Article:
Chan, A.H.H. orcid.org/0000-0002-5405-7155, Putra, P. orcid.org/0000-0002-7632-375X, Schupp, H. orcid.org/0000-0002-1725-9129 et al. (11 more authors) (2025) YOLO-behaviour: a simple, flexible framework to automatically quantify animal behaviours from videos. Methods in Ecology and Evolution. ISSN 2041-210X

https://doi.org/10.1111/2041-210x.14502

Received: 28 August 2024 | Accepted: 10 January 2025

DOI: 10.1111/2041-210X.14502

BRITISH ECOLOGICAL SOCIETY Methods in Ecology and Evolution

RESEARCH ARTICLE

YOLO-Behaviour: A simple, flexible framework to automatically quantify animal behaviours from videos

Alex Hoi Hang Chan[1,2,3] | Prasetia Putra[1,4] | Harald Schupp[1,4] | Johanna Köchling[1,4] | Jana Straßheim[1,4] | Britta Renner[1,4] | Julia Schroeder[5] | William D. Pearse[5,6] | Shinichi Nakagawa[7] | Terry Burke[8] | Michael Griesser[1,2,3,9,10] | Andrea Meltzer[1,2,3,10] | Saverio Lubrano[1,2,3,10] | Fumihiro Kano[1,2,3]

[1]Centre for the Advanced Study of Collective Behaviour, University of Konstanz, Konstanz, Germany; [2]Department of Collective Behavior, Max Planck Institute of Animal Behavior, Konstanz, Germany; [3]Department of Biology, University of Konstanz, Konstanz, Germany; [4]Department of Psychology, University of Konstanz, Konstanz, Germany; [5]Department of Life Sciences, Imperial College London, Berkshire, UK; [6]The Alan Turing Institute, British Library, London, UK; [7]Department of Biological Sciences, University of Alberta, Edmonton, Alberta, Canada; [8]Department of Animal and Plant Sciences, University of Sheffield, Sheffield, UK; [9]Department of Zoology, Stockholm University, Stockholm, Sweden and [10]Luondu Boreal Field Station, Arvidsjaur, Sweden

Correspondence
Alex Hoi Hang Chan
Email: hoi-hang.chan@uni-konstanz.de

Funding information
Deutsche Forschungsgemeinschaft, Grant/Award Number: EXC 2117—422037984 and GR 4650/2-1

Handling Editor: Graziella Iossa

Abstract

1. Manually coding behaviours from videos is essential to study animal behaviour but it is labour-intensive and susceptible to inter-rater bias and reliability issues. Recent developments of computer vision tools enable the automatic quantification of behaviours, supplementing or even replacing manual annotation. However, widespread adoption of these methods is still limited, due to the lack of annotated training datasets and domain-specific knowledge required to optimize these models for animal research.
2. Here, we present YOLO-Behaviour, a flexible framework for identifying visually distinct behaviours from video recordings. The framework is robust, easy to implement, and requires minimal manual annotations as training data. We demonstrate the flexibility of the framework with case studies for event-wise detection in house sparrow nestling provisioning, Siberian jay feeding, human eating behaviours and frame-wise detections of various behaviours in pigeons, zebras and giraffes.
3. Our results show that the framework reliably detects behaviours accurately and retrieve comparable accuracy metrics to manual annotation. However, metrics extracted for event-wise detection were less correlated with manual annotation, and potential reasons for the discrepancy between manual annotation and automatic detection are discussed. To mitigate this problem, the framework can be used as a hybrid approach of first detecting events using the pipeline and then manually confirming the detections, saving annotation time.

4. We provide detailed documentation and guidelines on how to implement the YOLO-Behaviour framework, for researchers to readily train and deploy new models on their own study systems. We anticipate the framework can be another step towards lowering the barrier of entry for applying computer vision methods in animal behaviour.

KEYWORDS
animal behaviour, behavioural recognition, computer vision, machine learning

1 | INTRODUCTION

Ever since the popularization of video cameras, animal researchers have been using videos to capture the behaviours of animals in captivity and in the field. Further propelled by user friendly video annotation tools like BORIS (Friard & Gamba, 2016), taking videos of animals and subsequently annotating for specific behaviours have become essential parts of data collection pipelines in animal behaviour. However this approach is time consuming (Chan, Liu, et al., 2024) and can be susceptible to low observer reliability and repeatability (Tuyttens et al., 2014). To solve these problems, advances in computer science have leveraged large video datasets to create computer vision-based solutions to automate the quantification of behaviours from animal videos, leading to a significant shift in the scale and efficiency of extracting behaviours from video data (Couzin & Heins, 2022; Mathis & Mathis, 2020).

There are a few general approaches for automatically quantifying animal behaviours from videos. We summarize published open-source toolboxes, along with their training data requirements, advantages, and disadvantages in Table 1. The first approach starts with 2D or 3D keypoint estimation on animals in a video frame, then uses supervised (Goodwin et al., 2024; Wittek et al., 2022) or unsupervised (Graving & Couzin, 2020; Hsu & Yttri, 2021) methods to quantify behaviours using predicted keypoint information. Keypoint estimation of animal body parts from videos has recently been popularized with the development of tools including DeepLabCut (Lauer et al., 2022; Mathis et al., 2018), SLEAP (Pereira et al., 2022) and DeepPoseKit (Graving et al., 2019), allowing fine-scaled body postures of animals to be measured precisely. However, keypoint estimation methods are often limited to captive settings (but see Chimento et al., 2024; Joska et al., 2021; Waldmann et al., 2024; Wiltshire et al., 2023), and obtaining large keypoint ground truth datasets is often labour intensive.

The second approach is to directly input video frames into neural networks, and output observed behaviours. With recent benchmark datasets like animal kingdom (Ng et al., 2022), KABR (Kholiavchenko et al., 2024), PanAf20k (Brookes et al., 2024) or MammalNet (Chen et al., 2023), there is a growing trend of directly using video input for behavioural classification, even though the accuracy of such methods is often low (e.g. 50%–60%; Kholiavchenko et al., 2024, table 1). Some open-source tools also leverage optical flow and feature extractors of video sequences for behavioural quantification (Bohnslav et al., 2021; Harris et al., 2023), but their use may be limited to controlled laboratory settings with single animals. Finally, a method for behaviour classification in more visually noisy scenes is to first isolate an animal in the video frame with a bounding box or mask, then input the cropped animal into a neural network classifier to classify behaviours (Lei et al., 2022; Yang et al., 2019). For example, tools like LabGym (Goss et al., 2024; Hu et al., 2023) first extract the contours of animals of interest, then generate a movement pattern image for behavioural classification. While these methods are promising, such approaches require among others segmentation masks and behavioural annotations of sequences as training data, which can be laborious to collect.

Computer vision methods have been shown to be useful for quantifying behaviours in different species, but there is a lack of agreement on the most effective method for any given dataset and study system. Yet, especially under a rapidly changing climate and biodiversity crisis, it is more important than ever to leverage developments in computer vision to aid data collection on fundamental behavioural monitoring (Christin et al., 2019; Tuia et al., 2022), including individual level behaviours such as feeding rates, visit rates or activity budgets, up to population level metrics. Such advances are not only important for deepening our understanding of biological systems, but to also gain insight into species conservation, and importantly increase the efficiency and wealth of data that can be collected and processed (Dell et al., 2014; Weinstein, 2018). However, while computer vision tools can be highly capable, they are often effective only in very specific contexts. These contexts are defined by unique characteristics such as the experimental setup, camera angles, lighting conditions, subject size and occlusions, all of which influence the suitability of a given method. Moreover, most frameworks require large amounts of effort to collect training data, and sophisticated workflows to achieve automated behavioural coding (e.g. first training a keypoint model, then fitting an unsupervised algorithm, followed by training a supervised classifier; see Hsu & Yttri, 2021). Consequently, the domain-specific nature of current open-source computer vision algorithms poses a significant barrier to their widespread adoption by biologists and psychologists.

To overcome these limitations, we present the YOLO-Behaviour framework, an automatic behavioural detection and classification tool based on the common object-detector YOLOv8 (Jocher et al., 2023). An object detector is a class of models in computer

TABLE 1 Comparison of existing methods with YOLO-Behaviour.

