



OPEN

DATA DESCRIPTOR

Identifying Cocoa Flower Visitors: A Deep Learning Dataset

Wenxiu Xu^{1,2,7}, Saba Ghorbani Barzegar^{2,7}, Dong Sheng^{1,2}, Manuel Toledo-Hernández^{2,3}, ZhenZhong Lan⁴ & Thomas Cherico Wanger^{2,5,6}

Cocoa is a multi-billion-dollar industry but research on improving yields through pollination remains limited. New embedded hardware and AI-based data analysis is advancing information on cocoa flower visitors, their identity and implications for yields. We present the first cocoa flower visitor dataset containing 5,792 images of Ceratopogonidae, Formicidae, Aphididae, Araneae, and Encyrtidae, and 1,082 background cocoa flower images. This dataset was curated from 23 million images collected over two years by embedded cameras in cocoa plantations in Hainan province, China. We exemplify the use of the dataset with different sizes of YOLOv8 models and by progressively increasing the background image ratio in the training set to identify the best-performing model. The medium-sized YOLOv8 model achieved the best results with 8% background images (F1 Score of 0.71, mAP50 of 0.70). Overall, this dataset is useful to compare the performance of deep learning model architectures on images with low contrast images and difficult detection targets. The data can support future efforts to advance sustainable cocoa production through pollination monitoring projects.

Background & Summary

Agricultural production has been instrumental in providing food, fiber and energy for billions of people, but environmental, health, and economic costs have rendered global farming systems a net loss system of USD 1.9 trillion annually¹. A global food systems transformation becomes the inevitable pathway towards an agricultural future that benefits both people and planet^{2,3}. Therein, biodiversity in agricultural landscapes and the associated ecosystem services such as biological pest control and pollination are critical to maintain ecosystem functioning and human wellbeing⁴. In contrast, biodiversity is also threatened by climate change, land use and pollution caused by agricultural production⁵. Effective biodiversity monitoring is, therefore, critical for global conservation efforts and to maintain ecosystem services⁶ and greatly enabled through new technologies. For instance, passive acoustic monitoring and global efforts to aggregate existing information into one place can help biodiversity conservation across ecosystems⁷. Insect ecology is advancing for instance through LiDAR monitoring, DNA metabarcoding and visual methods, often coupled with machine learning approaches⁸. In general, these technologies are transforming our understanding of spatial, temporal, and taxonomic aspects of biodiversity monitoring, while it is important to overcome technical limitations and subsequent access to data bound by international standards.

Throughout the history of biodiversity monitoring, visual methods have played a pioneering role that is now advanced by AI-based methods for embedded systems. Traditional camera traps have enabled continuous documentation of animal activity during day and night. Current AI-based cameras - in particular when integrated as embedded IoT networks - can cover larger areas and collect data that can provide insights into movement patterns, species interactions, demographic trends, and behavioral patterns. Such camera systems use object detection to generate large datasets for further individual, population, and network level analyses (for laboratory studies see⁹). The general process to train object detection models is done based on established datasets or newly sourced data when specific datasets are unavailable for instance for different target organisms, lighting, or stylistic conditions. Then, Generative Adversarial Networks (GAN) can be used to generate additional data (e.g.,

¹College of Environmental and Resource Sciences, Zhejiang University, Hangzhou, China. ²Sustainable Agricultural Systems & Engineering Laboratory, School of Engineering, Westlake University, Hangzhou, China. ³Sustainable Development Department, Instituto Tecnológico Vale, Belém, Brazil. ⁴School of Engineering, Westlake University, Hangzhou, China. ⁵Key Laboratory of Coastal Environment and Resources of Zhejiang Province, Westlake University, Hangzhou, China. ⁶Production Technology & Cropping Systems Group, Department of Plant Production, AgroScope, Nyon, Switzerland. ⁷These authors contributed equally: Wenxiu Xu, Saba Ghorbani Barzegar. ⁸e-mail: tomcwanger@gmail.com

CycleGAN can produce synthetic data¹⁰) or new empirical data is collected (for an example image training size effects on classification of European bee species, see Wanger and Frohn¹¹). Building new training datasets come with their own challenges. First, depending on the abundance of the target organism, obtaining enough data can be challenging. Second, the background against which the animal is captured can make the images complex and difficult to analyze^{12,13}. Lastly, labelling of newly built datasets is time-consuming and expensive but can be overcome through preprocessing techniques to reduce noise in images¹⁴.

