from large video datasets. Here, we demonstrate that 3D Residual Networks can be used to classify an array of complex behaviors in Lake Malawi cichlid fishes. We first apply pixel-based hidden Markov modeling combined with density-based spatiotemporal clustering to identify sand disturbance events. After this, a 3D ResNet, trained on 11,000 manually annotated video clips, accurately (>76%) classifies the sand disturbance events into 10 fish behavior categories, distinguishing between spitting, scooping, fin swipes, and spawning. Furthermore, animal intent can be determined from these clips, as spits and scoops performed during bower construction are classified independently from those during feeding.

INTRODUCTION

Animals respond to and interact with their environment using a rich repertoire of behavioral actions. A major challenge for neuroscientists is to understand how neural circuits coordinate these behaviors in response to sensory and internal stimuli. Automated identification and classification of behavioral actions will aid in this task, as most neural responses are stochastic, requiring a large number of replicates to accurately estimate relationships with complex behaviors (Egnor and Branson, 2016). In addition, many behaviors are executed over long timescales through the accumulated actions of thousands of individual decisions (e.g., foraging, construction, and social behaviors), making manual analysis of the full course of behavior entirely impractical (Russell et al., 2017; Feng et al., 2015; Mouritsen, 2018; Tucker, 1981).

One common approach is to manually observe snapshots of long-term behaviors over extended periods of time. This method is labor intensive and thus can severely limit the total number of animals that can be measured in a single study. Furthermore, it cannot provide a complete and detailed quantitative description of the full behavioral trajectory. An alternative approach is to design abbreviated assays that elicit the behavior of interest during a short period of time. One issue, however, is that many natural behaviors may be expressed differently over short timescales, and/or in unnatural/unfamiliar environments. Recently, deep learning approaches have revolutionized our ability to automatically analyze video and image data (Nath et al., 2019; Mathis et al., 2018; Weissbrod et al., 2013; Wild et al., 2018). Convolutional neural networks (CNNs) can be applied to images for the purpose of object detection, identifying all animals within an individual frame (Girshick, 2015; Ren et al., 2015; Redmon et al., 2016; Pereira et al., 2019). CNNs can also identify body parts, such as eyes, legs, or wings, allowing for the determination of an animal's posture at any specific time (Graving et al., 2019; Pereira et al., 2019; Kain et al., 2013; Petrou and Webb, 2012; Gunel et al., 2019; Andriluka et al., 2018; Nath et al., 2019). Although position and pose alone are not sufficient for defining animal behavior, postural time series can be used to define some behavioral actions. Such analysis has, for example, been used to quantitatively describe different types of stereotyped movements in *Drosophila* flies, such as different locomotor and grooming behaviors (Berman et al., 2014). Similar approaches have also been used successfully on other species (Stephens et al., 2008).

[1]School of Biological Sciences, Georgia Institute of Technology, Atlanta, GA 30332, USA

[2]Interdisciplinary Graduate Program in Quantitative Biosciences, Georgia Institute of Technology, Atlanta, GA 30332, USA

[3]Parker H. Petit Institute of Bioengineering and Bioscience, Georgia Institute of Technology, Atlanta, GA 30332, USA

[4]School of Physics, Georgia Institute of Technology, Atlanta, GA 30332, USA

[5]These authors contributed equally

[6]Lead Contact

*Correspondence: todd.streelman@biology.gatech.edu (J.T.S.), patrick.mcgrath@biology.gatech.edu (P.T.M.)

https://doi.org/10.1016/j.isci.2020.101591

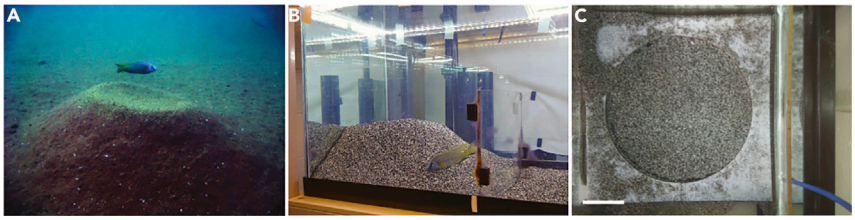


Figure 1. Measurement of Lake Malawi Cichlid Bower Behaviors in Laboratory Aquariums

(A–C) Approximately 200 species of Lake Malawi cichlids exhibit bower behaviors. In these species, sociosexual cues trigger reproductive adult males to construct large courtship structures by manipulating sand with their mouths. The geometric structure of the bower is species-specific. (A) Example of a castle structure built in Lake Malawi. (B) Example of a castle structure built in a standard aquatics facility aquarium. (C) Top down view of acrylic tray used to constrain bower building to a third of the tank. Video recordings using this view were used to characterize bower building behaviors throughout this article. Scale bar, 10 cm. Photo credit to Dr. Isabel Magalhaes (A).
See also Figure S1.

It remains challenging, however, to translate changes in position and posture into complex behaviors that are characterized by an animal's interaction with the environment. For example, many goal-directed behaviors involve significant manipulation of the physical environment in ways that are essential for survival or reproductive success, such as mice digging burrows, birds building nests, spiders weaving webs, and bowerbirds or cichlid fishes constructing courtship bowers (Hansell, 2000; Benjamin and Zschokke, 2000; Vollrath, 1992; Collias and Collias, 2014; Dawkins, 1982; McKaye et al., 2001). In such cases, information about changes to the physical environment itself is essential to fully describe the behavior.

One possible solution for analysis of these types of behaviors is to train a deep learning network that takes in videos as input and then outputs a prediction for the corresponding behavior type. For example, 3D Residual Networks (3D ResNets) have been successfully used to classify human behaviors, distinguishing between hundreds of different action classes (Hara et al., 2018; Qiu et al., 2017). These deep learning networks use 3D kernels with the ability to extract spatiotemporal features directly from videos. Videos are fed into these networks raw, without any individual body posture information beyond what can be learned from the training data. One major benefit of these 3D networks, when compared with networks that process each frame individually, is that they can integrate spatial and temporal information to recognize changes in the animal's environment that might indicate a particular behavior (e.g., digging, feeding, or construction behaviors). However, a significant challenge in applying action recognition to large videos is detecting actions of interest in a time and space frame, which is known as shot transition detection. This requires splitting large (e.g. 10-h-long) videos into small enough temporal units such that each unit contains only one action of interest and excludes as much irrelevant information as possible.

In this article, we describe an approach for analyzing construction, feeding, and mating behaviors from hundreds of hours of videos of Lake Malawi cichlids behaving freely in naturalistic and social home tank environments. Lake Malawi is the most species-rich freshwater lake on the Earth, and it contains 700–1,000 cichlid species that have rapidly evolved in the past 1–2 million years (Brawand et al., 2014). Approximately 200 of these species express long-term bower construction behaviors, in which males manipulate sand to create large courtship structures, or bowers, to attract female mates (York et al., 2015) (Figure 1A). Males construct pits and castles over the course of many days and make thousands of decisions about where to scoop up and then spit out mouthfuls of sand. Bower construction is therefore a useful model for understanding how goal-directed behaviors are executed over long time periods in environments that are physically and socially dynamic.

To measure bower construction, we first develop an action detection algorithm that uses hidden Markov models (HMMs) and density-based spatial clustering to identify regions of the video where the fish has manipulated sand using its mouth, fins, or other parts of its body. Then, after generating small video clips that encompass these events, we use a 3D ResNet to classify each sand disturbance event into one of ten action categories. We demonstrate that this approach can be used to quickly, accurately, and automatically identify hundreds of thousands of behaviors across hundreds of hours of video. Through this approach, we measure the times and locations of construction, mating, and feeding behaviors expressed over the course of many days in three different species and one hybrid cross.

Species	Type	n	Bower Shape	Training Set	Description
–	Empty	3	–	No	No fish in tank
CV	Feeding only	2	–	No	Four female fish
MC	Feeding only	2	–	No	Four female fish
TI	Feeding only	1	–	No	Four female fish
CV	Building	1	Pit	Yes	One male and four female fish
MC	Building	3	Castle	Yes[a]	One male and four female fish
TI	Building	2	Pit	Yes	One male and four female fish
MC/CV F1	Building	2	Pit/castle[b]	Yes	One male and four female fish

Table 1. Summary of Video Recordings Used in This Report

CV, Copachromis virganialis; MC, Mchenga conophoros; TI, Tramitichromis intermedius

[a]Only two of the three MC trials was manually labeled for training.

[b]F1 hybrids display codominant phenotype. Males initially build a pit structure and then transition to build a castle structure nearby the original pit.

RESULTS

Collection of Video Recordings of Male Cichlid Fish Constructing Bowers

If provided access to an appropriate type of sand and gravid females, Lake Malawi male cichlids will typically construct species-typical bower structures in standard aquarium tanks within 1–2 weeks (York et al., 2015) (Figure 1B). After testing sand from different suppliers, we found that males construct most vigorously with a sand mixture composed of black and white grains. Only half of each home tank was accessible for top-down video recording, due to physical constraints such as support beams and water supply/drain lines. To restrict sand manipulation behaviors to the video-accessible region of each tank, we placed a custom-built acrylic tray containing sand directly within the video camera field of view. We then introduced a single male individual to four female individuals (Figure 1C). For each trial, we collected 10 h of video daily for approximately 10 consecutive days, resulting in the collection of ~100 h of video per trial. We analyzed two types of trials: construction trials, in which males constructed bowers in the presence of four females, and control trials, in which tanks were empty or tanks contained four females but no male. We analyzed eight construction trials, encompassing three different species and one F1 hybrid cross: pit-digger *Copadichromis virginalis*, CV, n = 1; castle-builder *Mchenga conophoros*, MC, n = 3; pit-digger *Tramitichromis intermedius*, TI, n = 2; and pit-castle hybrid MCxCV F1, n = 2. Visual inspection confirmed that each male built a species-typical bower. We also analyzed three empty tank control trials, and five female-only control trials. Table 1 summarizes all trials that were analyzed in this article.

To initially analyze these videos, we first focused on a single MC trial, identifying 3 days when castle building occurred. A trained observer manually annotated all spit and scoop events for a single hour on each day. In total, 1,104 spit events and 1,575 scoop events were observed (there is not a 1:1 correspondence between spit and scoop events because males sometimes scoop sand from multiple locations before performing a single spit) (Figure S1). Analysis of these 3 h of videos required 36 h of human time, or a 12:1 ratio of human time to video time. Full analysis of a *single* trial would require 1,200 h of human time, or approximately 30 weeks of full-time work dedicated to manual annotation. Automated analysis of these videos was thus necessary for the full characterization of bower construction even in this limited set of trials, let alone any larger-scale investigation of bower behavior.

Automatic Identification of Sand Disturbance Events

In the process of manually annotating these videos, we noticed that spit and scoop events resulted in an enduring and visually apparent spatial rearrangement of the black and white grains of sand. Scoop and

spit events were associated with a permanent transition between pixel values. To test whether events could be identified despite the frequent occurrences of fish swimming over the sand, we inspected local regions in which a large number of pixels underwent permanent transitions. We observed temporary changes in pixel values when fish or shadows occluded the sand and permanent changes in pixel values at the locations of sand disturbance events (Figures 2A–2F). Consistent with this, we plotted the grayscale value of every pixel over entire 10-h videos and found that individual pixel values were, in general, fixed around a specific mean value for extended periods of time, but showed small oscillations and large but temporary deviations about this value across video frames (Figures 2G and 2H). In addition, we identified permanent transitions in pixel value, in which the mean grayscale value of a single pixel would change to, and retain, a new value.