Method	Annotations required	Types of models that require training	Case study types	Number of animals with behaviour detected simultaneously	Type of input to model	Pros	Cons	Citation
DeepEthogram	Frame-wise behaviours in video	Optic flow extractor Image feature extractor Sequence model	Lab	Single	Video	Optical flow captures temporal information GUI for annotation and training	Multiple training steps Single animal	Bohnslav et al. (2021)
DeepAction	Behaviours of short video clips	Classification model	Lab	Single	Video	GUI for labelling and training	Single animal MATLAB based (not open sourced)	Harris et al. (2023)
LabGym	Frame-wise behaviour in videos	Categorizer model	Lab	Single	Video	GUI for labelling and training Captures movement changes to classify behaviours	Require detailed behavioural annotation Require static camera	Hu et al. (2023)
LabGym2	Segmentation masks Behaviour of short sequences for each individual present	Detector model Categorizer model	Lab + field	Multiple	Video	Multiple animals Works with field data Can identify social behaviours	Require mask and behavioural annotation	Goss et al. (2024)
SIMBA	Keypoints Frame-wise behaviour	Keypoint estimation model Random forest classifier	Lab	Multiple	Video	GUI for labelling and training SHAP values for interpretability	Require multiple frameworks and workflow, including keypoint and behavioural annotation	Goodwin et al. (2024)
YOLO-Behaviour (Ours)	Bounding box of behaviours in single frames	Object detection model	Lab + field	Multiple	Single frames	Simple training data, easy to train Fast inference speed Multiple animals	Model uses single frames, requires visually distinctive behaviours	Current paper

vision that aims to localize an object within an image, by predicting a bounding box around a given object and its class. We leverage this model type to detect visually distinct behaviours in static frames, by providing training data of bounding boxes around behaviours within an image. The simplicity and robustness of the framework allows for widespread training and deployment by biologists with minimal training data and coding expertise, as demonstrated by the five case studies across study systems and taxa, which includes event-wise detection in (1) house sparrow (*Passer domesticus*) nestling provisioning, (2) Siberian jay (*Perisoreus infaustus*) feeding, (3) humans (*Homos sapiens*) eating; and frame-wise detections in (4) foraging pigeons (*Colomba livia*) as well as (5) roaming zebras (*Equus quagga* and *E. grevyi*) and giraffes (*Giraffa camelopardalis*). Furthermore, we provide detailed code and documentation to facilitate its implementation in new study systems. We hope the simplicity of the proposed pipeline can promote the adoption of computer vision in animal researchers, thereby reducing the time required for manual coding in behavioural research and species monitoring.

2 | MATERIALS AND METHODS

We first describe the datasets we used to evaluate the YOLO-Behaviour framework, and the ways we categorized the datasets for evaluation. Next, we describe the pipeline, from training to post-processing and to optimization. Finally, we introduce the methods for evaluating the pipeline in terms of its ability to detect events accurately, retrieve coded behavioural metrics and comparison with an existing tool.

2.1 | Datasets

We tested the robustness and generalization ability of YOLO-Behaviour by applying the method across five study systems across various taxa (Figure 1). Table 2 shows details of each dataset used, behaviours coded and size of training validation and test sets. We refer to the supplementary methods for detailed justification and description of each study system and data manipulation procedures. For each case study, we defined three dataset types, which differ slightly from conventional data splitting procedures. (1) Training images: annotated images used for training the YOLOv8 models. (2) Event validation set, a small number of videos for optimizing hyperparameters and detailed evaluation of detection accuracy. (3) Coded behavioural metrics dataset, the largest dataset available to evaluate how the method can estimate coded behavioural metrics (e.g. feeding or visit rate), when compared with human annotation. All annotations from the training set are publicly available, except for the human dataset, which will not be available due to privacy and ethical concerns.

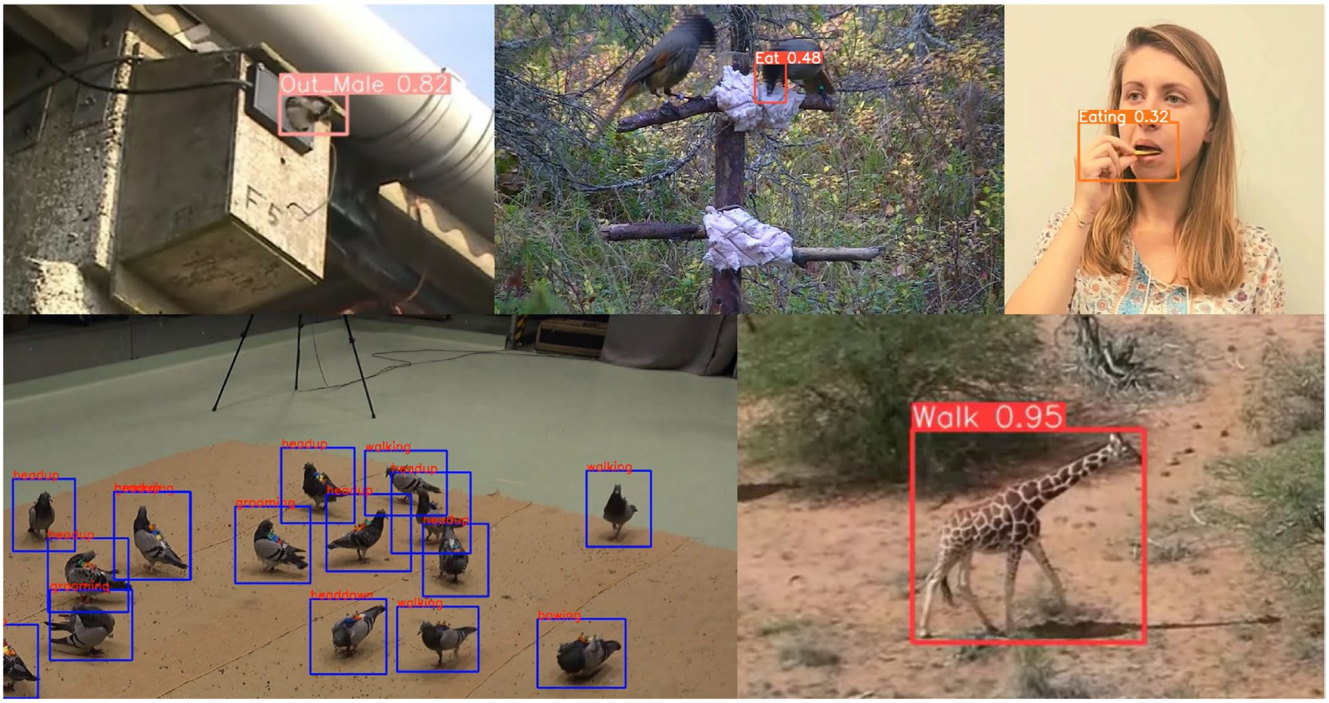

FIGURE 1 Case studies used to test the YOLO-Behaviour framework. Predictions of the YOLO model is overlayed onto each case study, with the predicted bounding box, class and model confidence. (a) House sparrow (*Passer domesticus*) provisioning videos collected on Lundy Island, UK. (b) Siberian jay (*Perisoreus infaustus*) feeding videos collected in Swedish Lapland. (c) Human (*Homo sapiens*) eating dataset, collected in Konstanz, Germany. Presented image is a sample, actual used images cannot be published due to data privacy. We have written consent by the shown subject to use the image for demonstration purposes. (d) Homing pigeons (*Colomba livia*) behaviours collected in Möggingen, Germany, based on the 3D-POP dataset. (e) Giraffes (*Giraffa camelopardalis*) behaviours collected in Mpala research centre, Kenya, part of the KABR dataset.

TABLE 2 Details of datasets used in the current study.

Dataset type	Dataset name	Study species	Location	Coded behaviours	Training images	Event validation set	Coded behavioural metrics dataset	Citation
Event-wise Detection	Sparrow provisioning	House sparrows (*Passer domesticus*)	Lundy Island, UK	Male Out Female Out In Around	1505	1506 videos (7 s long)	779 videos (1.5 h each)	Chan, Liu, et al. (2024)
Event-wise Detection	Jay feeding	Siberian jays (*Perisoreus infaustus*)	Swedish Lapland, Sweden	Eat	1567	5 videos (~30 min each)	260 videos (~30 min each)	~
Event-wise Detection	Human collective eating	Humans (*Homo sapiens*)	Konstanz, Germany	Eat	2216	10 videos (~15 min)	64 videos (~15 min)	~
Frame-wise Detection	3D-postures of pigeons (3DPOP)	Homing pigeons (*Colomba livia*)	Möggingen, Germany	Walking Head-up Head-down Grooming Bowing	1587	5 videos, 271,485 frames	59 videos (1–2 min each)	Naik et al. (2023)
Frame-wise Detection	In-situ dataset for Kenyan animal behaviour recognition (KABR)	Giraffes (*Giraffa camelopardalis*), Plain zebras (*Equus quagga*), Grevy's zebras (*Equus grevyi*)	Kenya	Walk Graze Browse Head-up Groom Trot Run	1400	972 videos, 285,130 frames	184 videos (mean 52 s)	Kholiavchenko et al. (2024)

All data presented was collected in accordance with relevant ethical permits. For the Lundy house sparrow provisioning videos, no primary data were collected as part of this study. Data from the Lundy Island sparrows is collected under a British Trust for Ornithology bird ringing permit and with permission from the Lundy Company and Field society; most recently: UK Home Office (PP7009092 and PP5873078) and BTO (S:6308). Experiments and observations that contributed to the Siberian Jay eating datasets were approved by Umea ethics board, A23-20. Bird ringing was done under the licence of the Swedish Museum of Natural History. For the human eating dataset, data collection was part of the Collective Appetite research project in the Centre for the Advanced Study of Collective Behaviour and was conducted in accordance with the guidelines of the German Psychological Society and the Helsinki Declaration. The study protocol was approved by the University of Konstanz's Ethics Committee (24/2020). 3D-POP and KABR case studies were publicly available datasets, so we refer to the original publication for information on the appropriate ethical approvals (Kholiavchenko et al., 2024; Naik et al., 2023).

