Supplementary material

In this supplementary information, we provide more in-depth and technical details of our work, please see our code for even more details. We start with discussing the camera setup used and display a selection of samples from our dataset of larval fish behavior, including videos (section S1). We move on to describe the models we used for this study and provide details on the training and evaluation procedures used (sections S3, S4). Finally, we detail the bootstrap procedure used to estimate the confidence of our strike rates (section S5.2) and the statistical models used to estimate the influence of environmental parameters on these rates (section S5). We test the robustness of our pipeline by performing statistical analysis using only samples detected by our best-performing classifier and comparing the results to the analysis with all strike events (section S5.4).

S1. Data acquisition

Here, we provide additional detail regarding the deployment of the submersible filming setup in the tanks, and detail the two variations on the setup used.

The camera system was placed in the tank at the beginning of each filming day, and removed at the end of it. The submersible flashlight was rigged to the setup with a flexible detachable arm. As a result, the position of the flashlight varied between deployments; creating variable lighting conditions even between videos taken in the same tank on consecutive days using the same setup. Light intensity provided by the flashlight was approximately equivalent to that of sunlight at 5m depth in clear coastal water (Sommerfeld and Holzman, 2019).

Two distinct camera setups were tested. The change in setup stemmed from an attempt to improve the sharpness of the videos. Initially, the camera was placed in a snug housing, leaving ∼0.6 m of water between the lens and the focal plane. Because the water in the rearing tanks is turbid, the resulting images were blurry, yet usable. We attempted to improve the setup by placing the camera in a longer housing, such that most of the optical path was through air (inside the housing), rather than through water. This resulted in much sharper images with larger areas appearing in focus, but the edges of the housing were also sometimes included in the frame. It is important to note that even though both setups had their shortcomings, feeding events were still detectable by our analysts, and the videos were successfully annotated.

Unlike the previously published dataset (Shamur et al., 2016), which was acquired in the laboratory under constant conditions (see section SS2 below) variable filming conditions make our videos appear different from one another. In addition to the visual differences caused by the two camera setups, videos varied considerably within each setup. This is because our filming documented the variable conditions within the rearing tank, as determined by the hatchery's rearing protocols. Specifically, the age of the larvae, the number of larvae in the tank, the type and amount of food, water turbidity, illumination, O_2, currents, and turbulence, all varied between filming days; affecting both larval behavior (e.g. feeding rate) and the visual appearance of the videos (see Fig. SS1). Additionally, as we are filming tiny creatures mid-water, there was no typical background for a particular tank or cohort, unlike camera traps positioned in the same location over time.

We bring examples of larval fish behavior in Video S1 (typical swimming behavior) and Video S2 (feeding strike behavior).

S1.1. Illumination system

In order to film in the pools backlight illumination was essential, particularly given for filming in high frame rates and under such optical constraints. Our flashlight provided light equivalent in strength to that of ambient daylight at 10m depth in an oligotrophic ocean (Sommerfeld and Holzman, 2019). As such, though the light stimulus introduced

a disturbance to the rearing pool environment, it is not different from what the animals would have encountered in the wild. We visually inspected the pool from above and did not detect behavioral changes in the larvae during the filming time. Furthermore, we did not detect changes in the number of larvae in the imaged volume throughout filming. We tested this by counting the number of larvae in 20 random frames along each sequence and running a mixed-effect model with the number of larvae as the dependent variable, time as the independent variable, and sequence ID as a random factor. The slope of the regression model was not significant ($P<0.5$, $R2<0.01$) indicating that aggregation or dispersion of larvae in response to the light source was unlikely. Additionally, the survivorship of larvae in the filmed tanks for we had survivorship data from the hatchery ($N=10$) was not different than the survivorship in the adjacent tanks in which we did not film (t-test, $P>0.1$). While all these are indirect evidence, they show that the effects of the filming system on feeding strike rates were probably low.

S2. Previous work on automated larval feeding detection

Shamur et al. (2016), attempted to classify the feeding behavior of fish larvae in a laboratory using classic vision methods. In their work, larvae were detected and classified using a pipeline of edge detection, action descriptors extractions, and SVM classifiers. Although they achieved reasonable results (AuROC = 0.82, ACC = 72.7±2.1), the analysis pipeline is not transferable to naturalistic settings, such as ours. Their pipeline requires manual tuning of thresholds for each new video analyzed to adjust for changes in lighting conditions and water clarity; the edge detection function works poorly for out-of-focus fish, which are common in our data; Image quality in the laboratory was superb compared to our *in-situ* filming because in the laboratory the water is cleaner and most of the optical path is traversed through air. For this reason, clips in the laboratory dataset were cropped around the mouth of the fish, not visualizing the whole body, unlike our dataset. In addition, to avoid occlusions and maintain fish in the focal field, the larvae were placed in a narrow arena (∼ 5mm), essentially creating a 2D scene. As a by-product, this confined space potentially limited the natural behavioral repertoire of the fish. For all these reasons, we do not assess our models on this dataset, nor run the old pipeline on our new data.

Note that our definition for "strike" is different than the one used by Shamur et al. (2016), who created an automated pipeline for the detection of larval fish feeding events in the laboratory. There, filming conditions allowed for a high-resolution visualization of the fish's mouth, and the action classifications were made on crops of the fish's head rather than the entire body. Shamur et al. (2016) used a narrow aquarium to keep the fish perpendicular to the camera and within the (narrow) focal plain; the camera and the light source were placed outside the aquarium such that their visual paths passed mostly through air (rather than the murky pool waters). Importantly, the Shamur et al. (2016) pipeline was not feasibly applicable to diverse datasets as it depended on the manual selection of certain parameters per video and did not generalize well to new videos. As is frequently the case under field conditions, the conditions in the pool forced a different configuration of the filming setup, in which the resolution and contrast were lower and fine-grained details (like food particles) were more difficult to discern. Despite these challenging conditions, our setup produced videos that clearly showed a diversity of natural behaviors in unprecedented quality and detail (see Fig S1).

S3. Classifier training and evaluation

Our pipeline consists of two models, trained separately: an action classifier and a fish detector. The action classifier was trained on a curated balanced dataset, and its performance was further evaluated on a larger dataset under more naturalistic conditions, emulating those encountered during the full application of the pipeline to videos. The following section describes these components in detail.