To automatically identify permanent transitions in pixel color, we used an HMM to calculate a hidden state for each pixel that represented its current color. In general, the HMM partitioned each pixel into a small set of values (~10–50) over the course of each 10-h video (Figures 2G and 2H). In testing the speed and accuracy of the HMM on raw data, we observed that large temporary deviations greatly impacted the accuracy of the HMM calls, potentially due to their violation of the assumption of a Gaussian distribution. Therefore, these large deviations were excluded using a rolling mean filter. We also found a speed/resolution trade-off to the frame rate. As we wanted to perform this analysis for the entire 1,296 × 972-pixel video, we needed to analyze over 1 million pixels through time. We found that 1 frame per second, sampled from the 30 frames per second in the raw video, was a reasonable trade-off—an entire video could be analyzed in approximately 2 h on a 24-core machine. After setting appropriate parameters, manual inspection of the HMM fits showed reasonable agreement with our expectations: enduring transitions in pixel values were associated with sand disturbance by fish, and three orders of magnitude more of these transitions were detected in tanks with fish versus empty tanks (Figure 2I). In addition to identifying when and where a sand disturbance occurs, this approach also allowed us to create background images that excluded fish and shadows from each frame (Figure 2J).

Clustering of HMM Changes to Detect Sand Disturbance Events

When a fish scoops up or spits out sand, HMM transitions occur in hundreds to thousands of nearby pixels. To group individual pixel transitions together, we applied density-based spatial clustering to the x, y, and time coordinate of each individual HMM transition (Ester et al., 1996) (Figure 3A). To find the best parameters, we used exploratory data analysis and parameter grid search (Figure S2). This approach allowed us to determine the spatial and temporal location of thousands of individual fish-mediated sand disturbances on each day of each bower trial. We also calculated distributions for the width, height, temporal span, and number of HMM transitions for every cluster (Figure S3). These data further demonstrated that clusters were associated with the presence of fish, as empty tanks showed a minimal number of clusters (Figure 3B). Inspection of a subset of these clusters confirmed that the clusters were associated with regions of sand that were undergoing lasting change (Figure 3C).

To further validate these clusters, we generated 200 × 200-pixel, 4-s video clips for a randomly sampled subset of clusters centered over their mean spatial and temporal position (~2,000 per trial for seven of the eight bower construction trials). Manual review of these video clips revealed that the majority (>90%, 13,288/14,234 analyzed events) of clusters were true sand change events caused by fish behaviors, with the remaining portion including reflections of events in the glass, shadows caused by stationary or slow-moving fish, or small bits of food, feces, or other debris settling on the sand surface.

To ensure this procedure recovered most sand disturbance events, we leveraged our manually annotated spit/scoop dataset (Figure S1) and performed HMM and density-based spatial clustering. For each event, the spatial and temporal center of the event was compared. The differences in time between human annotations and machine annotations follow a Gaussian distribution (Figure S4A). 95.6% (725/758) human annotations have at least one machine annotation in the (-1s, +1s) interval. Of these 725 events, 93.7% (679/725) also had a machine annotation event within 70 pixels (approximately 3.5 cm), which is approximately one-third of the oval fish length (Figure S4B). By these criteria, 89.6% (679/758) of human-annotated events can be retrieved by the automated machine HMM/clustering process.

Automatic Classification of Cichlid Behaviors with 3D Residual Networks

On average, ~1,000 clusters were identified in each hour of video (Figure 3B). We aimed to automatically identify the subset of these events corresponding to bower construction behaviors in each video. However,

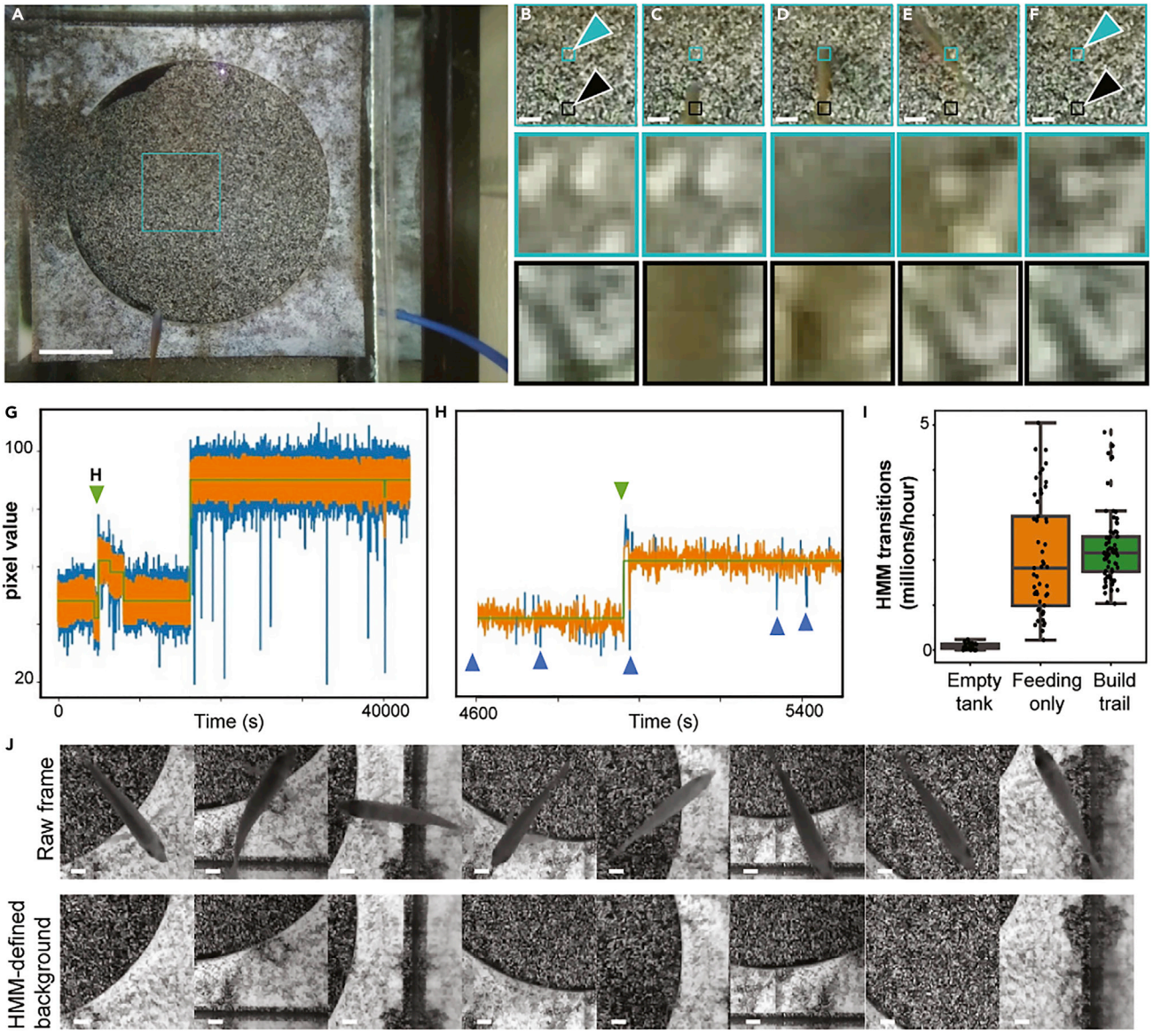


Figure 2. Automated Detection of Sand Change from Video Data

(A–H) The sand in this behavioral paradigm is composed of black and white grains (e.g., as seen in A), and therefore sand manipulation events during bower construction cause permanent rearrangement of the black and white grains at specific locations. We aimed to detect these events by processing whole video frames (A, with turquoise box indicating an example region of interest. Scale bar, 10 cm) sampled once per second and tracking the values of individual pixels throughout whole trials. Fish swimming over sand cause transient changes in pixel values (e.g., B–F, black arrows indicate an example location of a fish swimming over the sand; the bottom row depicts a zoomed in 20 × 20-pixel view of a location that the fish swims over, sampled from representative frames across 4 s). In contrast, sand manipulation behaviors cause enduring changes in pixel values (e.g., B–F, turquoise arrows indicate an example location of a fish scooping sand. Scale bar, 2cm; the middle row contains a zoomed in 20 × 20-pixel view of a location where the fish scoops sand). We used a custom hidden Markov model to identify all enduring state changes for each pixel throughout entire videos (G, green line indicates HMM-predicted state, orange line indicates raw grayscale pixel value, and blue lines indicate transient fluctuations beyond the pixel's typical range of values likely caused by fish swimming or shadows). Because fish swim over the sand frequently, a large number of transient changes are ignored (e.g., pixel value fluctuations indicated by blue arrows in H), while enduring changes are identified (e.g., pixel value change indicated by green arrow in (H).

(I) Number of HMM transitions identified per hour based on trial type. "Feeding only" are trials containing four females. "Build trial" contains four females and one male that builds a bower. The boxplot shows quartiles of the dataset while the whiskers show the rest of the distribution unless the point is an outlier.

(J) The HMM could be used to calculate a background image at a given time point, resulting in removal of the fish and the associated shadow from the image. Scale bar, 1cm.

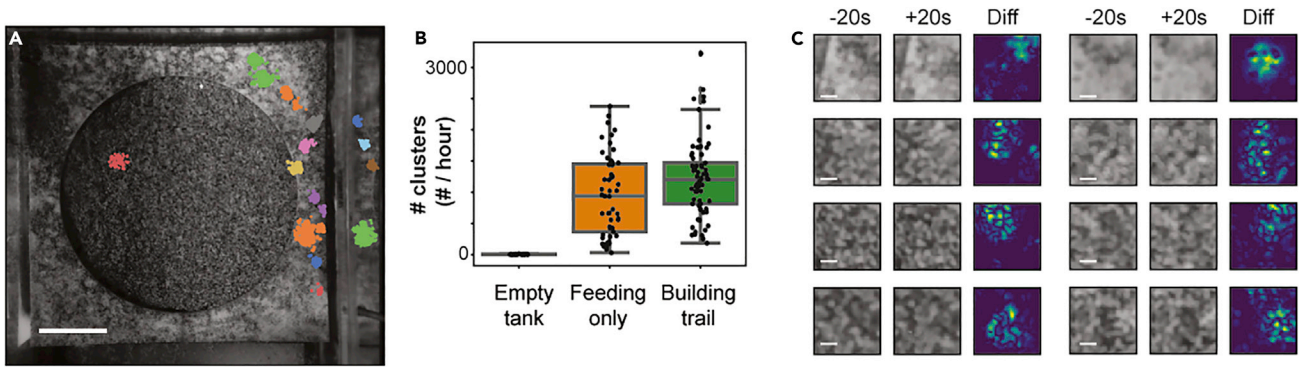


Figure 3. Clustering HMM Events Identifies Sand Disturbance Events

(A) Example of clusters identified in a 60 s period. HMM transitions are color coded based on their cluster membership. Scale bar, 10 cm.

(B) Number of clusters identified per hour based on trial type. "Feeding only" are trials containing four females. "Build trial" contain four females and one male that builds a bower. The boxplot shows quartiles of the dataset while the whiskers show the rest of the distribution unless the point is an outlier.

(C) Before and after images (20 s) for eight example clusters. Left and middle panels show raw grayscale images. Right panel shows heatmap displaying pixel value differences in the left and middle frames. Yellow indicates large changes. Blue indicates no change. Scale bar, 1 cm.