An area where advancements in biodiversity monitoring and object detection has not been implemented is yield-determining pollination in cocoa production. Cocoa is the crop that is needed to produce chocolate, a multi-billion USD industry. It grows in the tropical regions of the world, and the crop is pollination limited¹⁵. When the small cocoa flowers are not pollinated, no cocoa pods emerge, and no cocoa beans can eventually be harvested leaving the cocoa industry unable to satisfy the growing global demand for chocolate. While yield benefits of manual pollination range from 200–800% in Indonesia and Brazil^{16,17}, little is known about natural flower visitors such as midges and flies (Dipterans), thrips (Thysanoptera), and ants and parasitoid wasps (Hymenoptera). This information, however, is critical for an effective management of cocoa plantations for pollinator conservation and to reduced environmental impacts from climate change^{15,18}. Current methods for cocoa pollinator monitoring include invasive and indirect methods such as pan traps or glue that capture flower visitors but do not allow linking flower visitors to cocoa yields. Moreover, visual encounter surveys on cocoa flowers are challenging, because only 16% of all cocoa flowers are successfully pollinated and visitors spend potentially a very short time on the flowers¹⁹. Automated monitoring methods that can classify flower visitors are, hence, desirable because they allow linking pollination effectiveness with yields. It is, however, extremely difficult to compile the location specific datasets with flower visitors to train relevant models, because (i) a low visiting frequency leads to large amounts of images without flower visitors; and (ii) 24 h recording requires infrared illumination for night monitoring and results in greyscale images with little contrast and sometimes only fractions of rare flower visitors.

Here, we present the first cocoa flower visitor dataset consisting of 5,792 insect images and 1,082 flower ‘background’ images. Of these, 5,214 insect images and 782 background images were collected in 2023, featuring five common cocoa flower visitors: Ceratopogonidae (midges), Formicidae (ants), Aphididae (aphids), Araneae (spiders), and Encyrtidae (parasitoid wasps). In 2024, 578 insect images and 300 background images were collected, featuring three common cocoa flower visitors: Ceratopogonidae, Formicidae, and Encyrtidae. The dataset was curated from 23 million images collected over two years by embedded cameras²⁰ deployed in cocoa plantations at the Xinglong Tropical Botanical Garden, Hainan Province, China. We then use a YOLOv8 (You Only Look Once) algorithm for object detection, predicting bounding boxes and assigning class probabilities for multiple objects in flower visitor classes²¹.

Methods

Data collection. The data was collected in Xinglong Botanical Garden (18° 43′ 57.6″ N, 110° 11′ 55.8″ E), Hainan Province, China, where we monitored cocoa flowers with embedded computer vision cameras for at least 24 hours. In total, we monitored 741 flowers from April to September 2023, and 417 flowers from April to July 2024. The cameras used frame differencing and blob detection to detect activity on the flowers and then automatically stored detection events on on-board SD cards. This approach resulted on average in 20,000 images per flower with and without visitor detection.

We obtained a total of 23 million images in JPG format of a $1,944 \times 1,944$ pixel resolution. We checked 8,040,000 images manually for flower visitors before using the trained YOLO object detection model. We employed a screening process whereby manual screening was followed up by model training and testing. We used a subset of the data to make a screening model (256 images), tested the model on new data and thereby incrementally increased the training data and performance (Fig. 1A). We stopped manual screening when the model had reached 90% accuracy to save time and financial resources. Subsequently, we used the optimized model to screen for images containing insects from all the data collected in 2023. In total, we obtained 5,214 images contained flower visitors in five groups Ceratopogonidae, Formicidae, Aphididae, Araneae, and Encyrtidae from the 2023 data (Fig. 2).