2.2 | YOLO-Behaviour

The complete YOLO-Behaviour framework can be separated into three parts. (1) Data annotation and training: images were annotated and a YOLOv8 model was trained. (2) Post-processing: YOLO detections were processed and grouped using a tracking algorithm. (3) Optimization: a grid-search algorithm was used on the validation dataset to determine the best hyper-parameters for the final pipeline. We describe each of the steps in detail below. All code and documentation to apply the framework to novel systems can be found in the following link: https://github.com/alexhang212/YOLO_Behaviour_Repo.

2.2.1 | Data annotation and training

First, random frames were sampled from each dataset, and bounding boxes were manually annotated for behaviours of interest, ensuring the bounding box encloses a visually distinctive part of the image that characterizes the behaviour. For example, in the Siberian jay and human eating datasets, the eating behaviour was captured by annotating a bounding box around the hand/beak touching the mouth/food respectively (Figure 1b,c). For the pigeon and zebra/giraffe datasets, no manual annotation was done, since bounding boxes and behavioural labels were extracted using the dataset provided. We refer to supplementary methods for a detailed description of each dataset. After frames were extracted and annotated, the data were further split into training, validation and test sets using a 70%, 20%, 10% split. A YOLOv8-large model pre-trained on the COCO dataset (Lin et al., 2014) was then trained using the Ultralytics python package (Jocher et al., 2023), with default augmentation and training parameters. The default augmentation pipeline includes random adjustments to hsv image space, random translation and horizontal flip, and mosaic by combining multiple images.

2.2.2 | Post-processing

Once the YOLO models were trained, the models were used to detect behaviours from videos. However, the raw output of YOLO are bounding boxes with a given label and position in the frame, which is uninformative and can represent multiple behaviours being detected at the same time (e.g. two jays pecking at the food from both sides). In addition, the YOLO-Behaviour framework is also susceptible to short bursts of erroneous detections, such that behaviours are sometimes misclassified for a small number of frames. To solve these problems, we used the tracking algorithm SORT (Bewley et al., 2016) to group bounding box detections across spatial and temporal scales, by connecting closely detected bounding boxes of the same behavioural class as a single track. SORT is a widely used multi-object tracking algorithm, which is traditionally used for tracking the trajectory of detected objects in a video (e.g. human pedestrians walking in a video) and is well known for its short processing time and simplicity. Here, instead of tracking objects across the screen, we make use of the SORT to group bounding boxes of the same class to be classified as a single behavioural event. In this way, spatially close behavioural detections across frames can be combined as a single behavioural event, and short incorrect detections will not be assigned to any tracks and effectively filtered out. This post-processing step is crucial for the flexibility of YOLO-Behaviour, as it allows for multiple behaviours that are occurring in different parts of the video to be connected and recorded. As the output of the post-processing pipeline, users will obtain behavioural events of a given behavioural class, with its corresponding duration, start, end frames and bounding box locations.

2.2.3 | Optimization

Finally, the pipeline was optimized by selecting the best hyper-parameters using a grid-search algorithm. A grid-search algorithm is a brute-force algorithm that searches a user-defined hyper-parameter space to find the most optimal parameters. Here, the YOLO model was used for inference in the event validation set for each study system, and a range of hyper-parameters were defined manually. The hyper-parameters include: YOLO confidence threshold, which is the confidence threshold for a bounding box detection to be considered as a valid detection; minimum duration, which describes the minimum frame number of an event; and three separate thresholds for the SORT tracking algorithm, including min hits (minimum frames to define new track), max age (maximum frame gaps to connect two detections) and IOU threshold (intersection over union overlap to associate bounding boxes). These hyper-parameters influence how multiple behavioural detections are grouped together as behavioural events from the SORT tracker. After defining the range of hyper-parameters to explore, we computed the f1-score (i.e. summary score that balances precision and recall; see Table 3) for every possible combination of parameters, and then selected the best combination to be used in the final pipeline. In addition, we also determined the best combinations to obtain lowest false negative rates for the event detection case studies, and test whether the framework can be used in hybrid applications. We selected parameters that minimized the false negative rate since that would be the model where the most events were captured for further manual review. We did not optimize for false negative rates in the two frame-wise detection datasets because there is a behaviour prediction every frame, such that a hybrid application will entail reviewing every frame, which is unrealistic.

2.3 | Evaluation

2.3.1 | Event detection

First, we evaluated the reliability of event detection using the event validation dataset for each case study. We used the best parameters obtained from the grid search algorithm for inference, then extracted overall detection accuracy, precision, recall, false negative rate and f1-score (Table 3). Given the disparate characteristics of the datasets, it was necessary to utilize different definitions of 'events' for the purposes of evaluation. (1) Sparrow provisioning: each event was defined as a behaviour detection within a 7-s long video (see Chan, Liu, et al., 2024). (2) Jay feeding: each event was defined as 2s windows across the whole video to take into account possible human reaction delay when pressing a button in BORIS, compared to the frame-by-frame detections of YOLO. Detections were matched as whether a feeding event is present or not within each window. We also report results for a range of time windows to compare how window definition affects evaluation (Table S4). (3) Human eating: each event was defined as a 1s window due to similar delay in human reaction when coding in BORIS, and detections were matched if an eating event is present within the window. (4) 3D-POP/KABR: Since frame-wise annotations are available, each event is defined as a detection in a single frame.

In addition, we evaluated whether the size of the training dataset will affect the reliability of event detections, by training multiple models using a data subsets representing 20%, 40%, 60% and 80% of the original training datasets. To determine how accuracy improves with increasing dataset size, we assessed the f1-score, precision and recall of these models using the same event validation dataset. Furthermore, we investigated whether the training dataset

TABLE 3 Definitions of evaluation metrics used in the current manuscript.

Metric	Definition	Equation
Accuracy	Proportion of predictions that are correct overall	$\frac{TP+TN}{TP+TN+FP+FN}$
Precision	Proportion of model detections that are correct, indication of how well the model can predict the correct behaviour	$\frac{TP}{TP+FP}$
Recall	Proportion of ground truth labels that were detected, indication of how well the model can return all behavioural events	$\frac{TP}{TP+FN}$
False negative score	Inverse of recall, proportion of ground truth labels that were not detected by the pipeline, indication of how many events were completely missed by the pipeline	$\frac{FN}{TP+FN}$
F1-score	Summary score that balances precision and recall	$2\times\frac{\text{Precision}\times\text{Recall}}{\text{Precision}+\text{Recall}}$
Inter-class correlation (ICC)	Correlation metric to quantify inter-rater reliability for continuous data, indicative of how well manual and YOLO-behaviour agrees with each other for event detection	$\text{ICC}(3,1)=\frac{\sigma_r^2}{\sigma_r^2+\sigma_c^2+\sigma_e^2}$ σ_r^2: variance explained by class σ_e^2: variance explained by raters σ_c^2: residual variance (error)

Note: TP: stands for true positives (the model correctly predicts that a given behaviour is present), TN: true negatives (the model correctly predicts that a given behaviour is absent), FP: false positives (the model incorrectly predicts that a given behaviour is present), FN: false negatives (the model incorrectly predicts that a given behaviour is absent), respectively. All metrics ranges from 0 to 1, with higher values representing higher performance.

requirement could be reduced if the base model was pre-trained to identify animals. For this purpose, we applied the same training data proportion intervals with the KABR case study, but fine-tuned the MegaDetector model (Beery et al., 2019), a large pre-trained model to identify animals from camera trap images, based on an earlier YOLOv5 architecture.

2.3.2 | Extracting coded behavioural metrics

We used the coded behavioural metrics dataset for each case study to determine whether YOLO-behaviour is reliable for extracting behavioural metrics. For event detection case studies, we extracted feeding rates from jay eating (seconds spent eating per minute per individual) and human eating (food eaten per minute), as well as male and female visit rates (visits per hour) for the sparrow provisioning dataset. For frame-wise detection case studies in 3D-POP and KABR, we extracted the proportion of time spent on each behaviour. We then calculated the Pearson's correlation for each case study, as well as intraclass correlation coefficients (ICC3), which is used to evaluate inter-rater reliability for continuous variables (Gwet, 2014). Since ICC assumes data normality and homogenous variance (Bobak et al., 2018), we log transformed sparrow visit rates and human eating rates, as well as logit transformed proportions from the frame-wise detection datasets and visualizing data distributions to ensure these assumptions were met.

2.3.3 | Comparison with DeepEthogram

Finally, we compared the classification performance and inference speed of YOLO-Behaviour with DeepEthogram, an existing open source tool for automated behavioural classification from videos (Bohnslav et al., 2021). For the comparison, we chose to use the KABR dataset, as it provides frame-wise annotations with a single behaviour label per frame. We did not include the Siberian jay eating and pigeon behaviour case studies, as these involve multiple behaviours occurring within the same frame. Similarly, we excluded the sparrow provisioning and human eating case study, as their annotations are limited to randomly sampled frames. To ensure a fair comparison, we used the same images that trained our YOLO model but used the entire 3 s videos (90 frames) to train DeepEthogram due to requirements for short sequences. We trained DeepEthogram-medium following the provided training protocols, then evaluated both models with the same event validation dataset (Table 2). While the datasets used for each framework differ, Deepethogram receives sequential information, whereas YOLO processes bounding box data, we believe this comparison remains valuable for future users who are deciding which tool to choose. To calculate the inference speed, we ran each model through a video three times and reported the average speed in terms of frames per second. For calculating inference speed, we use a workstation with a 16GB Nvidia Geforce RTX 3070 GPU, 11th Gen Intel(R) Core(TM) i9-11900 H @ 2.50GHz CPU, and Sandisk 2TB SSD.