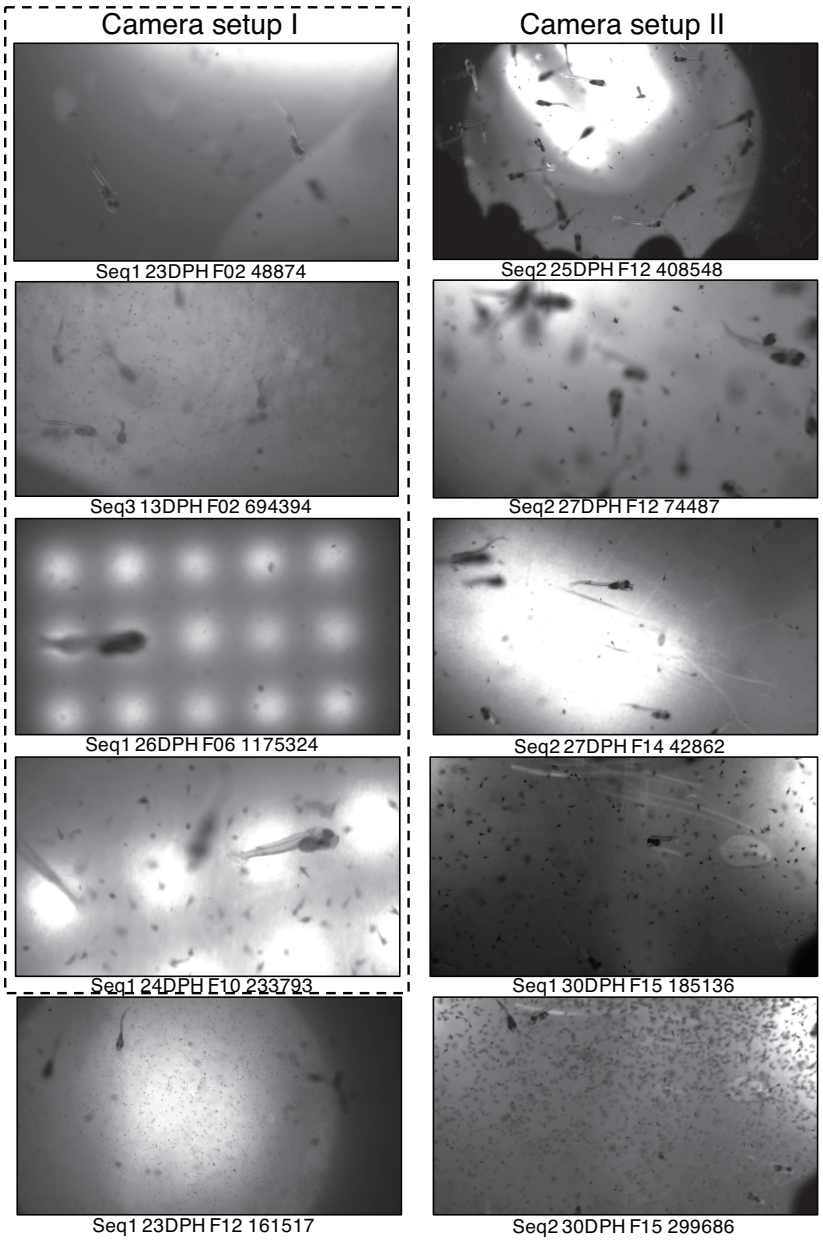

Figure S1: Sample frames. Examples of video frames from various videos from our dataset, from both filming setups, the first in the dashed box, the second outside it. The naming convention under each frame states the sequence number in that filming day (Seq); the age of the larvae (DPH, days post-hatching); the cohort identity (F, each cohort was filmed consecutively throughout its growth cycle), and finally the frame number in the sequence. Different combinations of both technical and environmental variables create a visually diverse dataset

S3.1. Datasets

From the full videos, we created two datasets of cropped clips to train and test our action classifiers.

Training dataset

To train our model we opted for a balanced training scheme, as we found experimentally that using a large number of "non-strike" clips and class re-weighting resulted in poor performance. Our balanced training set consisted of 66

Split	Swims	Strikes
Train	41	39
Validation	11	11
Test	19	16

Table S1
Action classifier dataset composition.

"strike" clips tightly cropped both temporally (10 frames before mouth opening, 5 frames after mouth closing) and spatially around a single focal fish (average±Sd duration= 45±18 frames; average±Sd height = 385±93 pixels). And 71 clips featuring non-strike behavior. To maximize the difference between the behaviors, we chose only samples containing routine forward swims. We extract clips of 200 frames and each clip was randomly cropped temporally to match the distribution of the "strike" class. Our balanced dataset was further partitioned to train, validation, and test datasets. We ensured that clips cropped from the same raw video were grouped in the same partition in order to avoid data leakage. Additionally, to avoid creating spurious features related to differences in filming setup between videos, we selected a similar proportion of clips from each filming setup in each of the two classes. The sample sizes of each partition are given in Table SS1

Test dataset

This dataset was generated with the help of our trained fish detector (see below). We used 11 videos from the dataset that featured the highest number of feeding strikes according to the manual annotation. Rather than going through the entire video, we applied the detector on randomly sampled frames. In addition, we strategically sampled frames in the temporal coordinates of our manually annotated feeding strikes. In total, we sampled 62 strike-related frames and 949 frames at random. We maintained a ratio of ∼1:15 between frames that contained events of interest and those that did not, to emulate the natural rarity of larval feeding strikes. For each frame sampled, we applied the detector as described in the main paper; spatiotemporally cropping clips around the detections. Temporally, a ±40 frame window was cropped around the detection frame. Spatially, small square clips were cropped around each detection, with the cropping size determined according to the typical size of the fish in each video (range: 250-650 pixels). There was a partial overlap between raw videos used in the balanced training dataset and those used in the test dataset. Five of the 11 videos were used in the test dataset and in the test set of the balanced dataset. Six out of the 11 videos used in this dataset were also used in the train or validation splits of the training dataset. While this overlap is not optimal, we note that the clips in each dataset were generated differently. Temporally, samples in the test dataset are not tightly cropped around the feeding event, and actually, clips are twice as long as an average feeding event; spatially, the clips were not tightly cropped around the fish of interest. Furthermore, four of these 6 videos featured additional strike events that were not included in the balanced dataset ($n = 13$ clips, range: $2 - 5$ per video). When evaluating our classifier on this dataset we report results for the entire corpus of data, as well as for only those clips from the five videos not included in the training split of the balanced dataset, which we term the naive set. The resulting dataset comprises 4,563 short clips (62 "strikes" and 4,501 "swims"); with the positive class accounting for ∼1.4% of the data. The entire dataset was annotated by a team of trained observers using custom labeling software. To prevent bias, all observers were blind to the results of the classifier. For each clip, observers labeled the main activity of the fish in the clip. Observers also noted parameters regarding the behavior of the fish and the photographic quality of each clip. We merged these behavioral and photographic labels into the following mutually exclusive categories: strikes, abrupt movements, non-routine swimming, compromised footage, routine swimming, and can't tell or no fish.