See also Figures S2–S4.

scooping and spitting sand during bower construction represents only a subset of behaviors that cause sand change in our paradigm. For example, feeding behaviors performed by both males and females also involve scooping and spitting sand, and are expressed frequently throughout trials. Quivering and spawning behavior, in which a male rapidly circles and displays his fins for a gravid female, are less frequent, and also cause significant sand change. Identifying bower construction behaviors therefore requires identifying other behaviors that cause sand change. To achieve this, we turned to a deep learning approach and assessed whether 3D Residual Networks (3D ResNets), which have been recently shown to accurately classify human actions from video data (Qiu et al., 2017), could accurately distinguish fish behaviors that cause sand change in our paradigm.

To create a training set for the 3D ResNet, we first generated a video clip for each cluster based on its location in space and time. This process narrowed down each event to a 4-s, 200 × 200-pixel video clip from the original 1,296 × 972-pixel video. To create a training set for the 3D ResNet, a trained observer manually classified 14,172 of these video clips (~2,000 per trial) into one of ten categories (bower scoop, bower spit, bower multiple, feed scoop, feed spit, feed multiple, drop sand, quivering, other, and shadow/reflection) (Videos S1, S2, S3, S4, S5, S6, S7, S8, S9, and S10). Feeding was the most frequently observed behavior, accounting for nearly half of all clips (47.0%, 6,659/14,172 annotated clips; feeding scoops, 15.2%; feeding spits, 11.6%; multiple feeding events, 20.2%). Bower construction behaviors were the next most common (19.5%; bower scoops, 9.4%; bower spits, 8.1%; multiple bower construction events, 1.9%). Quivering and spawning events were the least frequently observed, accounting for just 2.6% of all clips. The remainder of sand change events were annotated as sand dropping behavior (5.6%), "other" behaviors (e.g., brushing the sand surface with the fins or the body; 18.7%), or shadows/reflections (6.6%) (Figure 4).

A 3D ResNet was then trained on 80% (~11,200 clips) of the data, and the remaining 20% of the data was reserved for validation (2,752 clips) (Table 2 and Figures 5A and 5B). To place the ResNet predictions in the context of human performance, we also measured the accuracy of a previously naive human observer who underwent 12 h of training and then manually annotated a test set of 3,039 clips from three trials and all 10 behavior categories (Table S1). The 3D ResNet achieved ~76% accuracy on the validation set, which was better than the accuracy of the newly trained human observer (~73.9% accuracy, 2,246/3,039 clips). Confidence for 3D ResNet predictions on the validation set ranged from 22.1% to 100%, and confidence tended to be greater for correct predictions (mean confidence 92.93% ± 0.279%) than for incorrect predictions (mean confidence 78.28% ± 0.074%) (Figure S5A, C). We found an imbalance in the distribution of incorrect predictions across categories (Table 2). For some categories, such as "build multiple," "feed multiple," and "fish other," video clips could contain behaviors that also fit into other categories. For example, a "feed multiple" clip by definition contains multiple feeding scoop and/or feeding spit events, a "bower multiple" clip contains multiple bower scoops and/or bower spits, and a "fish other" clip may contain a bower scoop and a fin swipe (or some other combination of behaviors). We found that erroneous "within building"

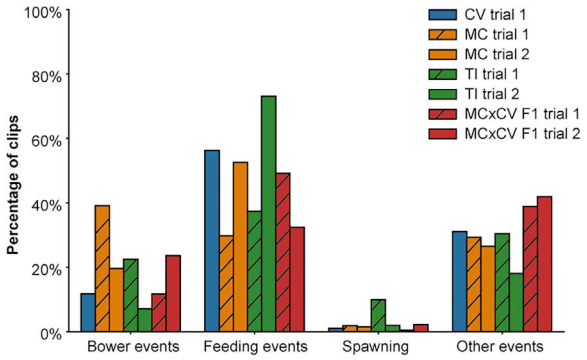


Figure 4. Distribution of Different Sand Perturbation Events

A human observer manually classified 14,234 video clips into one of ten categories. Bower events are scoops, spits, or multiple scoop/spit events associated with bower construction. Feeding events are scoops, spits, or multiple scoop/spit events associated with feeding behaviors. Spawning events involve male fish quivering to attract females. Other events include sand perturbations caused by fins or the body, reflections of events in the aquarium glass, or sand dropped from above.

category predictions for build multiple, "within feeding" predictions for feed multiple, and "fish other" predictions accounted for ~82% of all incorrect predictions. We further found that the area under the precision recall curve was 0.91 (Figure S5B). By setting a confidence threshold of 90%, most (~62%) incorrect predictions were excluded, whereas most (70%) correct predictions were included. At the same time, ~86% of correct bower scoop predictions and ~88% of correct bower spit predictions were included. Of all predictions, 69% were above the 90% confidence threshold, and overall accuracy for these high-confidence predictions was ~87% (Figure S5C).

We tested how many annotated clips are required to achieve a similar accuracy by inputting 5%–50% of the original labeled clips into the training dataset. A similar accuracy was achieved using just 50% of the clips, but a significant decrease in accuracy (75%–62%) was observed when the number of training clips was decreased from 50% to 30% of the original set (Figure 5B).

As there was an uneven distribution of the different categories in the training data, we were curious if our model might be biased toward the more frequent categories and have lower overall accuracy than a model trained on data with a balanced distribution of the different categories. To test this hypothesis, we built a model from an equal sampling of all 10 categories. First, we created an "equal sampling" set where each category in the training dataset has the same number of clips. Because the category "build multiple" has the least number of clips (277 clips out of 14,172 annotated video clips), for each category, we used 220 clips for training and 50 clips for validation. As a "random sampling" control, we randomly sampled an equal number of clips for training and validation as the uniform set. Specifically, there are 2,200 randomly sampled clips for training and 500 for testing. Overall, the accuracy was comparable, although we did note some large differences in accuracy based upon category (Figure S5F). For example, build multiple, a very rare category, was much less accurately predicted in the "random sampling" model comipared to the "equal sampling" model (18% versus 75%). When we compared the overall accuracy of each model, we noted that the higher-performing model depended on the validation set. When an equal sampling of validation clips was used, the equal sampling model performed better (64.0% accuracy compared to 60.5% accuracy). When a random sampling of validation clips was used, the random sampling method was more accurate (62.2% accuracy compared to 57.7% accuracy).

The spatiotemporal dimension of each video clip is a very important hyperparameter to tune to achieve the best classification accuracy. The spatial scale is optimal when the whole behavior is captured, and when irrelevant pixels are excluded from the frame as much as possible. The same is true for temporal scale. Therefore, we decided to test the amount of spatial and temporal data necessary to accurately classify each clip. We found that the classification accuracy initially increased with frame size and peaked at 120 × 120 pixels per frame (Figure S5D). The accuracy decreased from 76% to 74% when the entire 200 × 200-pixel frame was used as input. This decrease in accuracy further affirmed our choice to crop the clips before feeding them to the ResNet, as it eliminates information unrelated to action in the frame.

	Predicted Label													
Human label	Category	Build scoop	Feed scoop	Build spit	Feed spit	Build mult	Feed mult	Spawn	Shadow Reflect	Other	Sand drop	Total	Percent	Acc
	Build scoop	**187**	25	1	0	2	1	0	0	26	2	244	9%	77%
	Feed scoop	23	**303**	0	3	0	48	0	1	28	0	406	15%	75%
	Build spit	1	0	**194**	5	6	1	1	0	10	2	220	8%	88%
	Feed spit	1	6	15	**225**	0	38	2	1	27	26	341	12%	66%
	Build mult	8	0	9	0	**29**	1	1	0	7	0	55	2%	53%
	Feed mult	16	73	4	22	2	**402**	0	0	17	1	537	20	75%
	Spawn	0	0	0	0	1	0	**56**	0	15	1	73	3%	77%
	Shadow/ reflect	0	0	1	1	0	1	0	**169**	11	1	184	7%	92%
	Other	26	26	4	10	10	11	8	9	**388**	20	512	19%	76%
	Sand drop	1	0	1	12	0	1	0	2	14	**149**	180	7%	83%
	Total	263	433	229	278	50	504	68	182	543	202	2,752	100%	76%

Table 2. Confusion Matrix[a] for Sand Disturbance Events Classified with a 3D Residual Network

[a]Each row of the matrix represents the instances in an actual class while each column represents the instances in a predicted class.

Next, we tested the relationship between the number of frames per clip and the classification accuracy. We tuned two parameters, including the total number of frames sampled (n = 16, 32, 48) and the interval between each sampled frame (i = 0, 1, 2, 3, 4). Classification accuracy steadily increased from 67% (n = 16) to 72% (n = 48) when more frames were used. When the total number of frames was fixed, accuracy was greater when the sampling interval (i) was larger (Figure S5E).

Distinguishing between Feeding and Bower Construction Events

The differences between construction behaviors and feeding behaviors are subtle and are often indistinguishable to inexperienced observers (Table S1). Feeding and construction both involve scooping sand into the mouth, swimming, and then spitting sand from the mouth. Feeding behaviors are also typically performed more frequently relative to bower construction behaviors (Figure 4). We were therefore concerned that the 3D ResNet would be unable to accurately distinguish between feeding versus construction behaviors, which would prevent accurate measurement of spatial patterns associated with bower construction.

Much to our surprise, the ResNet reliably distinguished between feeding and construction events; for construction and feeding scoops and spits, the model achieved F1 scores of 0.74, 0.86, 0.72, and 0.73 (for build scoops, build spits, feeding scoop, and feeding spits, respectively), with balanced precision and recall scores. These F1 scores were comparable to the overall averaged F1 score (or accuracy) of the model, 0.76. Remarkably, the model outperformed a newly trained human observer in distinguishing between construction and feeding: the average F1 score for the model across these categories was 0.76, whereas the newly trained observer's average F1 score was 0.71. The difference in performance was further evident when we quantified the proportion of build-feed false-positives (build scoops mis-classified as feed scoops, build spits mis-classified as feed spits, and vice versa); these erroneous predictions were 2.5× more frequent in the newly trained observer's annotations compared with the model's predictions (201/1,469 predictions in these categories, or 13.7% for the human; 68/1,211 predictions in these categories, or 5.6% for the model). Differences in the spatial position and relative timing were readily observed among bower construction, feeding, and spawning events (Figures 5C and 5D).