Object detection models. Several models have galvanized as important in object detection. One such influential model is Faster R-CNN (Region-based Convolutional Neural Network)²², which generates candidate object regions and a subsequent object detection network to classify and refine these regions. Nieuwenhuizen *et al.*²³ detected tomato whitefly and its predatory bugs on yellow sticky traps by a faster R-CNN model. Du *et al.*²⁴ used ResNet50 and online hard example mining to improve faster R-CNN models and detected multiple insect types in field images. The Single Shot MultiBox Detector (SSD)²⁵ is a popular single-stage object detection model that achieves high efficiency by simultaneously predicting object classes and bounding box coordinates at different scales using a series of convolutional layers. Lyu *et al.*²⁶ used an optimized SSD feature fusion algorithm to detect pests among grains. And Garcia *et al.*²⁷ used SSD on a microcontroller to detect and count insects such as whiteflies and aphids on eggplant leaves. Additionally, models like YOLO (You Only Look Once)²⁸ excel at real-time object detection, making them highly suitable for applications on embedded devices. Ratnayake *et al.*²⁹ used YOLOv4 and KNN segmentation methods to count insect visitations on a particular flower. Kumar *et al.*³⁰ introduced channel and spatial attention modules to a YOLOv5 model and detected 23 categories of insects.

We use the YOLOv8 object detection algorithm that processes input images, generates bounding boxes with corresponding class probabilities, indicating object locations and likelihoods of belonging to specific classes. YOLOv8 architecture comprises convolutional layers, spatial pyramid pooling, and Path Aggregation Network modules, enabling effective feature extraction and aggregation for accurate object detection across diverse sizes (model architecture is discussed in detail elsewhere²¹). We used the YOLOv8 model for both model creation

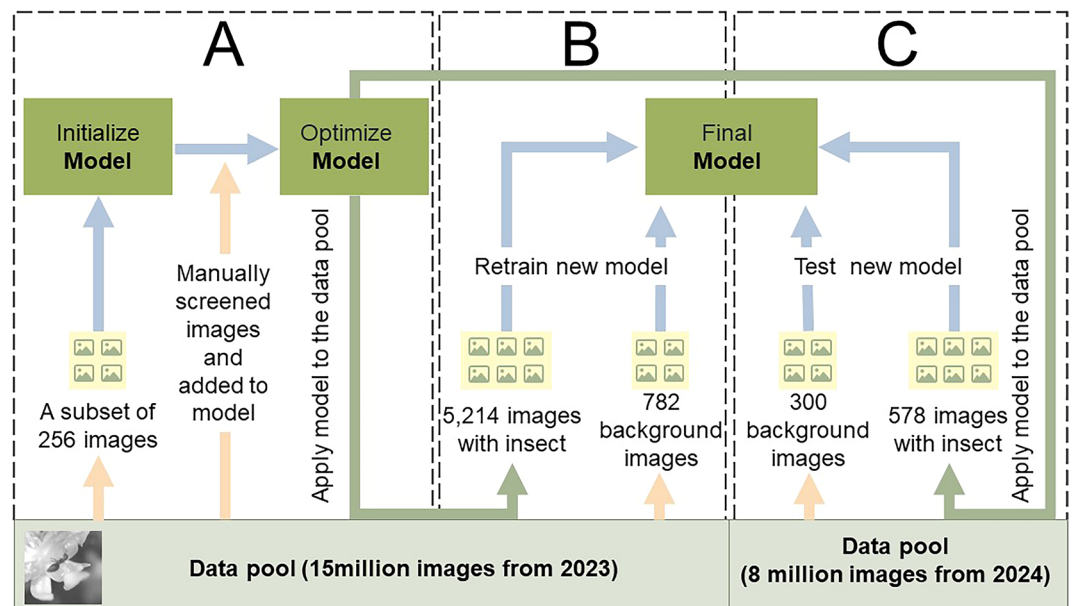


Fig. 1 Data collection and model training process. Initial training used a subset of 256 images, followed by manual screening to optimize the model, achieving 90% accuracy (A). The optimized model was then applied to the 2023 data pool, identifying 5,214 insect images. From these, 782 unique background images were manually selected for retraining (B). In 2024, the optimized model screened 578 insect images from the data pool, with 300 unique background images manually selected to test the final model (C).