For more details on the creation of these two datasets see Bar et al. (Unpublished results).

S3.2. Models

We compared the performance of several popular backbones for the action classification of "strike" and "swim"/"non-strike" events: an I3D network (Carreira and Zisserman, 2017); a two-stream SlowFast Network (Feichtenhofer et al., 2019) with a 3D-ResNet-50 (Feichtenhofer et al., 2017) backbone; and just the Slow pathway of the SlowFast network.

In the SlowFast model, the rate at which each pathway samples frames from the input clip is a user-specified hyperparameter. We chose to follow one of the settings suggested in the original paper (Feichtenhofer et al., 2019), with the Slow pathway sampling eight frames uniformly throughout the clip, and the Fast pathway sampling 32 frames throughout the clip. To ensure that this sampling provided good coverage of the feeding strikes, the sampled clips were manually inspected. The ratio between the channels in each pathway, specified by the β parameter, was set to $\beta = 1/8$, as we used pre-trained weights (see below). Specifications of the rest of the training hyperparameters are provided in the paper's code.

S3.2.1. Variance Image

We integrated our knowledge of the biology and behavior of the fish to generate an additional channel of information in our clips. We exploited the fact that "strike" behavior is characterized by abrupt movements, while "swim" behavior is typically a smooth undulatory movement. These differences are expected to affect the rate at which pixels change their intensity values throughout the clip, with "strike" pixels showing areas of higher variance.

Rather than calculating the optical flow for each clip, which is computationally intensive, we calculated the variance image of the entire clip (see Fig. S2). The variance image was duplicated along the temporal axis and stored as a third channel, alongside two duplicate channels of the clip's monochrome sequence. In previous work, we tested the contribution of this manipulation with an ablation study (Bar et al., Unpublished results).

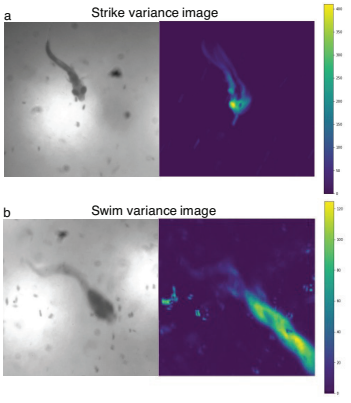

Figure S2: Example of variance images. The first frame in the raw clip (left) and variance image on the entire clip (right) for a "strike" clip (a) and a "swim" clip (b). Note the different movement patterns captured in each clip by the variance image, and note the different scales on the right.

S3.3. Evaluation results

Performance on the training dataset

As can be seen in Fig. S3 and Table S2, all pre-trained backbones classifiers did well on the training dataset, reaching high levels of saturation in both the ROC and PRC and correspondingly had high AuROCs and AuPRCs for all the dataset splits. However, clearly, most SlowFast-based backbones achieve better performance than I3D, with

Backbone	Train		Val		Test	
	AuROC	AuPRC	AuROC	AuPRC	AuROC	AuPRC
I3D	0.99 ± 0.014	0.98 ± 0.027	0.62 ± 0.147	0.61 ± 0.161	0.89 ± 0.036	0.81 ± 0.060
Slow	1 ± 0.003	1 ± 0.003	0.86 ± 0.093	0.88 ± 0.060	0.96 ± 0.028	0.94 ± 0.050
SlowFast - Kinetics	1 ± 0	1 ± 0	0.85 ± 0.043	0.88 ± 0.018	1 ± 0	1 ± 0
SlowFast - none	0.63 ± 0.008	0.61 ± 0.035	0.32 ± 0.0425	0.48 ± 0.017	0.39 ± 0.025	0.38 ± 0.007
SlowFast - SSv2	1 ± 0	1 ± 0	0.98 ± 0.017	0.98 ± 0.011	1 ± 0	1 ± 0

Table S2
Model performance on the training dataset. Numbers represent $meanAuC \pm std$ for each backbone, averaging results from training using three different random seeds, I3D and Slow are pre-trained on Kinetics, SlowFast backbones on Kinetics, SomethingSomethingV2 or with no pre-training.

Backbone	AuROC	AuPRC
I3D	0.67 ± 0.078	0.02 ± 0.007
Slow	0.82 ± 0.058	0.11 ± 0.037
SlowFast - None	0.45 ± 0.007	0.01 ± 0.0001
SlowFast - Kinetics	0.94 ± 0.001	0.61 ± 0.025
SlowFast - SSv2	$\mathbf{0.97 \pm 0.003}$	$\mathbf{0.66 \pm 0.015}$

Table S3
Model performance on the test dataset. Numbers represent $meanAuC \pm std$ for each backbone, averaging results from training using three different random seeds, I3D and Slow backbones were pre-trained on Kinetics. The pre-trained SlowFast variants are by far the best performing in both metrics.

the SSv2 pre-trained model showing the best results. The results of the Slow and SlowFast backbones pre-trained on Kinetics are comparable on this dataset, however, diverge when assessed on the test dataset (see below).

Performance on the test dataset

The test dataset was constructed in a way that captures much of the difficulties a classifier might encounter when deployed on full videos. This dataset poses challenges typical of naturalistic conditions, such as multiple occluding animals, extreme lighting conditions, motion blur, and severe class imbalance. Hence, achieving good performance on this dataset was critical.

Results of all backbones on the test dataset (n=4,563) are given in Table S3 and Fig. S4. As with the training dataset, the pre-trained SlowFast backbones showed superior performance with a mean AuROC of 0.94, 0.97 and mean AuPRC of 0.6, 0.66 for the Kinetics and SSv2 respectively. Note that the expected AuPRC for a random classifier under the observed class imbalance is 0.014. The I3D and Slow backbones performed considerably worse than on the balanced dataset. The model with no pre-training performed poorly, being in line with a random classifier. Results for the naive video set, excluding videos used for the train split of the classifier, were similar (Fig. S5 in SI). These results show, that despite the extremely low data regime, our training resulted in models that are capable of detecting larval fish feeding behavior under challenging conditions outside the laboratory.