Accuracy of Model on New Trials

The trained model showed high validation accuracy on the seven trials. However, our ultimate goal is to measure bower construction in hundreds of independent trials. To that end, we tested how generalizable this model is when applied to previously unobserved individuals. To test this, we retrained models using six of the seven trials

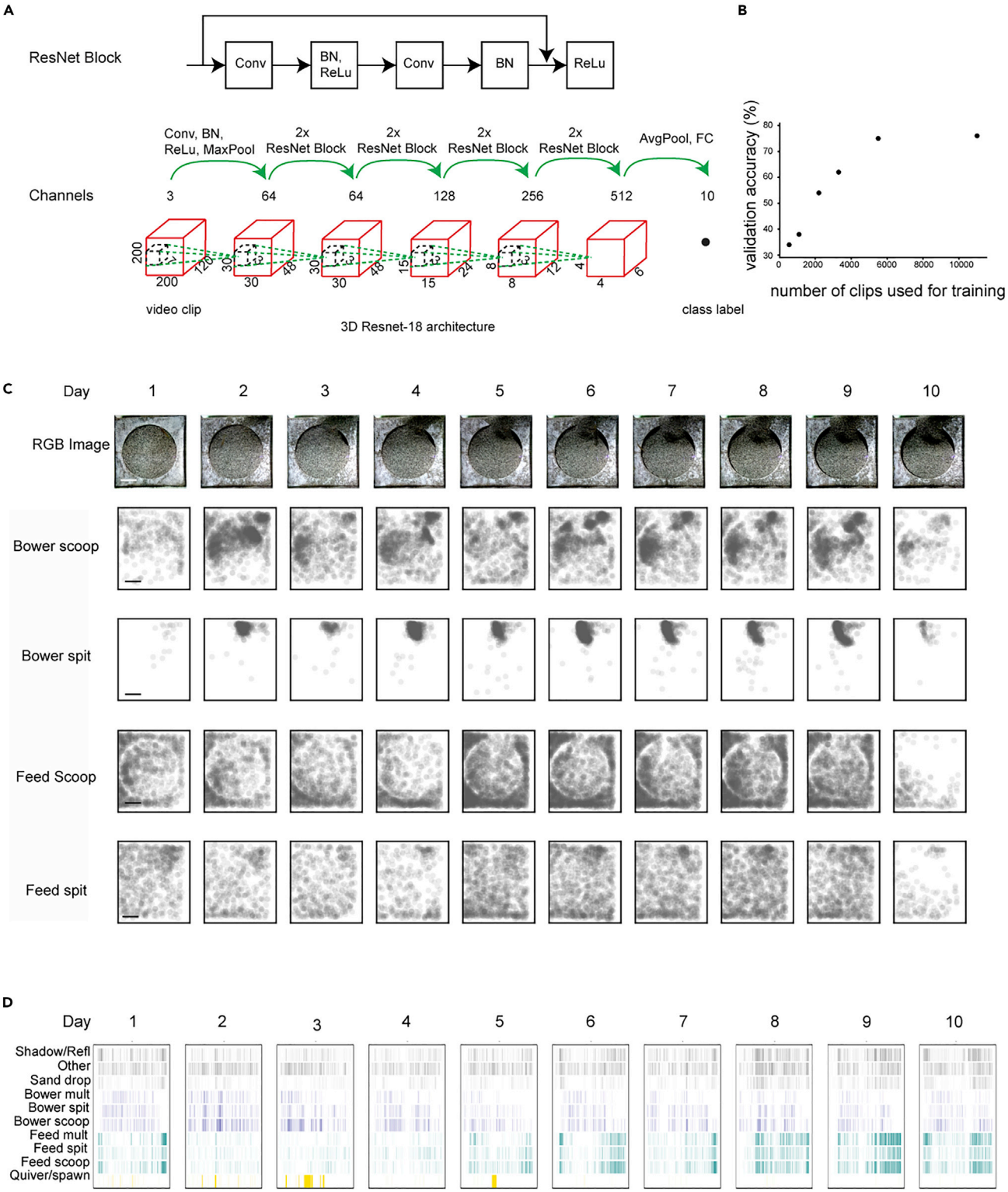


Figure 5. Automated Classification of Sand Disturbance Events Using a 3D Residual Network

(A) Schematic of 3D residual network.

(B) Validation accuracy of machine-learning labels as a function of number of videos used to train the model.

Figure 5. *Continued*
(C) Spatial position of four categories of sand manipulation events over 10 days of building by a *Mchenga conophoros*. A castle structure was built in the top middle of the field of view. Scale bar, 10 cm.
(D) Raster plot of time of each sand disturbance event by classification. (C and D) Building events occur at different spatial and temporal positions than feeding events.
See also Figure S5.

and tested their ability to classify video clips from the remaining trial. In all seven cases, we saw a significant drop in accuracy, from 76% to between 48% and 62% depending on the trial (Figure 6). We could, however, recapture much of this accuracy loss by including a small subset of videos from the excluded trial (~100–400), suggesting that including a limited number of labeled videos from new trials could dramatically increase the accuracy. Interestingly, trials that had the largest imbalance in the frequency of different categories compared with the mean also showed the largest decrease in accuracy (Figure S6).

DISCUSSION

Automated classification of behavior is an important goal for many areas of basic, applied, and translational research. Advances in hardware have made collection of large amounts of video data cost effective, using small, battery-operated cameras (e.g., Go Pro), mobile phones, or small microcomputers (e.g., Raspberry Pi). The small size and inexpensive nature of these hardware systems makes it possible to collect large volumes of data from many individuals. For example, here we collected hundreds of hours of video from seven home tanks in standard aquatics housing facilities. Inexpensive cloud data storage systems additionally allowed for the transfer and archiving of this data. We used Dropbox to store all video data for these experiments, and our long-term goal is to collect data from hundreds of trials in many species.

In this paradigm, our recording system collects >300 gigabytes of video data for each behavioral trial. A major challenge is thus designing a pipeline to identify behaviors of interest through large amounts of background noise. Here we demonstrate one possible solution that involves first identifying environmental disturbances and then classifying the behavioral causes of those disturbances. This approach is rooted in two main advances: first, an action detection algorithm for identifying times and locations of sand disturbances, allowing small video clips containing behaviors of interest to be generated on a large scale; and second, custom-trained 3D Residual Networks to classify each clip into one of ten possible categories.

For action detection, we tailored an HMM to recognize permanent changes in pixel color that occur whenever fish alter the sand. The core approach involves identification of lasting changes in the sand, which manifest as permanent changes in pixel color within specific subfields of view. This approach may be useful for studying many other animal behaviors that are defined by manipulation of the environment, such as nest construction, burrow digging, or web weaving. Many animals also disturb the environment as they forage and feed; for example, they may disturb or dig through ground substrate or ingest physical components of the environment such as leaves, fruits, berries, or algae. Thus, this approach may facilitate measurement of many behaviors in more complex and naturalistic environments. One limitation to this approach is that individual pixels must be focused on a specific region of the background. In our paradigm, we accomplished this by fixing the camera, but slightly moving cameras could also be used if an accurate means of registration is available to align frames at different time points.

Following identification of sand disturbance events, we used action recognition to classify events into ethologically meaningful behavioral categories. Previous machine learning strategies have relied on positional tracking and/or pose estimation data to classify animal behaviors (Hong et al., 2015; Anderson and Perona, 2014; Robie et al., 2017). In contrast, our paradigm was poorly suited to these methods, due to a lack of conspicuous stereotypical joint movement from top-down video and an abundance of stereotypical interactions between subjects and their environment, which were critical for defining the behaviors of interest. We found that a 3D ResNet classified video clips of animal behavior into 10 categories more accurately than a newly trained human observer, demonstrating that these networks can effectively quantify many different behaviors of interest from large volumes of raw video data. This result suggests that 3D ResNets may be a powerful tool for measuring animal behavior in naturalistic settings, and may drastically increase the scale of experimental designs in systems that, historically, have been constrained by the amount of human observation time required to measure behaviors of interest.

Feeding and construction behaviors have different underlying goals and are critical for survival and reproduction in the wild, but their physical execution is often indistinguishable to inexperienced observers.

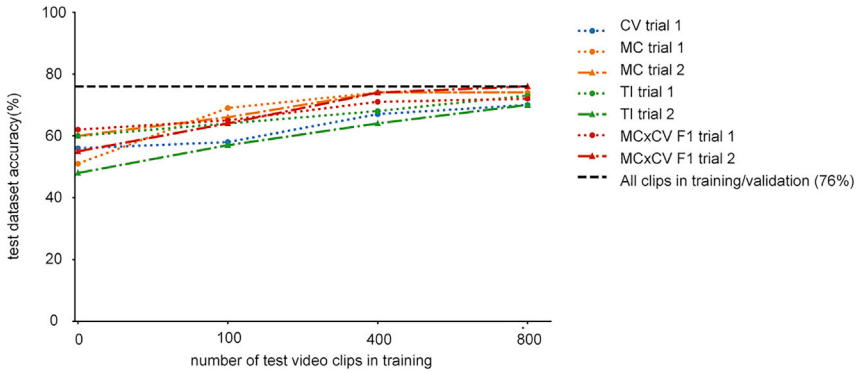


Figure 6. Application of the Model on New Trials

Accuracy of model on different trials using 0, 100, 400, or 800 training videos. We restricted training data for one of the seven trials and then tested the accuracy of the model on video clips specific to that trial. Models that used zero video clips for training showed a decrease in accuracy.

See also Figure S6.

Separating these behaviors is critically important in our paradigm, where there is a high risk that decisions to scoop and spit during bower construction will be undetectable through a high volume of feeding actions. Remarkably, the 3D ResNet was able to distinguish between feeding and construction behaviors more accurately than a newly trained human observer. The ability to accurately identify subtle differences in behavior may have important implications for better understanding the progression of neurological diseases characterized by subtle changes in locomotor function over extended periods of time, or monitoring efficacy of different treatment strategies for improving locomotor function.

Striking differences in the relative spatial positions and timing of predicted feeding versus construction behaviors were readily apparent across whole trials. For example, in castle-building MC males, bower spits were more spatially concentrated than bower scoops, feeding spits, and feeding scoops, consistent with the idea that castle construction is driven by scooping sand from dispersed locations and spitting into a concentrated region. Similarly, feeding behaviors and bower construction behaviors were expressed daily but at different times, whereas spawning events occurred infrequently in punctuated bursts. These data show that our system can be used to map tens of thousands of behavioral events in time and space, allowing future studies to unravel how these different complex behaviors are expressed in dynamic environments over extended time periods. It will be important to link these sand manipulation events with information about the actual structure being built by the fish. Development of an approach to characterize the shape of the bower at any given time will be necessary for connecting building behaviors with the structure.

The 3D ResNet accurately classified behaviors across three different species and one interspecies hybrid cross that all differ in morphology and color patterning. For example, *Copadichromis virginalis* males exhibit black body coloration, yellow heads and dorsal fins, and narrower jaws relative to *Tramitichromis intermedius* males, which exhibit yellow and red body coloration, blue heads, and a relatively wide jaw apparatus. Bower construction in particular has evolved in hundreds of species, and the evolution of feeding morphology and behavior is thought to be central to the explosive radiation of cichlids into thousands of species. Our results suggest that our system will likely be effective for measuring natural variation in these behaviors among hundreds of species. Lake Malawi cichlids can also be hybridized across species boundaries, enabling powerful genetic mapping approaches (e.g., quantitative trait loci mapping) to be applied in subsequent hybrid generations to identify genetic variants that influence complex traits. Our data show that our system is effective for phenotyping hybrid individuals, allowing future studies to identify specific regions of the genome that are responsible for pit versus castle building. Last, high prediction accuracy for quivering, a conserved and stereotyped sexual behavior expressed by many fish, supports that action recognition may be useful for analyzing mating behaviors in many fish species. More broadly, our results suggest that 3D ResNets may be effective tools for measuring complex behaviors in other systems even when individuals vary substantially in physical traits.

Limitations of the Study

It is important to note that although this report demonstrates that 3D Resnets should be useful for behavioral classification, it does not immediately generalize to other systems without additional work. For example, mouse behavior scientists could not immediately apply this to their behavior of choice.

Resource Availability

Lead Contact

Further information and requests should be directed to and will be fulfilled by the Lead Contact Patrick T. McGrath (patrick.mcgrath@biology.gatech.edu).

Materials Availability

This study did not generate new unique reagents.

Data and Code Availability

Original data for 14,000 annotations and video clips used to train and validate the 3D ResNet is available at Mendeley Data: https://doi.org/10.17632/3hspb73m79.1

All codes, including a ReadMe file explaining how to use them, for running action detection is available on GitHub at https://github.com/ptmcgrat/CichlidActionDetection. All codes for training the 3D ResNet for action classification is available on GitHub at https://github.com/ptmcgrat/CichlidActionRecognition.

METHODS

All methods can be found in the accompanying Transparent Methods supplemental file.