Fig. 2 Example images for the five flower visitor groups in our dataset. Ceratopogonidae (A), Formicidae (B), Aphididae (C), Araneae (D), and Encyrtidae (E). Panel F shows the cocoa farms in Hainan, China (F; © by Manuel Toledo-Hernández).

and prediction. The YOLOv8 model consists of various weights, each with a different number of parameters. To ensure efficiency, we incorporated all these weights and compared their performance to determine which one worked most effectively.

Dataset annotation. Annotating data is an important step prior to model training. It involves placing a bounding box around objects and assigning them a class for classification. For the data from 2023, we ran a preliminary YOLOv8 model on the dataset to automatically annotate objects. We then reviewed the annotations manually to delete false negatives and correcting any inaccuracies in the bounding boxes. For the data from 2024, we manually annotated it using Label-studio, and after completing the annotations, we performed a double check

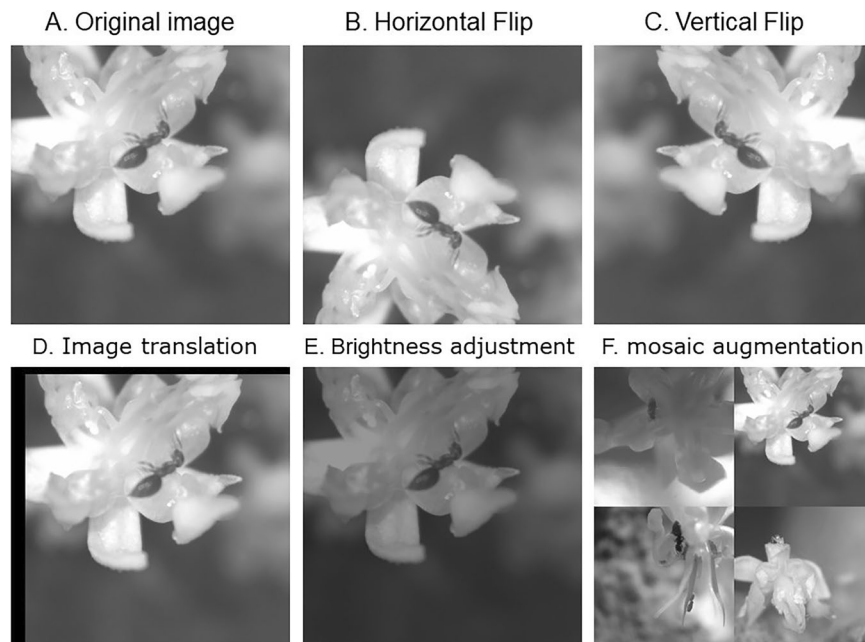


Fig. 3 Examples of the applied image augmentation.

to ensure their accuracy. After completing the annotation of all the data, the 5,792 images contained a total of 6,027 bounding boxes, with 2,056 for Ceratopogonidae, 3,003 for Formicidae, 628 for Aphididae, 176 for Araneae, and 164 for Encyrtidae.

Dataset augmentation. A general solution to limited training data that is labelled is image augmentation, whereby the images are transformed in shape and size. Augmentation techniques are dynamically applied in the training process of the model and include HSV random transformation (i.e., hue - the color tone; saturation - the intensity of color; and value - brightness of the color are modified at random). Image translation along the x & y axis introduces a position shift without structural changes and aims to train the model on different perspectives and spatial contexts. Horizontal flipping swaps pixels from one side to the other, scaling does not change the aspect ratio but object size and orientation in the frame. Lastly, mosaic augmentation uses four source images and - while preserving the aspect ratio - compiles them into a new image³¹. The original image sample size can be increased several folds, thereby enhancing generalizability of the model.

We performed horizontal and vertical flipping, image translation, and mosaic augmentation on images containing cocoa visitors. Additionally, we adjusted the brightness (V channel) in the HSV color space in our grey-scale images (e.g. Fig. 3). Through training data augmentation, we were able to expand our dataset 5-fold.

Experimental setup. The training and testing of the deep learning models were done on a workstation with the following specifications: the central processing unit (CPU) is an Intel(R) Xeon(R) W-2235 with a memory capacity of 64GB, the graphics processing unit (GPU) is an NVIDIA Quadro RTX 4000 with 40 GB of memory; the operating system is Windows 10 Pro; the PyTorch version is 2.0.0, and the CUDA version is 11.7.