S3.4. Results on the naive set

The naive set comprised clips from those raw videos that were not included in the train/validation splits of the balanced curated dataset (i.e., those videos that were not sources for clips the action classifier trained on). As can be

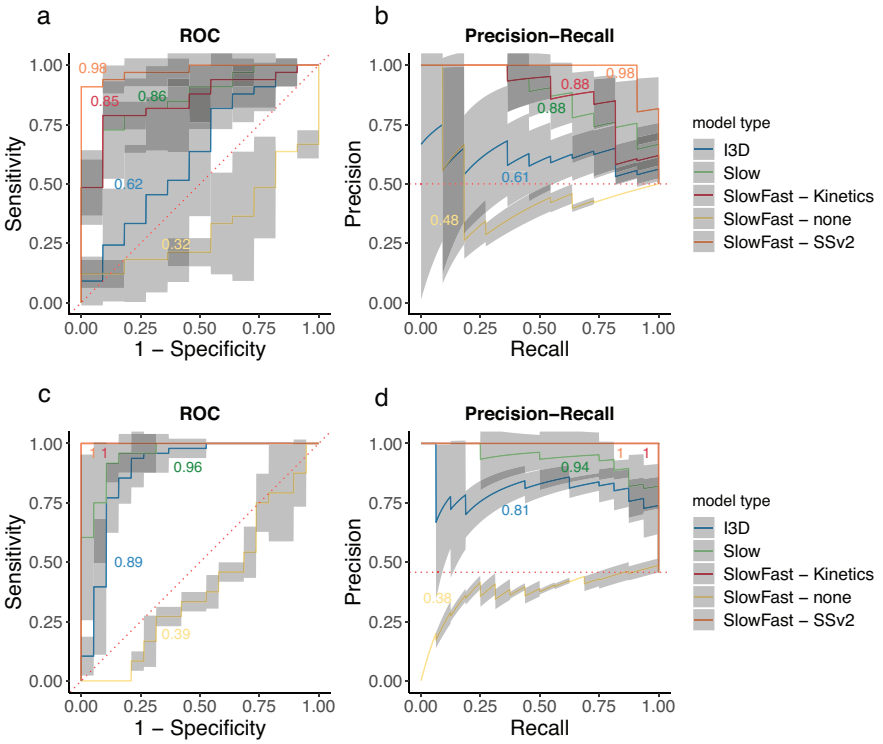

Figure S3: Comparison of action classifier backbones and pre-training methods on the balanced curated dataset. The mean ROC and mean PRC for all models averaged from training on three different random seeds. a,b) Results on the balanced validation set; c,d) Results for the balanced test set. Curves are presented for all backbones; I3D (blue), Slow (green), and SlowFast (red), all pre-trained on Kinetics; SlowFast with no pre-training (yellow) and SlowFast pre-trained on SSv2 (orange). The colored numbers are the area under each curve (AuROC/AuPRC). The shaded area around each curve represents the 0.95% confidence interval. The dashed red line represents the expected under a random untrained classifier. The pre-trained SlowFast backbones show superior performance.

seen in Fig. S5 these results show similar trends to those obtained on the entire naturalistic dataset with the pre-trained SlowFast models leading in both AuROC and AuPRC.

S3.5. Classifier error analysis

We investigated the possible sources of classification errors by analyzing the human annotations of the test dataset (N=4,563, see section S3.1) and the strike scores assigned to them by the two best models (SlowFast pre-trained on Kinetics and SSv2).

According to the human annotation of the test dataset, the detector had misclassified 118 samples as fish (∼2.5%), where no fish were found. Approximately 50% of the clips were labeled as high-quality clips featuring routinely swimming fish (n=1,841). Other distinct larval behaviors were those of abrupt movements (n=348) and non-routine swims (n=498; interrupted and reverse swimming, floating). The data also included a high percentage of low-quality clips with compromised footage (n=1,196), featuring over-exposed imagery, a moving background induced by strong flows, or extremely blurred fish.

The SSv2 pre-trained SlowFast performed consistently better than the Kinetics one in both AuROC and AuPRC scores (see Table S3) However when examining the distribution of strike scores per class (Fig. S6), it transpired that the improved performance of the SSv2 variant comes from a better mapping of the swim class to low strike scores (leading to a higher rate of true negatives), but this comes at a cost of missing ∼ 16% of the strike events (more false

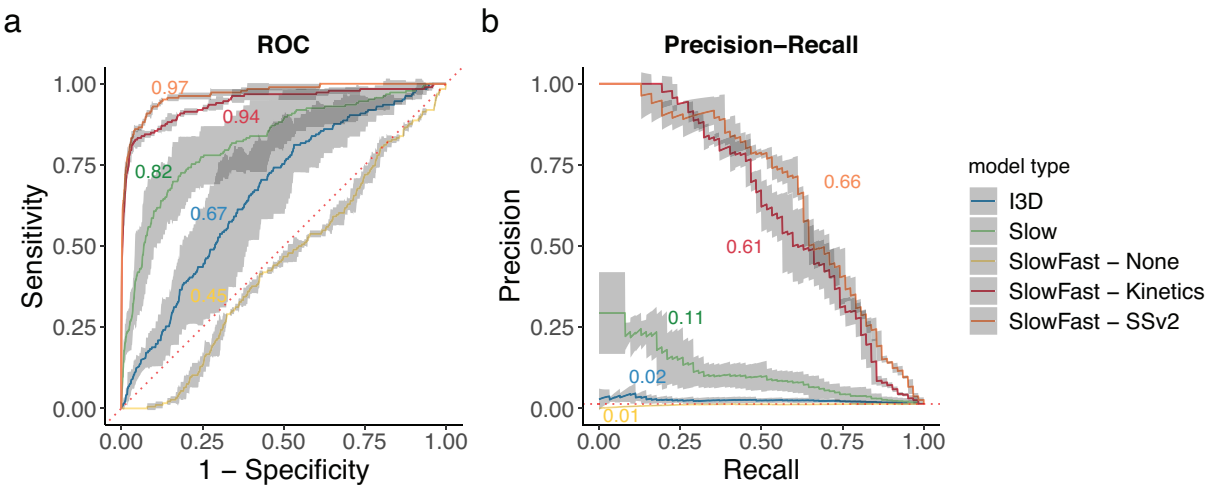

Figure S4: Comparison of backbones on the test dataset. a) ROC, plotting the True Positive Rate (Recall/Sensitivity) against the False Positive Rate (FPR/1-Specificity); b) PRC, presenting the Precision as a function of Recall. In both a & b dashed red lines represent the performance expected by a random classifier. Each curve is the mean ROC and PRC for each backbone, averaging performance on 3 different random seeds. The shaded area around each curve is the 95% confidence bounds. The I3D and Slow backbones (blue and green respectively) are pre-trained on Kinetics, the SlowFast are either not pre-trained (yellow) or pre-trained on Kinetics (red) or SSv2 (orange). The two pre-trained SlowFast backbones (Kinetics, SSv2) show superior performance by a large margin.

negatives). Conversely, the Kinetics variant is worse at mapping the swim class, with more potential false positive detections, but better at the strike class, with only 1-2 events being mapped to low strike scores (more true positives). Figure S6 shows that the kinetics variant was strongly affected by abrupt movements and low-quality clips, which were disproportionately mapped to higher strike scores > 0.5, potentially explaining some of the classification errors. The SSv2 variant assigned low strike scores to compromised footage but also gave disproportionately high scores to abrupt movements (Fig. S6 inset).