SUPPLEMENTAL INFORMATION

Supplemental Information can be found online at https://doi.org/10.1016/j.isci.2020.101591.

ACKNOWLEDGMENTS

We would like to thank Tucker Balch for suggestions on tank setup. We thank Andrew Gordus for helpful comments on writing the manuscript. This work was supported in part by NIH R01GM101095 to J.T.S., NIH R01GM114170 to P.T.M., NIH F32GM128346 to Z.V.J., and Georgia Tech Graduate Research Fellowships to J.L., V.A., and M.A.

AUTHOR CONTRIBUTIONS

Conceptualization, L.L., Z.V.J., and P.T.M.; Methodology, L.L., Z.V.J., and P.T.M.; Investigation, L.L., Z.V.J., J.L., T.L., V.A., and M.A.; Writing – Original Draft, L.L., Z.V.J., and P.T.M.; Writing – Review & Editing, L.L., Z.V.J., and P.T.M.; Funding Acquisition, J.T.S., P.T.M., and Z.V.J.

DECLARATION OF INTERESTS

The authors declare no competing interests.

Received: May 6, 2020
Revised: September 9, 2020
Accepted: September 16, 2020
Published: October 23, 2020

REFERENCES

Anderson, D.J., and Perona, P. (2014). Toward a science of computational ethology. Neuron *84*, 18–31.

Andriluka, M., Iqbal, U., Insafutdinov, E., Pishchulin, L., Milan, A., Gall, J., and Schiele, B. (2018). Posetrack: a benchmark for human pose estimation and tracking. Proc. IEEE Conf. Comput. Vis. Pattern Recognit. 5167–5176.

Benjamin, S.P., and Zschokke, S. (2000). A computerised method to observe spider web building behaviour in a semi-natural light environment. Eur. Arachnol. 117–122.

Berman, G.J., Choi, D.M., Bialek, W., and Shaevitz, J.W. (2014). Mapping the stereotyped behaviour of freely moving fruit flies. J. R. Soc. Interf. *11*, 20140672.

Brawand, D., Wagner, C.E., Li, Y.I., Malinsky, M., Keller, I., Fan, S., Simakov, O., Ng, A.Y., Lim, Z.W., and Bezault, E. (2014). The genomic substrate for adaptive radiation in African cichlid fish. Nature *513*, 375–381.

Collias, N.E., and Collias, E.C. (2014). Nest Building and Bird Behavior (Princeton University Press).

Dawkins, R. (1982). The Extended Phenotype (Oxford University Press).

Egnor, S.E., and Branson, K. (2016). Computational analysis of behavior. Annu. Rev. Neurosci. *39*, 217–236.

Ester, M., Kriegel, H.-P., Sander, J., and Xu, X. (1996). A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. Kdd, 226–231.

Feng, N.Y., Fergus, D.J., and Bass, A.H. (2015). Neural transcriptome reveals molecular mechanisms for temporal control of vocalization across multiple timescales. BMC Genomics *16*, 408.

Girshick, R. (2015). Fast r-cnn. Proc. IEEE Int. Conf. Comput. Vis. 1440–1448.

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

Gunel, S., Rhodin, H., Morales, D., Campagnolo, J., Ramdya, P., and Fua, P. (2019). DeepFly3D, a deep learning-based approach for 3D limb and appendage tracking in tethered, adult Drosophila. Elife *8*, e48571.

Hansell, M. (2000). Bird Nests and Construction Behaviour (Cambridge University Press).

Hara, K., Kataoka, H., and Satoh, Y. (2018). Can spatiotemporal 3d cnns retrace the history of 2d cnns and imagenet? Proc. IEEE Conf. Comput. Vis. Pattern Recognit. 6546–6555.

Hong, W., Kennedy, A., Burgos-Artizzu, X.P., Zelikowsky, M., Navonne, S.G., Perona, P., and Anderson, D.J. (2015). Automated measurement of mouse social behaviors using depth sensing, video tracking, and machine learning. Proc. Natl. Acad. Sci. U S A *112*, E5351–E5360.

Kain, J., Stokes, C., Gaudry, Q., Song, X., Foley, J., Wilson, R., and De Bivort, B. (2013). Leg-tracking and automated behavioural classification in Drosophila. Nat. Commun. *4*, 1910.

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

McKaye, K.R., Stauffer, J., Turner, G., Konings, A., and Sato, T. (2001). Fishes, as well as birds, build bowers. J. Aquaricult. Aquat. Sci. *9*, 121–128.

Mouritsen, H. (2018). Long-distance navigation and magnetoreception in migratory animals. Nature *558*, 50–59.

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

Pereira, T.D., Aldarondo, D.E., Willmore, L., Kislin, M., Wang, S.S., Murthy, M., and Shaevitz, J.W. (2019). Fast animal pose estimation using deep neural networks. Nat. Methods *16*, 117–125.

Petrou, G., and Webb, B. (2012). Detailed tracking of body and leg movements of a freely walking female cricket during phonotaxis. J. Neurosci. Methods *203*, 56–68.

Qiu, Z., Yao, T., and Mei, T. (2017). Learning spatio-temporal representation with pseudo-3d residual networks. Proc. IEEE Int. Conf. Comput. Vis. 5533–5541.

Redmon, J., Divvala, S., Girshick, R., and Farhadi, A. (2016). You only look once: Unified, real-time object detection. Proc. IEEE Conf. Comput. Vis. Pattern Recognit. 779–788.

Ren, S., He, K., Girshick, R., and Sun, J. (2015). Faster r-cnn: towards real-time object detection with region proposal networks. Adv. Neural Inf. Process. Syst. 91–99.

Robie, A.A., Seagraves, K.M., Egnor, S.R., and Branson, K. (2017). Machine vision methods for analyzing social interactions. J. Exp. Biol. *220*, 25–34.

Russell, A.L., Morrison, S.J., Moschonas, E.H., and papaj, D.R. (2017). Patterns of pollen and nectar foraging specialization by bumblebees over multiple timescales using RFID. Sci. Rep. *7*, 42448.

Stephens, G.J., Johnson-Kerner, B., Bialek, W., and Ryu, W.S. (2008). Dimensionality and dynamics in the behavior of C. elegans. PLoS Comput. Biol. *4*, e1000028.

Tucker, R. (1981). The digging behavior and skin differentiations in Heterocephalus glaber. J. Morphol. *168*, 51–71.

Vollrath, F. (1992). Analysis and interpretation of orb spider exploration and web-building behavior. Adv. Study Behav. *21*, 147–199.

Weissbrod, A., Shapiro, A., Vasserman, G., Edry, L., Dayan, M., Yitzhaky, A., Hertzberg, L., Feinerman, O., and Kimchi, T. (2013). Automated long-term tracking and social behavioural phenotyping of animal colonies within a semi-natural environment. Nat. Commun. *4*, 2018.

Wild, B., Sixt, L., and Landgraf, T. (2018). Automatic localization and decoding of honeybee markers using deep convolutional neural networks. arXiv, arXiv:1802.04557.

York, R.A., Patil, C., Hulsey, C.D., Anoruo, O., Streelman, J.T., and Fernald, R.D. (2015). Evolution of bower building in Lake Malawi cichlid fish: phylogeny, morphology, and behavior. Front. Ecol. Evol. *3*, 18.

iScience, Volume *23*

Supplemental Information

Automatic Classification of Cichlid Behaviors Using 3D Convolutional Residual Networks

Lijiang Long, Zachary V. Johnson, Junyu Li, Tucker J. Lancaster, Vineeth Aljapur, Jeffrey T. Streelman, and Patrick T. McGrath

Supplemental Figures

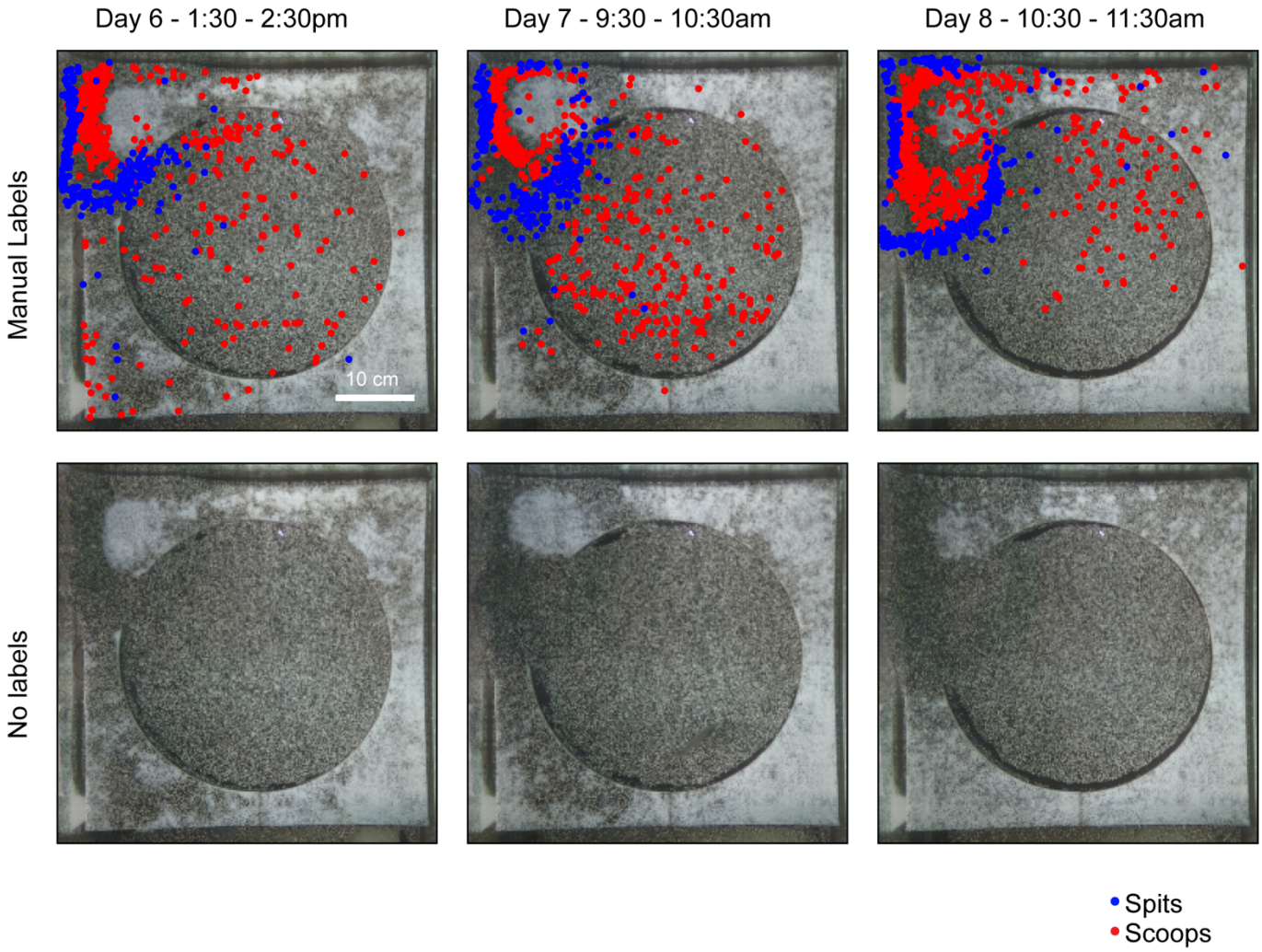


Figure S1. Example images on three different days during bower building by a Mchenga conophoros male, related to Figure 1. Top panel shows a top down view with all building scoops (red) and building spits (blue) during a one hour period identified by manual inspection. Bottom panel shows same images without scoops and spit events. A castle bower is being built on the middle left portion of the tray. Scale bar is 10cm.