Evaluation metrics. We evaluated the detection performance of the model with the standard metrics precision (P), recall (R), F1 score, mAP50, mAP50-95 and false positive rate (FPR). The formulas for these evaluation indicators are as follows:

$$P = \frac{TP}{TP + FP}$$

$$R = \frac{TP}{TP + FN}$$

$$F1 \text{ Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

The evaluation of insect detection is based on confidence scores. A confidence score of at least 0.5 is required to classify a flower visitor as a true positive (TP). Incorrectly identifying objects such as flower or background as a flower visitor is considered a false positive (FP). Failing to detect a flower visitor or incorrectly classifying it as a different category is considered a false negative (FN). True negatives (TN) are recorded when there is no flower visitor in the image.

Precision is the ratio of true positives to total detections made by the model, while Recall measures the proportion of true positives to the total actual objects. Additionally, we introduce the F1 score as a harmonic measure to comprehensively evaluate the model's precision and recall.

$$\text{mAP} = \frac{1}{k} \sum_{i=1}^k \text{AP}_i$$

Mean Average Precision (mAP) is a key evaluation metric used to assess the performance of object detection networks, as it takes both precision and recall into account. The mAP is the mean of the Average Precision (AP) values obtained at various recall levels from the Precision-Recall (PR) curve. The specific mAP50 and mAP50-95 measures the precision at an Intersection over Union (IoU) threshold of 0.50 and ranging thresholds between 0.50–0.95, respectively. The mAP50 is a measure of accuracy for 'easy' detection whereas the mAP50-95 is a more comprehensive assessment of detection performance. The IoU gives an indication of how well the predicted mask or bounding boxes match the ground truth data.

$$\text{FPR} = \frac{\text{FP}}{\text{TN} + \text{FP}}$$

The false positive rate (FPR) indicates the model's ability to distinguish cocoa visitors from background images by quantifying the rate at which background images are incorrectly identified as containing cocoa visitors.

Data Record

The dataset is available from Zenodo Data Records³² and comprises 5,792 insect images in JPG format, each with a corresponding annotation file in TXT format. Additionally, it includes 1,082 flower background images in JPG format.

The dataset was curated from approximately 23 million images collected over a two-year period by embedded cameras deployed in cocoa plantations at the Xinglong Tropical Botanical Garden, Hainan Province, China. A combination of automated selection using a trained object detection model and manual screening was employed to extract high-quality, relevant images from this large image pool, ensuring the accuracy and usability of the final dataset.

To facilitate model training and evaluation, the insect images were divided into three subsets: train, test, and val. Each subset contains two subfolders: images, which stores the raw insect photos, and labels, which contains annotation files detailing the insect species and their bounding box coordinates. A classes.txt file is included in each labels folder to define the mapping between numeric class IDs and the corresponding insect species names.

Background images are stored separately in a folder named background, which contains two subfolders corresponding to the years 2023 and 2024. In total, 782 background photos were collected in 2023 and 300 in 2024. All images in this dataset were collected during field observations conducted in 2023 and 2024, and folder names reflect the year of collection for organizational clarity.

Technical Validation

We used a two-step approach to build a robust and adaptable model. First, we analyzed all the data from 2023, which included 5,214 images containing identified cocoa flower visitors and 500 selected background images (Fig. 1B). The model was then tested on a randomly selected subset of test data. We used 80%, 10%, and 10% of the entire dataset for training, validation, and testing, respectively. The background images were used to test the false positive rate (FPR). Due to the non-uniform distribution of data across different insect species, we used a weighted calculation method to determine the overall performance.

Second, we used the trained and validated model based on 5214 images from 2023 and tested model performance on 578 new images from 2024 (Fig. 1C). The ratio of the training, validation, and testing sets is also 8:1:1. Additionally, we tested the model's false positive rate (FPR) using 300 background images collected from 2024. The testing included evaluations with different model sizes and varying proportions of background images. We repeatedly split the training and validation data sets randomly at a fixed ratio. Each model was trained three times, and the reported results are the average performance metrics from multiple test set evaluations to capture variation and model's true capacity. Our goal was to conduct adaptability tests to enhance the reliability of the model's real-world effectiveness and generalization capabilities. Due to the limited number of images from spiders and aphids in the test set, we only used three of the five classes - Ceratopogonidae, Formicidae and Encyrtidae - in our adaptability test.