3 | RESULTS

We applied and evaluated the YOLO-Behaviour framework over five case studies. All YOLO model training evaluation can be found in Table S1, and all datasets and annotation used are available via https://doi.org/10.17617/3.EZNKYV (Chan, Putra, et al., 2024). Qualitative results can be found in the Video S1.

We found that the YOLO-Behaviour framework is accurate across all study systems and case studies in the event validation set (Table 4), with an f1-score ranging between 0.62–0.94 and accuracy ranging between 0.70–0.98. Figure 2 shows the confusion matrices for each case study, also highlighting the high accuracy and consistency of most behaviours detected when using YOLO-Behaviour, with some exceptions. Particularly, eating detection for the human dataset is relatively low (0.6, Figure 2c), which is likely due to the inherent difficulty in distinguishing eating from other hand gestures involving the mouth region. In the KABR dataset, the accuracies for locomotion-based behaviours (walking, trotting, running) are more variable (0.27–0.73), as well as browsing and auto-grooming behaviour (0.069, 0.37 respectively, Figure 2e).

When varying the proportion of the total training dataset, we observed that the accuracy metrics initially increased with the availability of more training data, followed by slight plateaus as expected (Figures S1A and S2). The human case study was the most sensitive to changing dataset size, with high recall but low precisions at low training dataset size, indicative of false positives (Figure S2). For

TABLE 4 Evaluation metrics on validation set.

Dataset	Weighted average precision	Weighted average recall	Weighted false negative rate	Weighted average f1-score	Accuracy
Sparrow provisioning	0.78	0.79	0.21	0.78	0.71
Jay feeding	0.94	0.95	0.05	0.94	0.91
Human eating	0.63	0.60	0.40	0.62	0.98
3DPOP	0.77	0.70	0.30	0.72	0.70
KABR	0.77	0.73	0.27	0.75	0.73

Note: Presented metrics are the class proportional weighted average scores for the behaviour of interest, which excludes 'not feeding' for jay and 'not eating' for human datasets. For a description of each metric and its definition, we refer to Table 3.

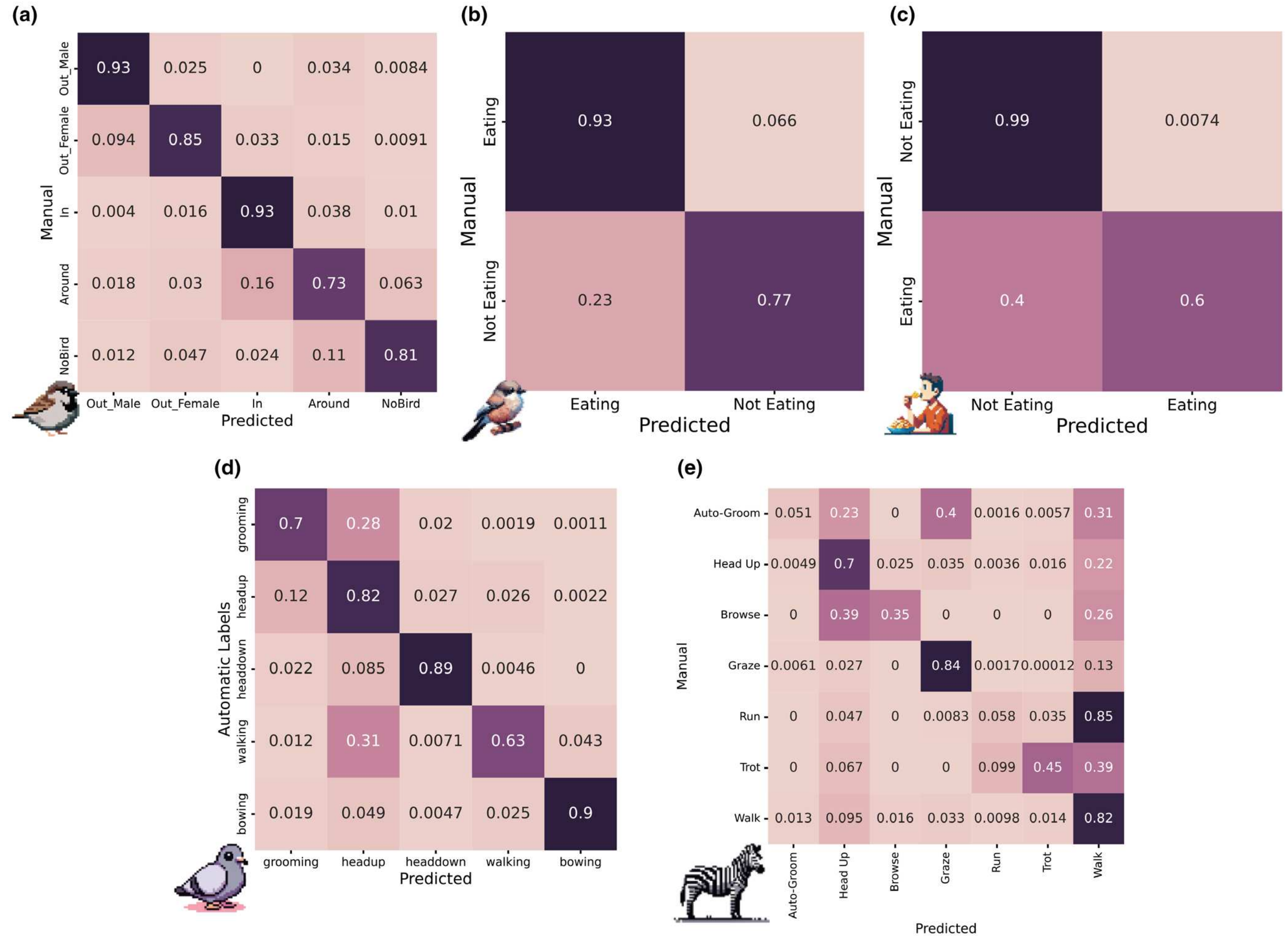

FIGURE 2 Confusion matrices of per-class classification accuracy across five case studies. Confusion matrices are designed to visualize discrepancies between predicted and annotated classes, helping to identify if certain classes are more frequently mislabelled than others. In each panel, the *x*-axis represents the predicted classes, and the y-axis represents the labelled classes. The numbers indicate the proportion of labelled data predicted as each behaviour, with correct predictions shown along the diagonal. Proportions were generated from the validation set of each case study, using parameters optimized for f1-score using a grid search algorithm. (a) House sparrow provisioning, (b) Siberian jay eating, (c) human eating, (d) pigeon behaviours from 3D-POP and (e) zebra and giraffe behaviours from KABR. Pixel art of each animal generated using Dall-E 3.

the KABR dataset, fine-tuning YOLOv8 outperforms MegaDetector across all data subsets except for the lowest proportion of 0.2 (Figure S1B). This indicates that leveraging a base model pre-trained on animals does not compensate for the architectural advancements between YOLOv5 and YOLOv8.

Through the metrics obtained from the coded behavioural metrics dataset, we show that the extracted metrics all significantly correlate with manual annotation (Figure 3), with frame-wise detection case studies having higher correlations and ICC3 values (Table 5; 0.78–0.92) compared to event detection (Table 5; 0.49–0.70), corresponding to 'good' to 'excellent' reliability for frame-wise detections and 'moderate' to 'good' reliability for event-wise detection (Koo & Li, 2016). Figure 3 also shows a general under-detection across all case studies.

We then compared YOLO-Behaviour with an existing open-source tool DeepEthogram, on the KABR dataset and found that the accuracy metrics are generally comparable between the two methods, even though YOLO-Behaviour being slightly more accurate (Table 6). In addition, the inference speed for YOLO-Behaviour was also faster compared to DeepEthogram (52.4 fps vs. 35.9 fps).

Finally, we tested whether the YOLO-Behaviour framework can be used in a hybrid approach by optimizing for low false negative rates instead of f1-score. We found that the method can obtain low false negative rates between 0.05–0.14 (Table 7), showing that only around 5%–15% of the overall events will be missed. Particularly in the human eating case study, the precision is 0.26, which shows that the model is full of false positives that can be manually corrected in the hybrid framework.