S3.6. Expected performance and decision threshold selection

We investigated the expected performance of our two top-performing models - Kinetics pre-trained SlowFast and SSv2 pre-trained SlowFast, by calculating the projected number of clips we'll have to review to recall a given % of the strike events in future data. To calculate this, we examine the per-class strike score distributions - the output of the classifier's "strike" neuron, for each clip in the "non-strike" and "strike" classes of the test dataset. The clips we will have to review are all those that have a score greater than the decision threshold, and they are the sum of the True Positives and the False Positives. We plotted this number as a function of the recall (S7) to select the decision threshold above which the % of clips to review increases sharply. We reasoned that a decision threshold of 0.75 provides a reasonable balance between review effort ($\sim$ 2% of the clips) and expected recall (0.8) (S7).

As can be seen, the SSv2 pre-trained model is better throughout, however, the gap in performance diminishes near the extremes of the plot (very low/high recall). To get 97 % recall, the Kinetics model will have a human analyze roughly 40% of the videos generated by the system, while the SSv2 will do the same work for around 31 %. It is true the Kinetics doesn't get a very low recall, due to the good mapping of the "strike" class to high strike scores, but the

price of False Positives is still so high, that at whatever point you choose on the graph, the SSv2 variant will yield fewer clips for the same recall.

S4. Detection module and the entire pipeline

Equipped with the classifier trained on the balanced curated dataset, we moved towards a pipeline for the analysis of longer videos rather than cropped clips. The pipeline comprised a detection module, followed by the classifier, as discussed next.

S4.1. Fish detector dataset and training

Inspired by the SlowFast paper (Feichtenhofer et al., 2019), we also used the Detectron2 framework (Wu et al., 2019) to train an object classifier to detect our fish, so that we can extract clips centered around individuals from full-frame videos. Below we describe our dataset for training this detector, the training procedure, and the results.

For the detection module, we trained a Faster-R-CNN (Ren et al., 2015) object detector with a ResNet-50-FPN backbone (He et al., 2016; Lin et al., 2017) using the Detectron2 framework (Wu et al., 2019). This detector was pre-trained on ImageNet (Deng et al., 2009) and fine-tuned on our detection dataset.

Dataset

Our fish detection dataset is composed of full frames (1920 × 1080 pixels) extracted from videos and annotated with bounding boxes around each individual fish. We extracted frames from 4 different raw videos from our dataset, to cover a range of filming conditions. From each video, we sampled a 10-minute sequence, within which we further sampled 1 frame every 0.33 seconds to allow variability in fish number and positions. The selected frames were then annotated by research assistants, who fitted a tight bounding box around each fish in the frame.

In order to ease the labeling process, initial guesses for bounding boxes were generated using the same Canny edge detection-based methodology used in creating the balanced curated "swim" class (see section 3.3.2 in the main text).

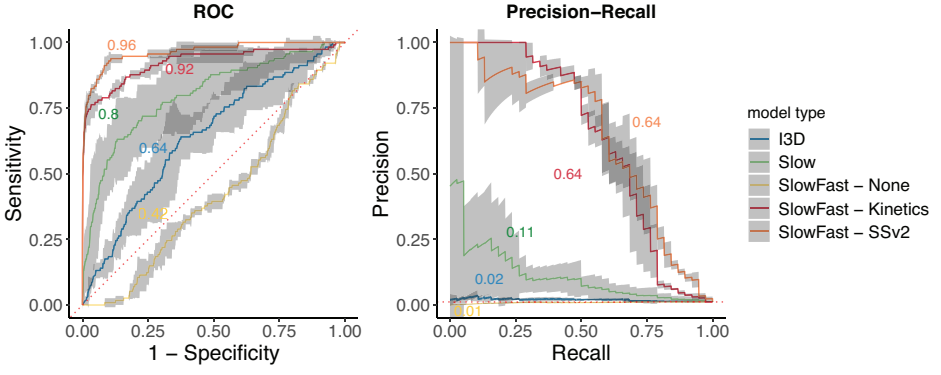

Figure S5: Evaluation of backbones on the naive subset of the naturalistic curated dataset. a) mean ROC; b) mean PRC. Results are shown for Kinetics pre-trained I3D (blue), Kinetics pre-trained Slow (green), SlowFast with no pre-training (yellow), SlowFast pre-trained on Kinetics (red), SlowFast pre-trained on SSv2 (orange). Each curve is a mean of three models trained on different random seeds. The shaded area around each curve is the 95% confidence interval. In both a & b dashed red lines represent the performance expected by a random untrained classifier. Results show similar trends to those shown on the entire dataset. The SlowFast backbones are still the best-performing.

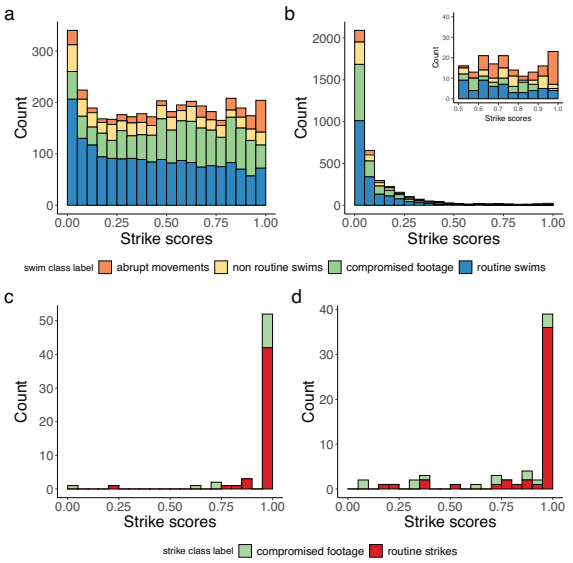

Figure S6: Error analysis. a,b) Distribution of strike scores and annotations for the naturalistic "swim" class as assigned by two SlowFast models a) Kinetics pre-trained, b) SSv2 pre-trained, inset is a close-up of the right tail of the distribution. Routine swimming appears in blue, compromised footage clips in green, non-routine swimming behavior (irregular, but not abrupt movements) in yellow, and abrupt movements (rapid, forceful movements) in orange. c,d) Distribution of strike scores for the naturalistic "strike" clips as assigned by the same models c) Kinetics pre-trained, d) SSv2 pre-trained. While SSv2 is better at classifying the "swim" clips, the Kinetics achieves superior results on the "strike" clips. Compromised footage and non-routine behaviors only partially explain misclassifications.