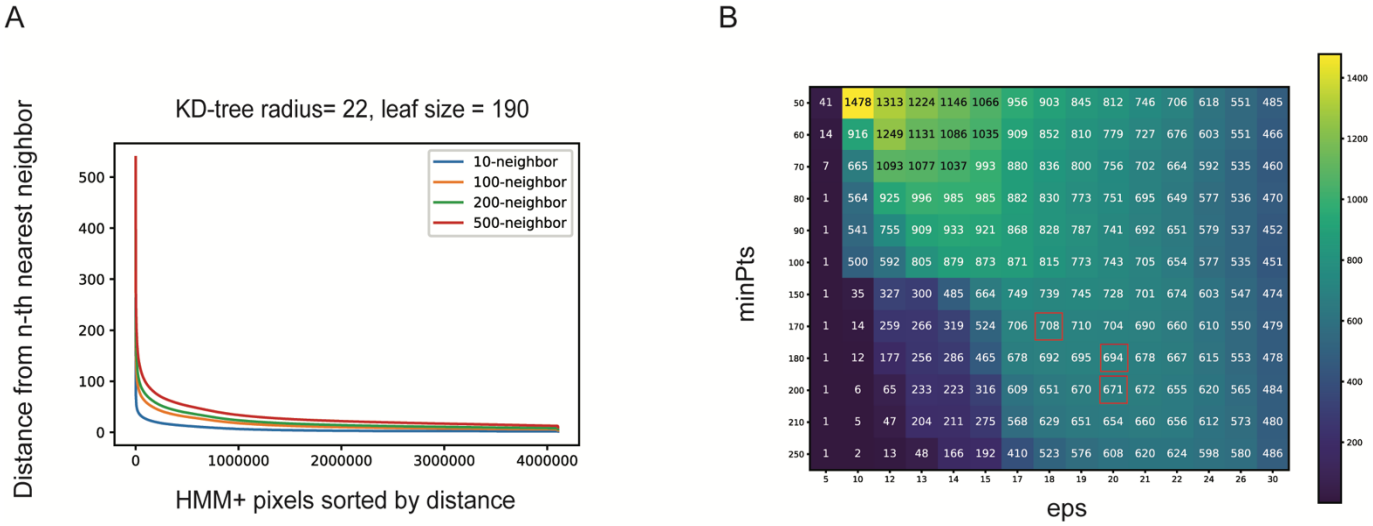


Figure S2. HMM pixel distance exploratory analysis and parameter search for DBSCAN, related to Figure 3. A. HMM+ pixels sorted by distance from n-th nearest neighbor. This plot was used to visualize the distribution of n-th nearest neighbor distances across HMM+ pixels. The knee point of the k-dist graph was used to estimate the optimal values for parameter eps to be 20-30. B. Number of identified clusters under different values for minPts and eps. This plot shows the number of identified clusters from segment of video data using different values for minPts and eps. Red boxes indicate values at which trained observers reviewed video clips of sand change clusters to identify optimal values for minPts and eps. See supplemental methods for further details.

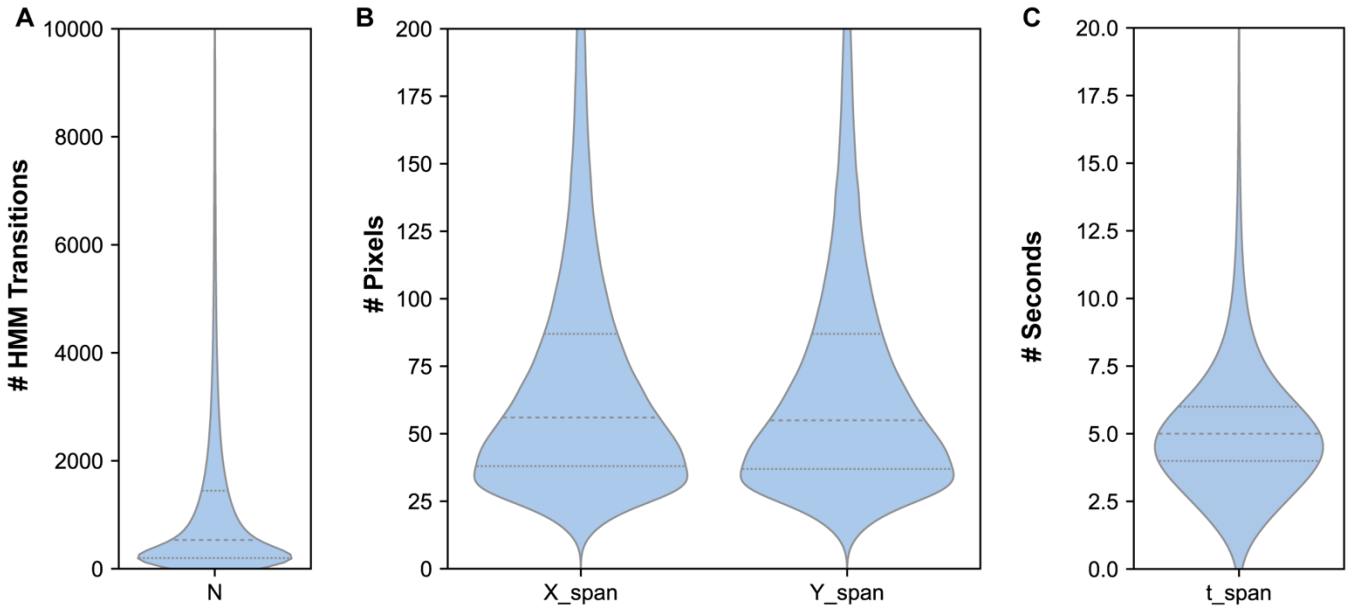


Figure S3. Violin plots showing distribution of cluster features, related to Figure 3. Dashed lines within the violins indicate the 1st quartile, median, and 3rd quartile values. A. Distribution of cluster sizes, using number of HMM transitions assigned to each cluster. B. Distribution of width and heights of each cluster. C. Distribution of time-length of each cluster. Since HMM transitions were only calculated at one second intervals, the number time-length must be an integer number.

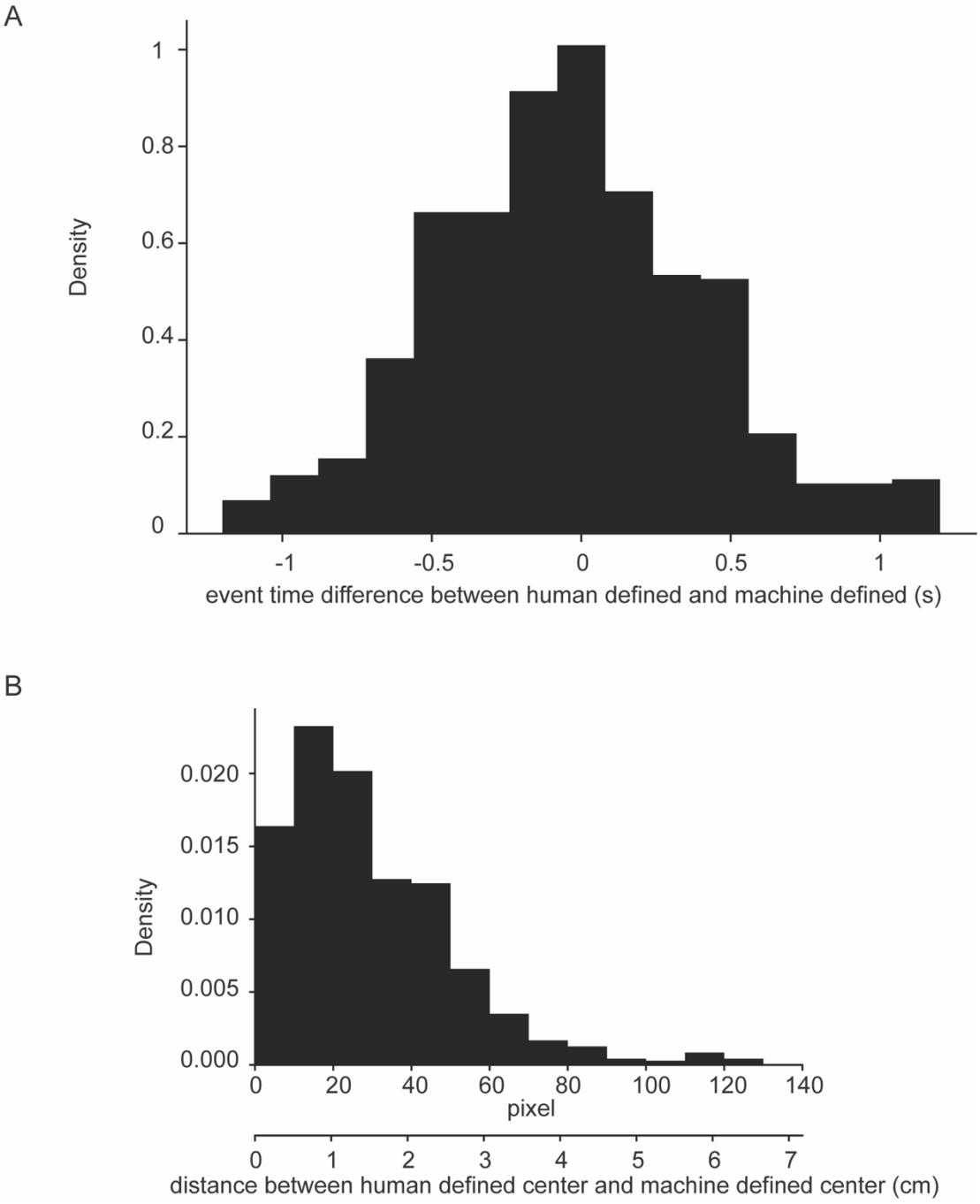


Figure S4. Spatial and temporal difference between human defined center and machine defined center, related to Figure 3. A. The distribution of time differences between human defined center and machine defined center. B. The distribution of distance between human defined center and machine defined center.