4 | DISCUSSION

In the current study, we presented the YOLO-Behaviour framework, a simple to implement and flexible framework for automated

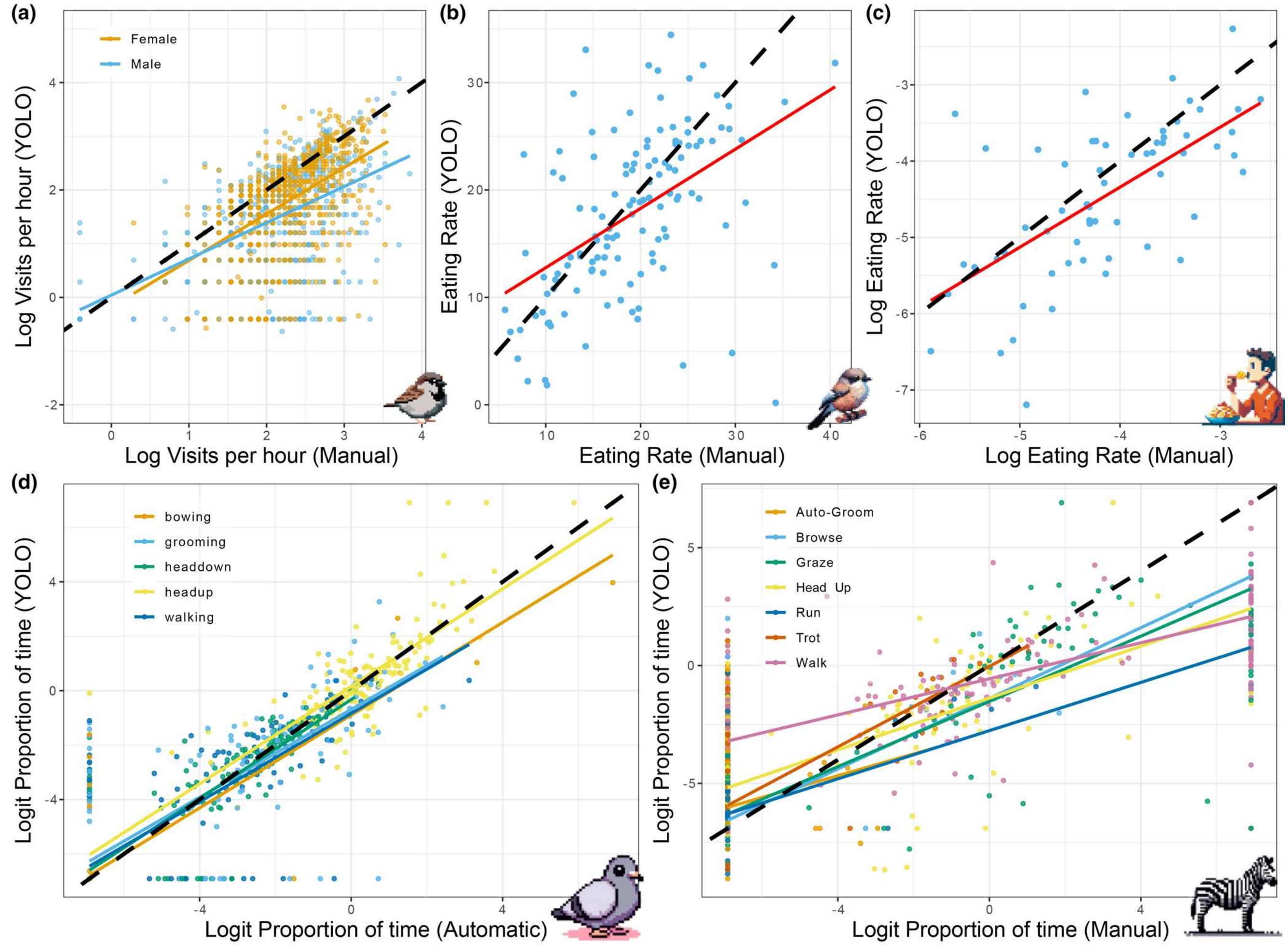

FIGURE 3 Correlations of the coded behavioural metrics across five case studies. Metrics were extracted from the coded behavioural metrics dataset of each case study. The *y*-axis showing automatic rates from YOLO and the *x*-axis showing rates from manual or labels from corresponding datasets. Black line in each plot represents the 1:1 correlation line. (a) Correlation of visit rates (visits per hour) in the sparrow provisioning dataset, separated by male and female visit rates. (b) Correlation of feeding rate (seconds spent feeding per minute per individual) in the Siberian jay dataset. (c) Correlation of eating rate (food eaten per minute) of the human eating dataset. (d) Correlation of the proportion of time spent for five separate behaviours in the pigeon 3D-POP dataset. (e) Correlation of the proportion of time spent for seven separate behaviours in the giraffe/zebra KABR dataset. Pixel art of each animal generated using Dall-E 3.

TABLE 5 Pearson correlation and intraclass correlation (ICC3) values for coded behavioural metrics across five case studies.

Dataset	Metric	Pearson's correlation	ICC3
Sparrow provisioning	Visits per hour	0.51	0.46
Jay feeding	Seconds spent feeding per minute per individual	0.50	0.49
Human collective eating	Food eaten per minute	0.70	0.69
3DPOP	Proportion of time per behaviour	0.92	0.92
KABR	Proportion of time per behaviour	0.78	0.77

Note: Metrics for each case study were extracted from the coded behavioural metrics dataset. We refer to Table 2 for description of the datasets and Table 3 for the definition of ICC3. All Pearson's correlation values were significant ($p < 0.05$).

coding for basic animal behaviours from videos. We illustrated the robustness of the framework with the high detection accuracy across a large range of study systems and video types. The framework is easy to train and implement, with the full documentation and user guidelines for applying to a new system available here: https://github.com/alexhang212/YOLO_Behaviour_Repo.

TABLE 6 Evaluation metrics on the validation set of the KABR dataset, comparing YOLO-Behaviour with DeepEthogram.

Dataset	Framework	Weighted average precision	Weighted average recall	Weighted average false negative rate	Weighted average f1-score	Accuracy	Average inference speed (frames per second)
KABR	YOLO-Behaviour	0.77	**0.73**	**0.27**	**0.75**	**0.73**	**52.40**
	Deep Ethogram	0.77	0.66	0.34	0.71	0.65	35.9

Note: We refer to Table 3 for definitions of metrics. Average inference speed was calculated by running inference with each framework three times and averaging the processing speed in frames per second. The better-performing method for each metric is highlighted in bold.

TABLE 7 Evaluation metrics from the validation set, optimized by low false negative rates for event detection case studies.

Dataset	Average precision	Average recall	Average false negative rate	Average f1-score	Accuracy
Sparrow provisioning	0.78	0.87	0.13	0.82	0.71
Jay feeding	0.94	0.95	0.05	0.95	0.91
Human collective eating	0.26	0.86	0.14	0.40	0.94

Note: Metrics were optimized for low false negative rates instead of f1-score during the grid search algorithm, to test whether the framework can be used as a hybrid method.

Using the YOLO-behaviour framework, we contributed to efforts towards quantifying behavioural metrics across five diverse study systems. For example, in the Lundy house sparrow system, we obtained parental provisioning rates for more than double the sample size compared to manual annotated data (Chan, Liu, et al., 2024), allowing for stronger basis to discover the drivers and consequences of parental care behaviour, including indirect genetic effects (Schroeder et al., 2019) or its fitness outcomes (Schroeder et al., 2013). In Siberian jays, the method can be applied across the huge backlog of videos to gain insight into the co-feeding behaviour of this family living species where groups also can include unrelated non-breeders (Griesser, 2003; Griesser et al., 2015), to reveal the mechanisms facilitating the evolution of breeding systems and cooperation (Drobniak et al., 2015). In the human eating dataset, while the current dataset was obtained from a specific experiment, future data collection can benefit from the framework to reduce annotation time. Finally, while the 3D-POP and KABR datasets were used from computer vision datasets to demonstrate the ability for the framework to do frame-wise detections, the framework shows promise for larger scale deployment to quantify activity budgets of animals from drones (Koger et al., 2023; Schad & Fischer, 2023) or from citizen science data.

4.1 | Evaluation

The framework was first evaluated via the event validation set for each case study to determine its accuracy (in terms of precision) and its capacity to return manually coded events (in terms of recall). Overall, the framework performed well for all study systems with exceptions. First, the precision-recall for the human eating dataset was relatively low (around 0.6), which can be caused by hand gestures near the participant's mouth region, which is visually similar to the eating behaviour. To address this issue, additional training data might need to be added, or a hybrid human-in-the-loop method can be considered (see below). For the KABR dataset, we identified a few behaviours that the framework could not detect accurately. The first behaviour concerns locomotion, including run, trot and walk, which were difficult to distinguish without temporal context, since YOLO is a frame-wise method. For deployment of the model to detect these locomotion behaviours, a possible solution can be to add a speed threshold to separate the three behaviours, which cannot be tested in the current manuscript since the animal subjects in the KABR dataset were always centred in frame (Kholiavchenko et al., 2024). We note that auto-grooming and browsing behaviour was also not well detected using the YOLO-Behaviour framework (Figure 2e), which was probably caused by the large variation in postures/visual appearances of both behaviours in zebras and giraffes, making the model unable to generalize beyond the training data.