These initial guesses were then fine-tuned by the research assistants and any missing fish were added using the CVAT labeling tool (Sekachev et al., 2020). This process resulted in 1,664 annotated frames with 3,139 bounding boxes, split into 1,166 frames (2,196 boxes) in the train set, 245 frames (489 boxes) in the validation, and 253 frames (454 boxes) in the test set. Unlike in the action classification dataset, we randomly assigned frames from all raw videos into the partitions. This was done in order to create a detector that will work well under various filming conditions. Except for one video, none of the raw videos in the detection dataset were used in the classification dataset. The video used in both appeared in the train partition of the classification dataset.

Training procedure

Our detector was a Faster-RCNN (Ren et al., 2015) object detector with a ResNet-50-FPN backbone (He et al., 2016) (Lin et al., 2017). This detector was pre-trained on ImageNet (Deng et al., 2009) and fine-tuned on our detection dataset. We followed the recommended procedure in the official Detectron2 tutorial, for fine-tuning a pre-trained model on a custom dataset. We trained the model for 4,380 iterations, using 8 images per batch, a base learning rate of 0.00025, and, unlike the tutorial, we did not use warm-up in our training regime. For the full set of hyperparameters, and training procedures see the paper's repository.

Detection training results

The overall average precision (AP) and average precision at a threshold of 0.5 overlap (AP50) for our Faster-R-CNN fish detector are recorded in Table S4, it can be seen that using a lower threshold yields far better results. Accordingly, we chose to set the bounding box score threshold at 0.5 for our pipeline.

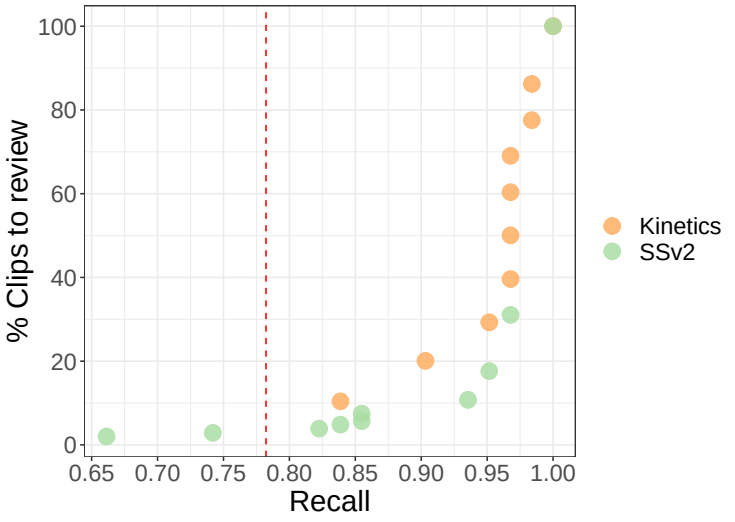

Figure S7: Expected workload for a human in the loop. The percent of clips to review (true positives and false positives) as a function of recall. The Kinetics pre-trained model (orange) consistently yields a higher amount of clips to analyze, compared to SSv2 (green). Red dashed represents the operational decision threshold chosen for the SSv2 classifier in pipeline deployment on the full videos.

Split	AP	AP50
Train	72.86	97.17
Validation	53.73	83.67
Test	50.39	85.05

Table S4
Fish detector training results for each split (first column), both average precision (AP, second column) and average precision at a threshold of 0.5 overlap with ground truth (AP50, third column), are shown

Overlapping clip removal

For frames containing many fish, it occurred that clips created based on the predictions of the fish detector were highly overlapping. To deal with this, we used a post-hoc heuristic to remove clips whose area was mostly overlapping, much in the spirit of non-maximal suppression. Please see our code for full details.

S5. Statistical analysis

In this section, we provide details on our statistical analysis.

S5.1. Video selection

Out of a total of 17 cohorts filmed, videos from one cohort were disregarded due to significant frame drops due to compression errors during or after acquisition. Though our classifier was able to find strikes even in these damaged videos, they were not analyzed manually and we couldn't tell how many strikes we were potentially missing by using our scheme on this damaged data. We further removed from our data ages under 8 DPH (n=8) and above 30 DPH (n=4) due to low sample sizes and the significantly different biology/behavior at these ages. This removal process resulted in the dataset of 223 videos discussed in this study. Specifically for the analysis of environmental effects, we removed 4 more videos for which we did not have temperature data, resulting in a sample size of 219 videos for the statistical analysis of response to environmental variables (section S5.3).

S5.2. Rarefaction Bootstrap analysis

The purpose of this analysis was to check whether we sampled sufficiently to characterize the feeding strike rates of the larval fish population we investigated. To this end, we employed a rarefaction style bootstrap analysis as follows; we divide our entire labeled and reviewed dataset (995 minutes in 223 videos) into segments of 20 seconds. In each segment, we counted the number of strike events detected by the SSv2 classifier (i.e., clips labeled as 'Strike' by annotators and assigned a strike score higher than 0.75 by the SSv2 classifier). Our rarefaction algorithm is specified in our GitHub repository.

In essence, for each iteration, we sample a certain amount of segments with replacement from our segment pool. Using these segments, we model the strike rate as a function of age group (count/Negative Binomial part) and mean density (zero/Bernoulli part) using a zero-inflated Negative Binomial model (using the R package pscl Zeileis et al., 2008) and look at the model coefficients as the rates as predicted by the model for each age group. We chose a zero-inflated model because even when our sampling unit was an entire video, the data were already plagued with zeros, this is exacerbated when our sampling unit is a 20-second segment. We chose these particular explanatory variables because age is known to affect larval feeding behavior, and the density captures other processes that might not be related to feeding but are dominating the chances of viewing false negative zeros in our data. Using the model's estimate of strike rates instead of bootstrapping the raw strike rates allows us to model our process of interest, strike rates, rather than other density-dependent processes that cause fish to not appear in the focal volume. In the rarefaction, we start by sampling the whole length of minutes available and reduce by 1 minute each time, we stop at 10 minutes as the sample size was insufficient below this point. Models in the lower sample sizes were unstable and had frequently failed to converge, in such cases a new sample was drawn. Additionally, to make sure all four age groups (08-14, 15-20, 21-25, 26-30 days post-hatch) were represented, at least one sample from each group was required in every sample drawn. The results of this analysis are brought in Figure 4 of the main text.

S5.3. Zero-inflated negative binomial details

We modeled the relationship between strike rates (strikes per fish per hour) and all environmental parameters (temperature, O_2, and pH) and age (in DPH, a continuous variable). As explained above in section S5.2, we used a zero-inflated Negative Binomial to account for the excess of zeros in our data. The model summary for our mode is brought in full in Table SS5.