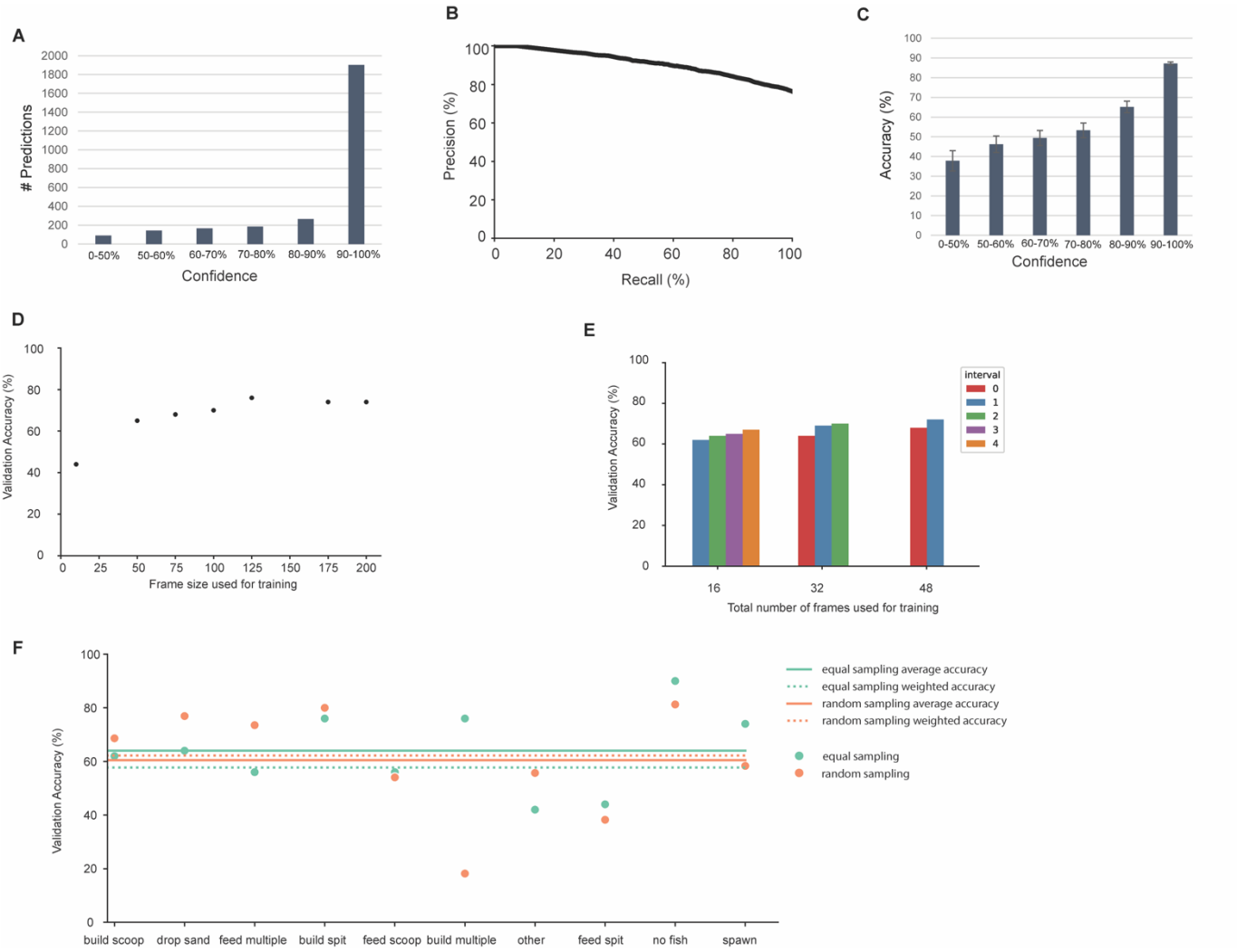


Figure S5. Ablation studies and performance analysis for the 3D ResNet, related to Figure 5. A. Number of predictions in validation set by confidence. B. Precision Recall Curve for the action classifier. C Accuracy (mean ± standard error) of predictions grouped by confidence scores. D. Model validation accuracy by frame size used for training. E. Model validation accuracy by total number of frames used for training. F. Per category accuracy by data sampling method.

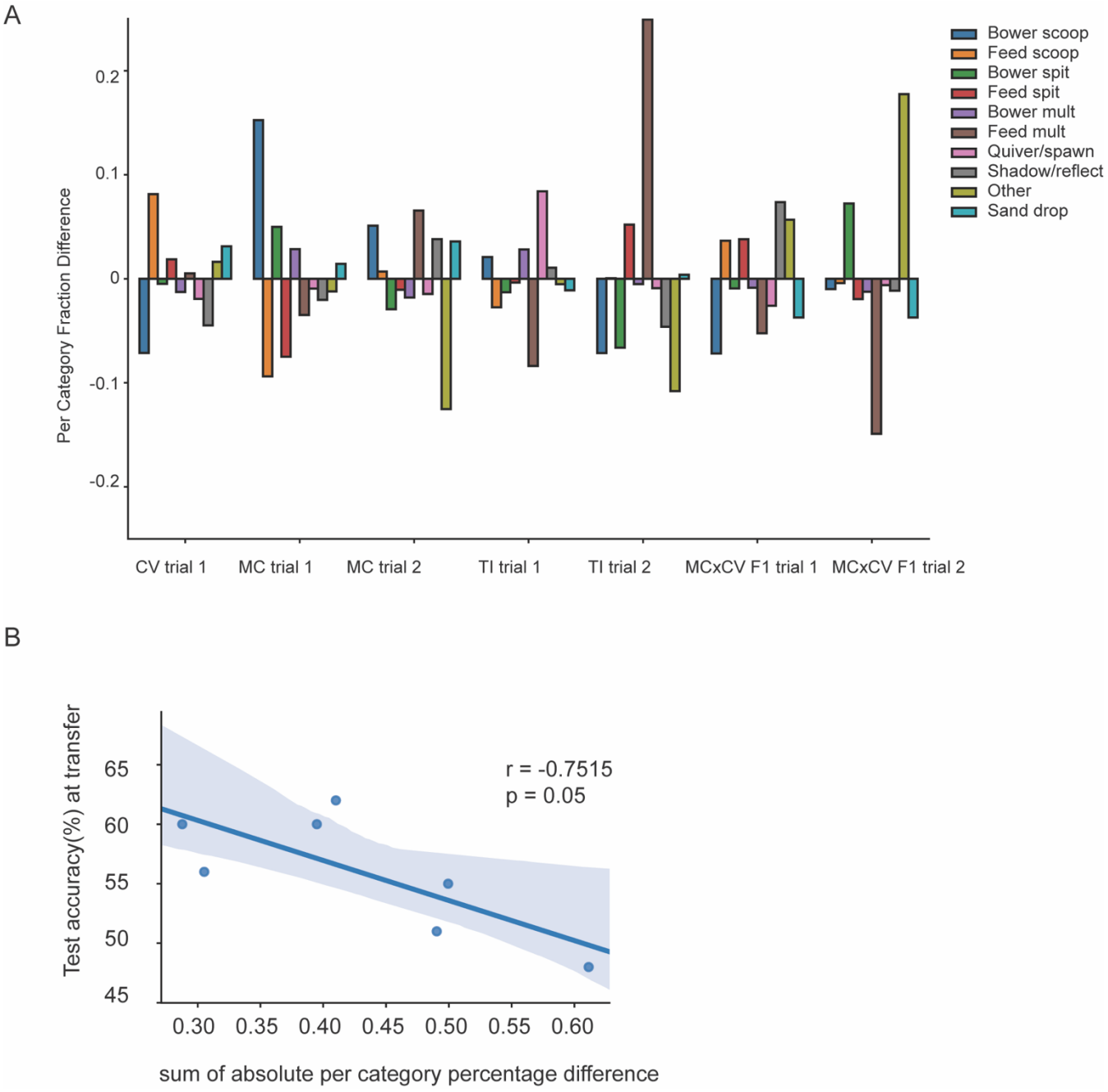


Figure S6. Relationship between category distribution difference and the 3D ResNet performance, related to Figure 6. A. Category distribution difference for each trial versus distribution of all other trials. B. Test trial accuracy versus its distribution difference from the training dataset. For each trial, the sum of absolute per category difference (from panel A) is used as x-axis and its accuracy when used as test dataset while the rest 6 trials as training dataset is used as the y-axis. A 90% confidence interval is shown as shaded blue.

Supplemental Tables

		Newly-trained label												
	Category	Build scoop	Feed scoop	Build spit	Feed spit	Build mult	Feed mult	Spawn	Shadow Reflect	Other	Sand drop	Total	Percent	Acc
Well-trained label	Build scoop	**277**	63	88	14	3	4	0	1	15	0	465	15.3%	59.6%
	Feed scoop	58	**356**	5	20	1	7	0	2	8	2	459	15.1%	77.6%
	Build spit	3	0	**196**	16	3	4	0	3	4	1	230	7.6%	85.2%
	Feed spit	3	4	64	**209**	0	6	1	0	2	26	315	10.4%	66.3%
	Build mult	4	0	10	3	**12**	4	0	1	6	0	40	1.3%	30.0%
	Feed mult	14	43	8	34	36	**498**	0	2	7	2	644	21.2%	77.3%
	Spawn	0	0	0	0	0	0	**66**	1	10	0	77	2.5%	85.7%
	Shadow/Reflect	0	1	1	3	0	0	0	**219**	6	1	231	7.6%	94.8%
	Other	19	15	5	11	2	5	3	13	**273**	2	348	11.5%	78.4%
	Sand drop	0	0	67	10	0	1	1	0	11	**140**	230	7.6%	60.9%
	Total	378	482	444	320	57	529	71	242	342	174	3039	100.0%	73.9%

Table S1 Confusion matrix for sand disturbance events classified by a newly-trained human observer, related to Figure 5.

Transparent Methods

1. Animals and husbandry

1.1 Subjects

Lake Malawi bower-building species (*Copadichromis virginalis, Tramitichromis intermedius, Mchenga conophoros*) derived from wild-caught stock populations, as well as genetically hybrid individuals derived from two of these species (described below), were housed in social communities (20-30 individuals) in 190 liter glass aquaria (90.2 cm long x 44.8 cm wide x 41.9 cm tall) into adulthood (>180 days). Aquaria were maintained under conditions reflective of the Lake Malawi environment: pH=8.2, 26.7ºC water, and a 12 h:12 h light:dark cycle with 60-minute transitional dim light periods. Fish were fed twice daily with dried spirulina flakes (Pentair Aquatic Eco-Systems).

1.2 In vitro hybridization

Reproductively active males and females were visually identified based on abdominal distension (females), nuptial coloration (males), and expression of classic courtship behaviors (e.g. chasing/leading and quivering). A hybrid cross was created between *Mchenga conophoros* (female) and *Copadichromis virginalis* (male). To cross-fertilize, a petri dish was filled with water from the home tank, and eggs were collected into the dish by applying gentle pressure between the pectoral region and the anal pore of the female. Eggs remained fully submerged while the male's sperm was extracted into the same dish by applying gentle pressure to both sides of the abdomen. The mixture was immediately and gently agitated and then eggs were gently rinsed twice with fresh aquarium water to reduce polyspermy. Eggs were transferred into a beaker containing a fresh oxygen tube, fresh aquarium water, and a drop of methylene blue to minimize risk of fungal infection. Water replacement was performed at least once daily until hatching (approximately 5-6 days post-fertilization).

2. Behavioral trials

Bower building occurs for multiple days. Animal care guidelines required that testing over such extended time periods had to be done in the home tank (as opposed to external testing arenas). In our facilities, home tanks are supported on tank racks with built-in piping and support beams that partially occlude top-down fields of view (FOVs). Additionally, all tanks have a central support crossbeam that partially occludes top-down FOVs. We found that a ~36 cm diameter sand tray placed on one half of the home tank provided a sufficient volume of sand for males to construct bowers, and was small enough to fit into an unobstructed top-down FOV. We designed a custom acrylic platform to surround the sand tray to prevent subjects from spitting sand over the edge of the tray onto the bottom of the aquarium. Thus, in this design, subject males and females could freely enter and exit the sand tray region throughout the trial.

For all behavioral experiments, a single reproductive adult male and four reproductive adult stimulus females of the same species or hybrid background were introduced into designated home tanks equipped with additional LED strip lighting (10 h:14 h light:dark cycle synced with full lights on), and a custom-designed hollow acrylic case (43.1 cm long x 43.1 cm wide x 10.2 cm tall, with a 35.6 cm diameter circular opening) surrounding a circular plastic tray (35.6 cm diameter x 6.4 cm deep, and elevated 3.8 cm above the aquarium bottom) filled with sand (Carib Sea; ACS00222). Sand trays were positioned approximately 30 cm directly below a custom-designed transparent acrylic tank cover (38.1 cm long x 38.1 cm wide x 4.4 cm tall) that contacted the water surface to eliminate rippling for top-down video recordings. Subjects and stimulus females were allowed to freely interact throughout the entirety of the recording trial.