While precision recall values were high for the Siberian jay feeding dataset, we note that we chose a 2 s time window to match behaviours, as we observed a large mismatch between human annotation and automated detection. We also report results for different time window intervals (Table S3), and we show that all evaluation metrics improve up until the 2 s time window, which we assumed was the appropriate window. This mismatch can be caused by human reaction delay, especially when videos were coded in real time, but can also be due to incorrect detections by the model. However, upon visually inspecting qualitative results (Video S1), it seems more likely that there is a mismatch in frame-by-frame detections between automated and human manual annotations. When using the YOLO-Behaviour framework in future research, we emphasize that the time window selected during validation will significantly impact classification performance. Choosing an appropriate time window is important to avoid artificially inflating accuracy metrics. The mismatch between human and automated analyses will be an important issue to consider when evaluating future machine learning models for detecting behavioural events. Understanding the extent of this problem and exploring ways to address it will require further investigation as the field develops.

We also examined how the size of the training dataset size influences model accuracy and found that f1-scores, precision and recall consistently increased as training data size grew across case studies. The human eating dataset was the exception, with an increasing f1-score and precision, but decreasing recall. This indicates that as training dataset size increases, the model were able to detect eating more accurately albeit detecting less of the events. Across case studies, the analysis further suggests that model accuracy could potentially improve with additional training data, as evidenced by the steady rise in f1-score between proportions of 0.6–1.0, particularly in the KABR case study (Figure S1A). Interestingly, fine-tuning a MegaDetector model improved accuracy only for the smallest dataset, comprising 20% of the original data size. While fine-tuning a model pre-trained on animals would theoretically reduce the amount of data required to achieve comparable accuracy, our results indicate that the advancements in model architecture from YOLOv5 to YOLOv8 are the primary drivers of the observed improvements in prediction accuracy.

Next, we used the coded behavioural metrics dataset to test whether the YOLO-Behaviour framework can be used to extract behavioural metrics. Overall, we found high correlations and ICC values for both pigeon and zebra/giraffe frame-wise detections, but lower values for the other event-wise detection case studies. For human eating detection, this was expected due to the low evaluation metrics from the event-wise validation, and the low correlation can be the effect of misidentification of the eating behaviour itself. However, for the jay eating and sparrow provisioning datasets, we found low correlation albeit high precision-recall metrics from the event validation. For the jay dataset, this can be attributed to a mismatch between manual annotation of feeding behaviour in BORIS and the automatic method, and for the sparrow dataset, this can be caused by the many observers who annotated the dataset over the years that can result in inconsistent visit rates. We also note that there is a general under-detection of coded behavioural metrics extracted by YOLO-behaviour across datasets, meaning false negative rates (missed detections) were a stronger contributor to the low accuracy, which can potentially be improved with additional training data. Finally, we acknowledge that the correlation and ICC values might not be directly indicative of how well the YOLO-Behaviour framework can predict coded behavioural metrics, and additional evaluation like hypothesis testing (see Chan, Liu, et al., 2024) could be useful to further validate the method.

We then compared the performance and inference speed of YOLO-Behaviour with that of DeepEthogram (Bohnslav et al., 2021) on the KABR dataset. Overall, accuracies in terms of precision recall and f1-score were comparable between the two methods, though YOLO-Behaviour was consistently more accurate, and processed videos faster. The difference in accuracy was surprising, as DeepEthogram uses video sequences and optical flow input to parse temporal movement information, while YOLO only relies on single frames. However, comparing the confusion matrix of the DeepEthogram output (Figure S3), the predictions seem to be more consistent across classes compared to YOLO, so the accuracy values of YOLO-Behaviour might be inflated by good prediction of certain classes. We also note that we chose DeepEthogram-medium for its balance between accuracy and speed, though accuracy might be further improved with the different architecture choice. Nevertheless, the comparison was conducted with the aim of providing insight into how the methods relate to one another, rather than for comprehensive benchmarking, due to the unavoidable differences in training data. Due to the flexibility and uniqueness of the presented YOLO-Behaviour, most current open-source tools could not support the type of dataset presented here (multiple individuals, multiple behaviours), making further comparisons difficult. We suggest that researchers to carefully consider the type of video and behavioural data they have, before considering which open-source tool to select.

4.2 | Hybrid applications

In cases where automated detection accuracy might be low and insufficient for a certain study system, we also tested whether the YOLO-Behaviour framework can be used as a pre-processing step in a hybrid approach. Instead of optimizing for f1-score, we optimized for low false negative rates using grid search and found very low false negatives across all event detection datasets. For example, in the human eating dataset, we retrieved low false negative rates (0.14) and high recall (0.86) by trading off low precision (0.26). Hence, by first using YOLO-behaviour to extract events then manually confirm whether the detections were correct, we can potentially reach up to 0.86 precision in human eating detection, compared to the 0.63 precision when using the framework in a fully automated manner. While some manual annotation is still required, this hybrid approach would further reduce annotation time with the assurance that extracted behavioural events are accurate. In the provided code, we also provide example code to run the pipeline with a human–in-the-loop approach.

4.3 | Limitations

The YOLOv8 model used in the current framework only takes a single frame as input, which might not be able to reliably detect behaviours that have a temporal aspect, like the locomotion behaviours in KABR. Unlike other methods, the framework does not capture fine-scaled kinematics that might require posture estimation, nor complex behavioural sequences like courtship behaviour (Janisch et al., 2021). However, we do note that the current framework was still able to detect walking and bowing behaviours in the pigeon dataset reliably, likely due to other visual cues (e.g. leg up, puffed up neck). For behaviours that have an important temporal component, other methods that take video input (Bohnslav et al., 2021; Rodríguez-Moreno et al., 2019), or first do posture estimation (Mathis & Mathis, 2020) might be considered. Still, we note that compared to the 86.7% per-instance accuracy reported in the original KABR publication (Kholiavchenko et al., 2024) using a temporal based X3D method, the current YOLO-Behaviour framework still managed to recover similar accuracies (72%, Table 3), Whether this difference in accuracy is important for quantifying behaviours will depend on the specific use case.

Finally, as with any deep learning model, the need for training data must be carefully considered. While the case studies presented in this paper utilized minimal training data (~1000–1500 images), the framework may perform less effectively when the behaviour of interest is rare and represented by only a few instances. In such cases, alternative approaches like few-shot object detection models (Köhler et al., 2023) maybe more appropriate. However, compared to methods that first do posture estimation and then behavioural recognition, our current framework significantly reduces the amount of required training data. This is achieved by relying solely on the behavioural annotations within a frame, rather than detailed posture annotations and time-series annotations of behavioural sequences (e.g. Hu et al., 2023; Wittek et al., 2022).

We also emphasize that, in practice, any object detector model can also be used in the same way to quantify behaviour, but we chose YOLO here because of the ease of use and robustness. Moreover, the annotation effort required with our approach is considerably lower than traditional manual annotations methods, especially for long-term datasets like the sparrow provisioning or Siberian jay eating case studies. For example, previous research demonstrated that each 90-min sparrow provisioning video takes on average 65.4 min to manually annotate (Chan, Liu, et al., 2024). Given a backlog of ~2000 videos that have not been analysed, this would have taken 2180 annotation hours, or 54.5 forty-hour work weeks, or around a year of full-time labour to annotate, making manual annotation impractical. In contrast, assuming it takes 10 s to annotate a single frame with a bounding box, annotating 1500 images used in this study would take approximately 4 h. This highlights the substantial time efficiency of YOLO-Behaviour for the coding of events in large streams of behavioural video data.

5 | CONCLUSIONS

In conclusion, we presented the YOLO-Behaviour Framework, a simple, flexible and robust method for automating video classification of simple behaviours. We demonstrated the efficiency of the pipeline in five distinct case studies and highlighted that the framework works well across a wide range of behaviours and videos. With the increased use of deep learning and machine learning for measuring behaviours in animals, we hope the framework can be another step towards lowering the barrier to train and deploy these methods and replacing time-consuming manual annotation in the field of behaviour research.

AUTHOR CONTRIBUTIONS

Alex Hoi Hang Chan, Prasetia Putra and Harald Schupp conceived the ideas and designed methodology; Harald Schupp, Johanna Köchling, Jana Straßheim, Britta Renner, Julia Schroeder, William D. Pearse, Shinichi Nakagawa, Terry Burke, Michael Griesser, Andrea Meltzer and Saverio Lubrano collected the data; Alex Hoi Hang Chan analysed the data; Alex Hoi Hang Chan led the writing of the manuscript. Fumihiro Kano supervised the project. All authors contributed critically to the drafts and gave final approval for publication. Our study introduces a new method for automated behavioural annotation and was tested across a wide range of species and study systems from different countries and field sites around the world, and the co-authors of the paper also represents a large range of institutions and fields, to ensure the work has diverse perspectives and ideas.

ACKNOWLEDGEMENTS

This work is funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany's Excellence Strategy—EXC 2117—422037984. We additionally thank Francesca Frisoni for additional manual annotation, and all students/field assistants who collected and manually annotated the datasets over the years. I also thank Stephen Tyndel for sharing my enthusiasm on the wonders of YOLO and commenting on an initial draft of the manuscript. MG was supported by a Heisenberg Grant no. GR 4650/2-1 by the German Research Foundation DFG. Open Access funding enabled and organized by Projekt DEAL.

CONFLICT OF INTEREST STATEMENT

William D. Pearse is an associate editor of *Methods in Ecology and Evolution*, but took no part in the peer review and decision-making processes for this paper.