In the main text, the predictions of the model are plotted for each observed temperature in our data (with the rest of the explanatory variables held at their median, Figure 5 in the main text). Here we plot the same predictions and curves and add the observed strike rates (Fig S8).

For completeness, we bring similar plots to illustrate the effects of oxygen (Fig SS9) and pH (Fig SS10), both of which were determined insignificant. The pH clearly has no effect on strike rates at the measured ranges, while oxygen does show a certain negative correlation whereby strike rates reduce with ascending oxygen concentrations.

S5.4. Robustness to changing classifiers

In our statistical analysis, to increase the power, we used all strikes we encountered; regardless of detection method (manual, type of classifier). To verify that we would achieve similar results by using our top-performing classifier, we performed the statistical analysis detailed above using only strikes detected by the SSv2 pre-trained classifier (Table SS6). Comparing the estimates of this model to those of the full model (Table SS5) we see that nearly all trends remain the same (i.e., no estimate changes from positive to negative or vice versa) aside from pH, which had standard errors that are x10 time larger than the estimate in the two models. The main difference is that effects are less significant when using just the SSv2 strikes, which is to be expected due to the smaller number of events detected.

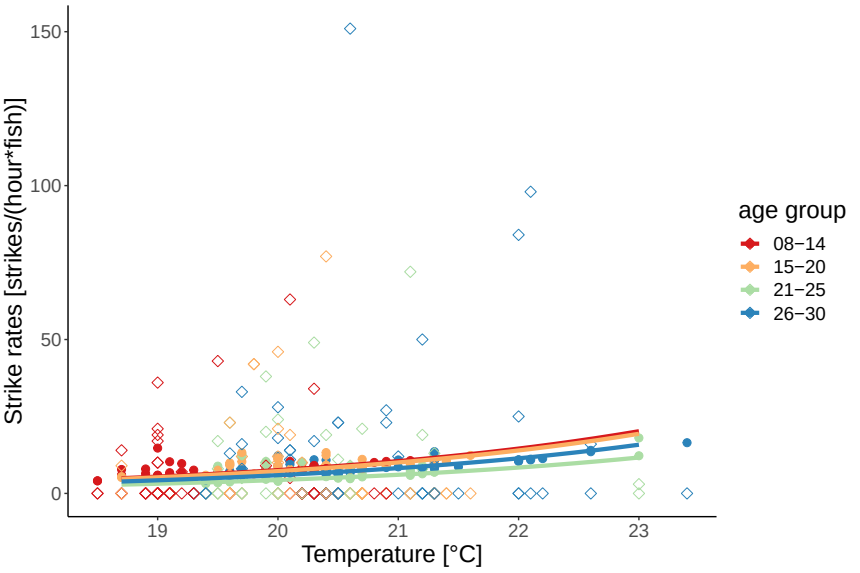

Figure S8: Modeled effect of temperature on strike rates. The strike rates (y-axis) predicted by a zero-inflated negative binomial model for the observed temperature gradient (x-axis) are plotted for each age group (color). The predictions of the model are plotted in full circles while raw observations are plotted in hollow diamonds. The predictions were calculated for the median values of oxygen, pH, and fish density. The curves are the model's fit across the temperature gradient (25 data points in the observed range). Temperature has a significant positive effect on strike rates, and differences between age groups were insignificant.

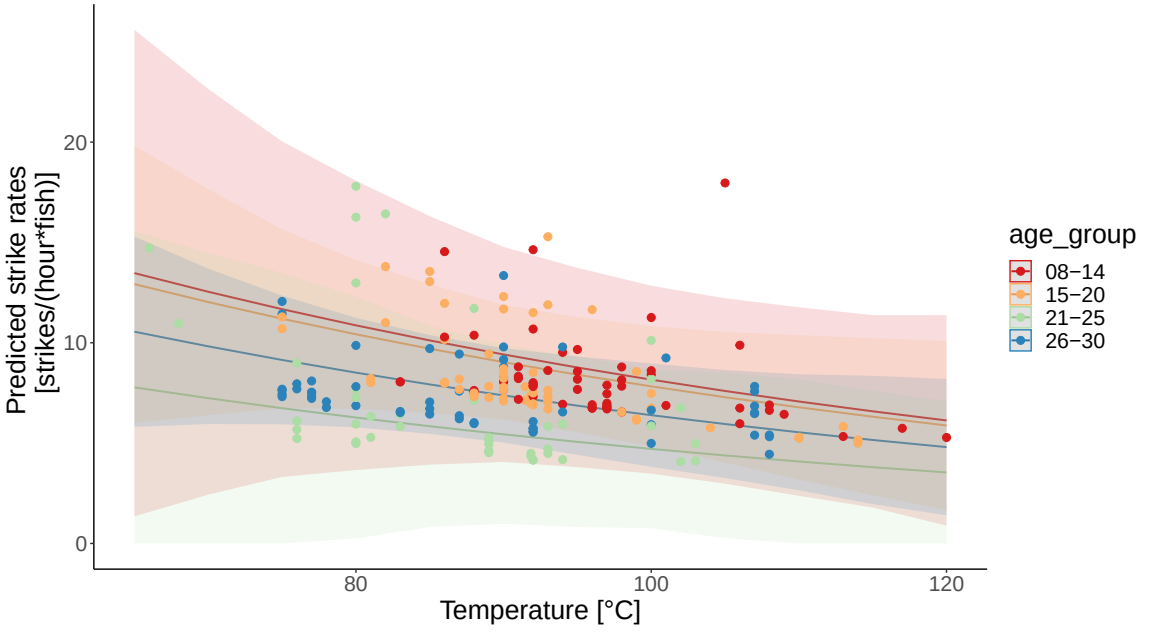

Figure S9: Modeled effect of oxygen on strike rates. The strike rates (y-axis) predicted by a zero-inflated negative binomial model for the observed oxygen gradient (x-axis) are plotted for each age group (color). Shaded areas are the 95 % confidence intervals. The model predictions were calculated for the median values of temperature, pH, and fish density. The round markers are model predictions for the measured oxygen in our data, and the curves are the model's fit across an oxygen gradient (25 data points in the observed range). Oxygen had an insignificant negative effect on strike rates, and differences between age groups were also insignificant.

Table S5
Results of a zero-inflated Negative Binomial model describing the relationship between strike rates and all environmental variables and age.