3. Recording and monitoring hardware

We used a Raspberry Pi 3 Model B (RASPBERRYPI3-MODB-1GB; Raspberry Pi Foundation) connected a Raspberry Pi camera v2 (RPI 8MP CAMERA BOARD; Raspberry Pi Foundation) and a 1 TB external hard drive (WDBUZG0010BBK-WESN; Western Digital) to collect video clips for each trial. Data was stored locally on the hard drive until the end of the trial and then transferred to Dropbox through an Ethernet connection for analysis. The Raspberry Picamera was placed approximately 58 cm above the sand tray.

4. Data collection and analysis software

4.1 Video Collection

Upon start of a trial, an automated recording protocol was initiated collecting RGB video data during full light hours (08:00 to 18:00 EST) for 7-10 days. h264-encoded videos were collected at a 30 frame per second frame rate and a resolution of 1296x972 using custom Python scripts that used the picamera package (https://picamera.readthedocs.io). h264 videos were encoded into mp4 videos using ffmpeg (https://www.ffmpeg.org/). Data was transferred to a laboratory Dropbox account using rclone (https://rclone.org/).

4.2 Identification of HMM state changes

All code for running action detection is available on github at https://github.com/ptmcgrat/CichlidActionDetection. This repository included code that accomplishes the following: (i) uses a Hidden Markov Model (HMM) algorithm to detect changes in pixel values through time by sampling one frame per second, (ii) uses a density-based clustering algorithm to identify clusters of HMM+ pixels, or putative sand change events, and (iii) creates video clips and frames for manual labeling and machine learning classification.

To calculate HMM-predicted pixel values, we used the opencv and numpy packages for Python to decompress color mp4 videos into gray scale numpy arrays (0-255) to access data for each pixel across the entire video. To filter out short-term changes caused by fish, we calculated two rolling mean values for each pixel across a 120 second window either before and after each time point. Pixel values that were 7.5 units above or below either rolling mean value were removed and then interpolated using the 'numpy' package. Enduring changes in pixel values were identified using the 'hmmlearn' package for Python. Initial testing indicated that the time to calculate the HMM for the entire video would be on the order of days. Since the time to calculate the HMM states is on the order of $O(N^2)$, where N is the number of hidden states, we performed two additional pre-processing states to reduce the number of hidden states for each pixel. First, we only considered mean state values (i.e 0,2,..252,254). Second, since each pixel did not explore the entire range of possible values, we also calculated the values for each pixel that were found 10 or more times in the entire video. By using these heuristics, we could reduce the number of states to ~10-20.

We also found that changes in lighting over the course of the day created small changes in mean pixel value that resulted in HMM state transitions. To limit the number of these small HMM changes, we also prevented transitions less than 4 units by modifying the transition probability matrix.

4.3 Clustering of HMM state changes together into sand-manipulation events

In order to group HMM state changes together that were caused by the same fish-mediated event, we used density-based spatial clustering of applications with noise (DBSCAN) within the Python package 'scikit-learn' to identify clusters of HMM change in the presence of noise. DBSCAN analyzes the region surrounding each HMM pixel in time and space, determines if the neighboring region contains a minimal number of HMM+ pixels using a KD-tree, expands on dense groups of points, and repeats. DBSCAN parameters were set based on the observed size of sand change events and based on a k-dist graph. This enabled us to identify spatiotemporal clusters of HMM+ pixels, representing putative sand change events. Detailed discussion of different aspects of clustering follows.

Pre-processing

Occasionally there were large changes in lighting that resulted in state changes for the majority of pixels. We filtered out HMM+ changes from times when 1% or more of the pixels changed.

Parameters for density-based clustering

DBSCAN minPts and eps:

minPts: observers reviewed several hundred putative sand perturbance events and estimated the minimum size of a true sand change cluster to be 10 pixels x 10 pixels x 3 frames, and HMM+ pixels change to cover at least 15% of the putative sand change region. Based on these estimates we calculated the range for the minimum number of pixels in a sand change event to be between 50-250 pixels. For this paper we used 90 points for the minPts parameter.

eps: For a given k we defined a function k-dist from the database D into the non-negative real numbers, mapping each point to the distance from its k-th nearest neighbor. After sorting the points in the database in descending order based on their k-dist values, the graph of this function suggested a density distribution in the database. This graph is called the sorted k-dist graph, as described previously (Ester et al., 1996). We then fit a nearest neighbor tree to all points and used the k neighbors query to find the $minPts^{th}$ nearest neighbor for each point, and the k-dist graphs for minPts = 200. We found that most of the points were close to each other; and most points had at least 200 points within 40 units.

We used the knee point of the first k-dist graph (at minPts = 200; Figure S2) to estimate the optimal values for eps to be 20-30. We then ran DBSCAN on a grid of parameters and quantified the number of clusters labeled under each set of parameters. Three observers then annotated three sets of clips corresponding to minPts and eps values (Figure S2). After comprehensive review, we found the eps = 18 and minPts = 90 to best reflect true sand change clusters.

Nearest Neighbor KD-tree treeR/neighborR and leaf size:

treeR and neighborR are equivalent parameters for constructing KD-trees (Pedregosa et al., 2011). Within a radius around each point, all distances between this point and other points are calculated. DBSCAN queries the distances within eps (eps=18 in our analysis) for each point, so the treeR/neighborR ≥ eps. We set this parameter to 22 to prepare the distance matrix for DBSCAN with eps <= 22.

leaf_size: this parameter is a threshold below which the calculation switches from traversing tree to brute-force. For small data sets (N less than 30 or so), brute force algorithms can be more efficient than a tree-based approach. Changing leaf_size will not affect the results of a query, but can significantly impact the speed of a query and the memory required to store the constructed tree as seen in (Pedregosa et al., 2011) and here: https://jakevdp.github.io/blog/2013/04/29/benchmarking-nearest-neighbor-searches-in-python/#Scaling-with-Leaf-Size. We set leaf_size above minPts 90 (leaf_size=190).

Timescale:

Since DBSCAN uses one radius to search clusters in all dimensions, we scaled the time dimension so that the temporal lengths of events were similar in magnitude to their spatial width, such that events were, in general, roughly spherical in 3D. By manually reviewing hundreds of events, we determined that the duration of sand change events was < 5 seconds, and the spatial widths were typically < 50 pixels. Based on this, the time dimension (on frame/second) was scaled by 10x.

4.4 Creation of video clips for each cluster

Finally, to create video clips for machine learning, we used the “opencv” package for Python to create small video clips around the center of each cluster. The width, height and length of these videos were 200 pixels, 200 pixels, and 120 frames (4 seconds). For manual labeling, we also included cluster information on the pixels that underwent HMM transitions during the 4 second time window.

5. Machine Learning

5.1 Behavioral definitions for manual annotation

The following 10 categories were used to categorize each of the manually-labeled video clips.

Bower scoop: subject male collects sand into its mouth during bower construction.

Bower spit: subject male expels sand from its mouth during bower construction.

Bower multiple: multiple bower scoops and/or spits are expressed by the subject male within the same video clip.

Feeding scoop: fish collects sand into its mouth during feeding.

Feeding spit: fish expels sand from its mouth during feeding.

Feeding multiple: multiple feeding scoops and/or spits are expressed by a fish within the same video clip.

Spawn/quiver: the subject male rapidly vibrates his body left to right while simultaneously circling, often but not necessarily with a female in frame. The male's body is typically arched left to right, with his anal fin (egg spots) displayed directly in front of the female. The female may also be present and circling in immediate proximity with the male.

Sand dropping: A fish expels or releases sand from the mouth either while high in the water (after which the sand sprinkles down through the water before settling), or release of sand upon initiation of a rapid burst of swimming (typically chasing or being chased). A rarer subset of sand dropping events includes filtering sand through the operculum while swimming, typically during feeding.

Other: Changes to the sand caused by any other fish activity not described above, often as a result of swiping of the fin or rubbing of the ventral surface of the body along the sand during performance of other behaviors. More rare cases included instances in which two fish both perform behaviors in the same clip but the sand change was designated as a single cluster.

Shadow/reflection

Other changes that are not caused by fish manipulating or changing sand, most commonly reflections of activity in the aquarium glass and shadows cast by a stationary or very slow-moving fish, or in rare instances food, feces, or other debris settling on the sand surface.

All labeled data can be found at:

https://data.mendeley.com/datasets/3hspb73m79/draft?a=b72c1f6d-505a-431a-ba3d-824cd148c01e

5.2 Deep learning of cichlid behaviors

All code for running behavior classification of available on github at https://github.com/ptmcgrat/CichlidActionRecognition. A trained observer manually classified 14,172 video clips randomly selected from representative days across seven trials, spanning seven subjects, three species, and one hybrid cross. Each clip was classified into one of the ten categories listed above. We randomly selected 80% of manually annotated clips for training an 18 convolutional layer 3D ResNet, and the remaining 20% of clips were used for validation. Briefly, 3D ResNets are 3D convolutional neural networks (CNNs) that incorporate features of Residual Networks (ResNets), in which signals are bypassed across convolutional layers during training. This approach allows 3D ResNets to be deeper and more accurate than traditional 3D CNNs for action classification tasks. For training, validation, and prediction we used a 18- layer Resnet3D model as previously described (Qiu et al., 2017). The architecture, including the shape of each layer, of the neural network is shown in Figure 5A. Each ResNet Block consists of 2 convolutional layers, each has a kernel size of 3x3x3. The first convolutional layer has a kernel size of 7x7x7. ReLu is used as the activation function across the neural network. Prior to training, each video clip was randomly cropped at a 120x120 frame. Each training video clip was also randomly cropped temporally down to 96 continuous frames. Random horizontal and vertical flip, at a rate of 0.5, was also used for data augmentation. Finally, each channel was then normalized based on the mean value for that channel. Validation and test video clips were always cropped in the center spatially and temporally. For training, stochastic gradient descent was used

to optimize the parameters of the neural network. Specifically, the learning rate was set to 0.1 (and set to decrease after 10 consecutive epochs of no change in validation loss), momentum was set to 0.9, dampening was set to 0.9, and weight decay was set to 1.0×10^{-5}. The network was only initialized at the start and was trained for 100 epochs with a batch size of 8 per epoch. The final accuracy is calculated as the average the last five epochs when the accuracy reached plateau by visual inspection.

5.3 Testing model generalizability

In order to test the generalizability of the learned model, we used 6 out of the 7 projects for training/validation and the other one for testing. We tested all 7 combinations of these 7 projects. Sample split, network architecture and data augmentation were the same as above and test accuracy is calculated as the average the last five epochs when the test accuracy reached plateau. In order to test if label some of the test video clips could help the model to generalize to the new dataset, we randomly put 100,400,800 randomly selected test clips in training and the remaining test clips in test. After this, the new training/validation dataset went through the data augmentation and network computation. The new test accuracy is calculated on the remaining test video clips.

To figure out why accuracy decrease in test trial, we compared the distribution of the categories for each test trial and that of the training trials. For each trial, we first calculated the percentage for each category. This distribution is subtracted by that of the training trials to get the distribution difference. Finally, the accuracy when this trial is used as test dataset is regressed on the sum of absolute per category difference. The p value is calculated from Pearson's Correlation Coefficient.

Supplemental References

PEDREGOSA, F., VAROQUAUX, G., GRAMFORT, A., MICHEL, V., THIRION, B., GRISEL, O., BLONDEL, M., PRETTENHOFER, P., WEISS, R. & DUBOURG, V. 2011. Scikit-learn: Machine learning in Python. *Journal of machine learning research,* 12**,** 2825-2830.