PEER REVIEW

The peer review history for this article is available at https://www.webofscience.com/api/gateway/wos/peer-review/10.1111/2041-210X.14502.

DATA AVAILABILITY STATEMENT

Data and code to reproduce analysis and evaluation in the current manuscript can be accessed via https://doi.org/10.5281/zenodo.14639346 (Chan, 2025). Code and documentation for implementing the pipeline can be found in the following repository: https://github.com/alexhang212/YOLO_Behaviour_Repo. Additional datasets for sample code and model weights can be downloaded via: https://doi.org/10.17617/3.EZNKYV (Chan, Putra, et al., 2024).

ORCID

Alex Hoi Hang Chan https://orcid.org/0000-0002-5405-7155
Prasetia Putra https://orcid.org/0000-0002-7632-375X
Harald Schupp https://orcid.org/0000-0002-1725-9129
Johanna Köchling https://orcid.org/0000-0003-1842-797X
Jana Straßheim https://orcid.org/0009-0002-5393-7182
Britta Renner https://orcid.org/0000-0001-8385-2839
Julia Schroeder https://orcid.org/0000-0002-4136-843X
William D. Pearse https://orcid.org/0000-0002-6241-3164
Shinichi Nakagawa https://orcid.org/0000-0002-7765-5182

Terry Burke https://orcid.org/0000-0003-3848-1244
Michael Griesser https://orcid.org/0000-0002-2220-2637
Andrea Meltzer https://orcid.org/0000-0003-4550-9620
Saverio Lubrano https://orcid.org/0009-0002-0963-5419
Fumihiro Kano https://orcid.org/0000-0003-4534-6630

REFERENCES

Beery, S., Morris, D., & Yang, S. (2019). Efficient pipeline for camera trap image review. *arXiv*, https://doi.org/10.48550/arXiv.1907.06772

Bewley, A., Ge, Z., Ott, L., Ramos, F., & Upcroft, B. (2016). Simple online and realtime tracking. In *2016 IEEE international conference on image processing (ICIP)* (pp. 3464–3468). ICIP. https://doi.org/10.1109/ICIP.2016.7533003

Bobak, C. A., Barr, P. J., & O'Malley, A. J. (2018). Estimation of an inter-rater intra-class correlation coefficient that overcomes common assumption violations in the assessment of health measurement scales. *BMC Medical Research Methodology*, *18*, 93. https://doi.org/10.1186/s12874-018-0550-6

Bohnslav, J. P., Wimalasena, N. K., Clausing, K. J., Dai, Y. Y., Yarmolinsky, D. A., Cruz, T., Kashlan, A. D., Chiappe, M. E., Orefice, L. L., Woolf, C. J., & Harvey, C. D. (2021). Deepethogram, a machine learning pipeline for supervised behavior classification from raw pixels. *eLife*, *10*, e63377. https://doi.org/10.7554/eLife.63377

Brookes, O., Mirmehdi, M., Stephens, C., Angedakin, S., Corogenes, K., Dowd, D., Dieguez, P., Hicks, T. C., Jones, S., Lee, K., Leinert, V., Lapuente, J., McCarthy, M. S., Meier, A., Murai, M., Normand, E., Vergnes, V., Wessling, E. G., Wittig, R. M., … Burghardt, T. (2024). PanAf20K: A large video dataset for wild ape detection and behaviour recognition. *International Journal of Computer Vision*, *132*, 3086–3102. https://doi.org/10.1007/s11263-024-02003-z

Chan, H. H. (2025). Code to reproduce evaluation from: YOLO-Behaviour: A simple, flexible framework to automatically quantify animal behaviours from videos. *Zenodo*. https://doi.org/10.5281/zenodo.14639346

Chan, A. H. H., Liu, J., Burke, T., Pearse, W. D., & Schroeder, J. (2024). Comparison of manual, machine learning, and hybrid methods for video annotation to extract parental care data. *Journal of Avian Biology*, *2024*, e03167. https://doi.org/10.1111/jav.03167

Chan, H. H., Putra, P., Schupp, H., Köchling, J., Straßheim, J., Renner, B., Schroeder, J., Pearse, W. D., Nakagawa, S., Burke, T., Griesser, M., Meltzer, A., Lubrano, S., & Kano, F. (2024). Sample dataset for YOLO-Behaviour: A simple, flexible framework to automatically quantify animal behaviours from videos. https://doi.org/10.17617/3.EZNKYV

Chen, J., Hu, M., Coker, D. J., Berumen, M. L., Costelloe, B., Beery, S., Rohrbach, A., & Elhoseiny, M. (2023). MammalNet: A large-scale video benchmark for mammal recognition and behavior understanding. *Presented at the Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13052–13061.

Chimento, M., Chan, A. H. H., Aplin, L. M., & Kano, F. (2024). Peering into the world of wild passerines with 3d-SOCS: Synchronized video capture and posture estimation. *bioRxiv* 2024.06.30.601375. https://doi.org/10.1101/2024.06.30.601375

Christin, S., Hervet, É., & Lecomte, N. (2019). Applications for deep learning in ecology. *Methods in Ecology and Evolution*, *10*, 1632–1644.

Couzin, I. D., & Heins, C. (2022). Emerging technologies for behavioral research in changing environments. *Trends in Ecology & Evolution*, *38*, 346–354. https://doi.org/10.1016/j.tree.2022.11.008

Dell, A. I., Bender, J. A., Branson, K., Couzin, I. D., de Polavieja, G. G., Noldus, L. P., Pérez-Escudero, A., Perona, P., Straw, A. D., & Wikelski, M. (2014). Automated image-based tracking and its application in ecology. *Trends in Ecology & Evolution*, *29*, 417–428.

Drobniak, S. M., Wagner, G., Mourocq, E., & Griesser, M. (2015). Family living: An overlooked but pivotal social system to understand the evolution of cooperative breeding. *Behavioral Ecology*, *26*, 805–811. https://doi.org/10.1093/beheco/arv015

Friard, O., & Gamba, M. (2016). BORIS: A free, versatile open-source event-logging software for video/audio coding and live observations. *Methods in Ecology and Evolution*, *7*, 1325–1330. https://doi.org/10.1111/2041-210X.12584

Goodwin, N. L., Choong, J. J., Hwang, S., Pitts, K., Bloom, L., Islam, A., Zhang, Y. Y., Szelenyi, E. R., Tong, X., Newman, E. L., Miczek, K., Wright, H. R., McLaughlin, R. J., Norville, Z. C., Eshel, N., Heshmati, M., Nilsson, S. R. O., & Golden, S. A. (2024). Simple behavioral analysis (SimBA) as a platform for explainable machine learning in behavioral neuroscience. *Nature Neuroscience*, *27*, 1411–1424. https://doi.org/10.1038/s41593-024-01649-9

Goss, K., Bueno-Junior, L. S., Stangis, K., Ardoin, T., Carmon, H., Zhou, J., Satapathy, R., Baker, I., Jones-Tinsley, C. E., Lim, M. M., Watson, B. O., Sueur, C., Ferrario, C. R., Murphy, G. G., Ye, B., & Hu, Y. (2024). Quantifying social roles in multi-animal videos using subject-aware deep-learning. *BioRxiv* 2024.07.07.602350. https://doi.org/10.1101/2024.07.07.602350

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

Graving, J. M., & Couzin, I. D. (2020). *Vae-sne: A deep generative model for simultaneous dimensionality reduction and clustering*. https://doi.org/10.1101/2020.07.17.207993

Griesser, M. (2003). Nepotistic vigilance behavior in Siberian jay parents. *Behavioral Ecology*, *14*, 246–250. https://doi.org/10.1093/beheco/14.2.246

Griesser, M., Halvarsson, P., Drobniak, S. M., & Vilà, C. (2015). Fine-scale kin recognition in the absence of social familiarity in the Siberian jay, a monogamous bird species. *Molecular Ecology*, *24*, 5726–5738. https://doi.org/10.1111/mec.13420

Gwet, K. L. (2014). *Handbook of inter-rater reliability, 4th edition: The definitive guide to measuring the extent of agreement among raters*. Advanced Analytics, LLC.

Harris, C., Finn, K. R., Kieseler, M.-L., Maechler, M. R., & Tse, P. U. (2023). DeepAction: A MATLAB toolbox for automated classification of animal behavior in video. *Scientific Reports*, *13*, 2688. https://doi.org/10.1038/s41598-023-29574-0

Hsu, A. I., & Yttri, E. A. (2021). B-SOiD, an open-source unsupervised algorithm for identification and fast prediction of behaviors. *Nature Communications*, *12*, 5188. https://doi.org/10.1038/s41467-021-25420-x

Hu, Y., Ferrario, C. R., Maitland, A. D., Ionides, R. B., Ghimire, A., Watson, B., Iwasaki, K., White, H., Xi, Y., Zhou, J., & Ye, B. (2023). LabGym: Quantification of user-defined animal behaviors using learning-based holistic assessment. *Cell Reports Methods*, *3*, 100415. https://doi.org/10.1016/j.crmeth.2023.100415

Janisch, J., Perinot, E., Fusani, L., & Quigley, C. (2021). Deciphering choreographies of elaborate courtship displays of golden-collared manakins using markerless motion capture. *Ethology*, *127*, 550–562. https://doi.org/10.1111/eth.13161

Jocher, G., Chaurasia, A., & Qiu, J. (2023). *Yolo by ultralytics*.