	Count model coefficients (negbin with log link):
	Estimate(Std.Error)
DPH	−0.027
	(0.019)
temperature	0.407***
	(0.151)
oxygen	−0.011
	(0.010)
pH	0.166
	(1.102)
Intercept	−4.828
	(10.298)
	Zero-inflation model coefficients (binomial with logit link):
	Estimate(Std.Error)
mean density	−0.232***
	(0.082)
Intercept	1.126***
	(0.194)
Observations	219
Log Likelihood	−413.188
Note:	*p<0.1; **p<0.05; ***p<0.01

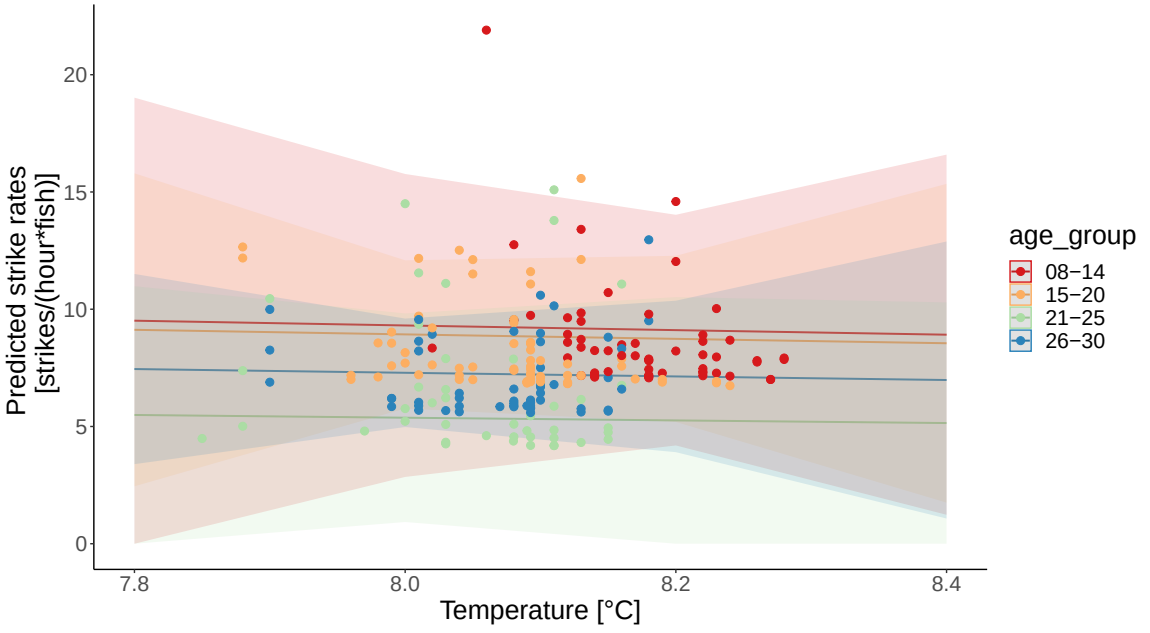

Figure S10: Modeled effect of pH on strike rates. The strike rates (y-axis) predicted by a zero-inflated negative binomial model for the observed pH gradient (x-axis) are plotted for each age group (color). Shaded areas are the 95 % confidence intervals. The model predictions were calculated for the median values of temperature, oxygen, and fish density. The round markers are model predictions for the measured pH in our data, and the curves are the model's fit across a pH gradient (25 data points in the observed range). pH had an insignificant negative effect on strike rates, and differences between age groups were also insignificant.

Table S6
Results of a zero-inflated Negative Binomial model using strike rates calculated using just the top performing classifier and estimating their relationship with all environmental variables and age.

	Count model coefficients (negbin with log link):
	Estimate(Std.Error)
DPH	−0.030
	(0.030)
temperature	0.448**
	(0.210)
oxygen	−0.003
	(0.016)
pH	−0.319
	(1.646)
Intercept	−2.515
	(15.139)
	Zero-inflation model coefficients (binomial with logit link):
	Estimate(Std.Error)
mean density	−0.126*
	(0.072)
Intercept	1.509***
	(0.209)
Observations	219
Log Likelihood	−297.021
Note:	*p<0.1; **p<0.05; ***p<0.01

References

S. Bar, L. Levy, S. Avidan, and R. Holzman. Analysis of larval fish feeding behavior under naturalistic conditions. *bioRxiv*, Unpublished results. doi: https://doi.org/10.1101/2022.11.14.516417.

J. Carreira and A. Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6299–6308, 2017. URL https://doi.org/10.1109/CVPR.2017.502.

J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. URL https://doi.org/10.1109/CVPR.2009.5206848.

C. Feichtenhofer, A. Pinz, and R. P. Wildes. Spatiotemporal multiplier networks for video action recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4768–4777, 2017. URL https://doi.org/10.1109/CVPR.2017.787.

C. Feichtenhofer, H. Fan, J. Malik, and K. He. Slowfast networks for video recognition. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 6202–6211, 2019. URL https://doi.org/10.1109/10.1109/ICCV.2019.00630.

K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. URL https://doi.org/10.1109/CVPR.2016.90.

T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2117–2125, 2017. URL https://doi.org/10.1109/CVPR.2017.106.

S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28, 2015. doi: 10.1109/TPAMI.2016.2577031. URL https://doi.org/10.1109/TPAMI.2016.2577031.

B. Sekachev, N. Manovich, M. Zhiltsov, A. Zhavoronkov, D. Kalinin, B. Hoff, TOsmanov, D. Kruchinin, A. Zankevich, DmitriySidnev, M. Markelov, Johannes222, M. Chenuet, a andre, telenachos, A. Melnikov, J. Kim, L. Ilouz, N. Glazov, Priya4607, R. Tehrani, S. Jeong, V. Skubriev, S. Yonekura, vugia truong, zliang7, lizhming, and T. Truong. opencv/cvat: v1.1.0, Aug. 2020. URL https://doi.org/10.5281/zenodo.4009388.

E. Shamur, M. Zilka, T. Hassner, V. China, A. Liberzon, and R. Holzman. Automated detection of feeding strikes by larval fish using continuous high-speed digital video: a novel method to extract quantitative data from fast, sparse kinematic events. *Journal of Experimental Biology*, 219 (11):1608–1617, 2016. doi: 10.1242/jeb.133751. URL https://doi.org/10.1242/jeb.133751.

N. Sommerfeld and R. Holzman. The interaction between suction feeding performance and prey escape response determines feeding success in larval fish. *Journal of Experimental Biology*, 222(17):jeb204834, 2019. doi: 10.1242/jeb.204834. URL https://doi.org/10.1242/jeb.204834.

Y. Wu, A. Kirillov, F. Massa, W.-Y. Lo, and R. Girshick. Detectron2. https://github.com/facebookresearch/detectron2, 2019.

A. Zeileis, C. Kleiber, and S. Jackman. Regression models for count data in R. *Journal of Statistical Software*, 27(8), 2008. URL http://www.jstatsoft.org/v27/i08/.