Joska, D., Clark, L., Muramatsu, N., Jericevich, R., Nicolls, F., Mathis, A., Mathis, M. W., & Patel, A. (2021). AcinoSet: A 3D pose estimation dataset and baseline models for cheetahs in the wild. In *2021 IEEE international conference on robotics and automation (ICRA)* (pp. 13901–13908). ICRA. https://doi.org/10.1109/ICRA48506.2021.9561338

Kholiavchenko, M., Kline, J., Ramirez, M., Stevens, S., Sheets, A., Babu, R., Banerji, N., Campolongo, E., Thompson, M., Van Tiel, N., Miliko, J., Bessa, E., Duporge, I., Berger-Wolf, T., Rubenstein, D., & Stewart, C. (2024). kabr: In-situ dataset for kenyan animal behavior recognition from drone videos. *Presented at the 2024 IEEE/CVF Winter Conference on Applications of Computer Vision Workshops (WACVW)*, IEEE, Waikoloa, HI, USA, pp. 31–40. https://doi.org/10.1109/WACVW60836.2024.00011

Koger, B., Deshpande, A., Kerby, J. T., Graving, J. M., Costelloe, B. R., & Couzin, I. D. (2023). Quantifying the movement, behaviour and environmental context of group-living animals using drones and computer vision. *The Journal of Animal Ecology*, *92*, 1357–1371. https://doi.org/10.1111/1365-2656.13904

Köhler, M., Eisenbach, M., & Gross, H.-M. (2023). Few-shot object detection: A comprehensive survey. *IEEE Transactions on Neural Networks and Learning Systems*, *35*(9), 1–21. https://doi.org/10.1109/TNNLS.2023.3265051

Koo, T. K., & Li, M. Y. (2016). A guideline of selecting and reporting intraclass correlation coefficients for reliability research. *Journal of Chiropractic Medicine*, *15*, 155–163. https://doi.org/10.1016/j.jcm.2016.02.012

Lauer, J., Zhou, M., Ye, S., Menegas, W., Schneider, S., Nath, T., Rahman, M. M., Di Santo, V., Soberanes, D., Feng, G., Murthy, V. N., Lauder, G., Dulac, C., Mathis, M. W., & Mathis, A. (2022). Multi-animal pose estimation, identification and tracking with DeepLabCut. *Nature Methods*, *19*, 496–504. https://doi.org/10.1038/s41592-022-01443-0

Lei, Y., Dong, P., Guan, Y., Xiang, Y., Xie, M., Mu, J., Wang, Y., & Ni, Q. (2022). Postural behavior recognition of captive nocturnal animals based on deep learning: A case study of Bengal slow loris. *Scientific Reports*, *12*, 7738. https://doi.org/10.1038/s41598-022-11842-0

Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., & Zitnick, C. L. (2014). Microsoft coco: Common objects in context. In *European conference on computer vision* (pp. 740–755). Springer.

Mathis, A., Mamidanna, P., Cury, K. M., Abe, T., Murthy, V. N., Mathis, M. W., & Bethge, M. (2018). DeepLabCut: Markerless pose estimation of user-defined body parts with deep learning. *Nature Neuroscience*, *21*(9), 1281–1289. https://doi.org/10.1038/s41593-018-0209-y

Mathis, M. W., & Mathis, A. (2020). Deep learning tools for the measurement of animal behavior in neuroscience. *Current Opinion in Neurobiology*, *60*, 1–11.

Naik, H., Chan, A. H. H., Yang, J., Delacoux, M., Couzin, I. D., Kano, F., & Nagy, M. (2023). 3d-pop - an automated annotation approach to facilitate markerless 2d-3d tracking of freely moving birds with marker-based motion capture. *Presented at the Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 21274–21284.

Ng, X. L., Ong, K. E., Zheng, Q., Ni, Y., Yeo, S. Y., & Liu, J. (2022). Animal kingdom: A large and diverse dataset for animal behavior understanding. *Presented at the Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 19023–19034.

Pereira, T. D., Tabris, N., Matsliah, A., Turner, D. M., Li, J., Ravindranath, S., Papadoyannis, E. S., Normand, E., Deutsch, D. S., Wang, Z. Y., McKenzie-Smith, G. C., Mitelut, C. C., Castro, M. D., D'Uva, J., Kislin, M., Sanes, D. H., Kocher, S. D., Wang, S. S.-H., Falkner, A. L., … Murthy, M. (2022). Sleap: A deep learning system for multi-animal pose tracking. *Nature Methods*, *19*, 486–495. https://doi.org/10.1038/s41592-022-01426-1

Rodríguez-Moreno, I., Martínez-Otzeta, J. M., Sierra, B., Rodriguez, I., & Jauregi, E. (2019). Video activity recognition: State-of-the-art. *Sensors*, *19*(14), 3160. https://doi.org/10.3390/s19143160

Schad, L., & Fischer, J. (2023). Opportunities and risks in the use of drones for studying animal behaviour. *Methods in Ecology and Evolution*, *14*, 1864–1872. https://doi.org/10.1111/2041-210X.13922

Schroeder, J., Cleasby, I., Dugdale, H. L., Nakagawa, S., & Burke, T. (2013). Social and genetic benefits of parental investment suggest sex differences in selection pressures. *Journal of Avian Biology*, *44*, 133–140.

Schroeder, J., Dugdale, H., Nakagawa, S., Sparks, A., & Burke, T. (2019). *Social genetic effects (IGE) and genetic intra-and intersexual genetic correlation contribute to the total heritable variance in parental care*. https://doi.org/10.32942/osf.io/nh8m2

Tuia, D., Kellenberger, B., Beery, S., Costelloe, B. R., Zuffi, S., Risse, B., Mathis, A., Mathis, M. W., van Langevelde, F., & Burghardt, T. (2022). Perspectives in machine learning for wildlife conservation. *Nature Communications*, *13*, 1–15.

Tuyttens, F. A. M., de Graaf, S., Heerkens, J. L., Jacobs, L., Nalon, E., Ott, S., Stadig, L., Van Laer, E., & Ampe, B. (2014). Observer bias in animal behaviour research: Can we believe what we score, if we score what we believe? *Animal Behaviour*, *90*, 273–280.

Waldmann, U., Chan, A. H. H., Naik, H., Nagy, M., Couzin, I. D., Deussen, O., Goldluecke, B., & Kano, F. (2024). 3D-MuPPET: 3D multi-pigeon pose estimation and tracking. *International Journal of Computer Vision*, *132*, 4235–4252. https://doi.org/10.1007/s11263-024-02074-y

Weinstein, B. G. (2018). A computer vision for animal ecology. *The Journal of Animal Ecology*, *87*, 533–545.

Wiltshire, C., Lewis-Cheetham, J., Komedová, V., Matsuzawa, T., Graham, K. E., & Hobaiter, C. (2023). DeepWild: Application of the pose estimation tool DeepLabCut for behaviour tracking in wild chimpanzees and bonobos. *The Journal of Animal Ecology*, *92*, 1560–1574. https://doi.org/10.1111/1365-2656.13932

Wittek, N., Wittek, K., Keibel, C., & Güntürkün, O. (2022). Supervised machine learning aided behavior classification in pigeons. *Behavior Research Methods*, *55*(4), 1624–1640. https://doi.org/10.3758/s13428-022-01881-w

Yang, A., Huang, H., Yang, X., Li, S., Chen, C., Gan, H., & Xue, Y. (2019). Automated video analysis of sow nursing behavior based on fully convolutional network and oriented optical flow. *Computers and Electronics in Agriculture*, *167*, 105048.

SUPPORTING INFORMATION

Additional supporting information can be found online in the Supporting Information section at the end of this article.

Figure S1: Change in weighted average f1-score with different proportion subset of the training dataset.

Figure S2: Change in weighted average precision and recall with different proportion subset of the training dataset.

Figure S3: Confusion matrix for behavioural classification in the KABR dataset using DeepEthogram.

Table S1: YOLO evaluation metrics from the training data validation set.

Table S2: Final hyper parameters used for evaluation after optimizing for the highest f1-score from the grid search algorithm.

Table S3: Final hyper parameters used for evaluation after optimizing for the lowest false negative rates from the grid search algorithm, for event detection case studies.

Table S4: Evaluation metrics for different time window choice for the Siberian jay dataset.

Video S1: Qualitative video results for YOLO-Behaviour.

How to cite this article: Chan, A. H. H., Putra, P., Schupp, H., Köchling, J., Straßheim, J., Renner, B., Schroeder, J., Pearse, W. D., Nakagawa, S., Burke, T., Griesser, M., Meltzer, A., Lubrano, S., & Kano, F. (2025). YOLO-Behaviour: A simple, flexible framework to automatically quantify animal behaviours from videos. *Methods in Ecology and Evolution*, *00*, 1–15. https://doi.org/10.1111/2041-210X.14502