The model demonstrates high detection precision (0.98), recall (0.95) and F1-score (0.96), indicating accurate and comprehensive recognition across target objects. It achieves an mAP50 of 0.97 and an mAP50-95 of 0.60, showing stable performance across various IoU thresholds (Table 1). Overly optimistic detection results on test sets of different insect species are likely due to the homogeneity of the dataset. Strong internal consistency often leads to model overfitting to these patterns, high performance on the training and validation sets, but problems when deployed³³. Furthermore, the model has a relatively high false positive rate (FPR) of 9% on background images. This issue can be resolved with more data from different locations to enhance its generalization capability of the model.

Background image addition for model improvement. Background images in the training dataset can enhance object detection accuracy by enabling the model to distinguish between objects and their surroundings. We included different background images in the training dataset to avoid false negative detections. We gradually

Class	Images	Precision	Recall	F1-score	mAP50	mAP50-95
Overall	521	0.98	0.95	0.96	0.97	0.60
Ceratopogonidae	172	0.98	0.98	0.98	0.99	0.62
Formicidae	255	0.97	0.95	0.96	0.97	0.57
Aphididae	60	0.95	0.85	0.90	0.92	0.50
Araneae	22	1.00	1.00	1.00	0.99	0.76
Encyrtidae	13	1.00	0.92	0.96	0.96	0.74

Table 1. Performance evaluation of the five-classes YOLOv8 model.

Background images ratio	Precision (overall)	Recall (overall)	F1score (overall)	mAP50 (overall)	mAP50-95 (overall)	FPR
0%	0.74	0.65	0.69	0.69	0.30	0.041
5%	0.72	0.636	0.67	0.67	0.31	0.033
8%	0.78	0.65	0.71	0.70	0.31	0.026
10%	0.77	0.67	0.71	0.74	0.33	0.031
12%	0.72	0.59	0.64	0.65	0.307	0.013
15%	0.74	0.63	0.68	0.67	0.30	0.012

Table 2. Performance evaluation of the five-class YOLOv8 model with different background images ratio.

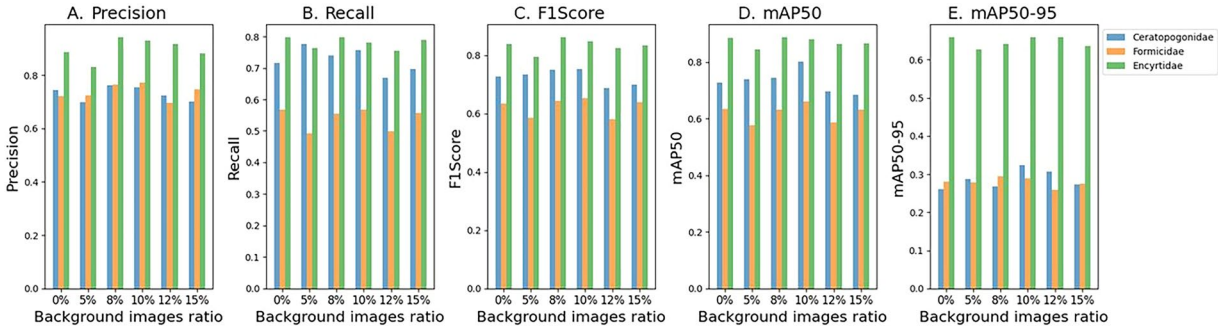


Fig. 4 Evaluation metrics for the adaptability test of the optimal models trained with background image proportions increasing from 0%, 5%, 8%, 10%, 12%, to 15%.

increased the percentage of background images in the training dataset from 0% to 15%, based on the 5,214 images collected in 2023.

When testing the model on images with completely unseen backgrounds, we found that training the model with an 8% background image ratio achieved the best Precision (0.78) and F1 Score (0.71) on the test set, while also yielding the lowest FPR (0.026), indicating the lowest risk of false positives. On the other hand, training with a 10% background image ratio resulted in the highest Recall (0.67), F1 Score (0.71), and mAP50 (0.74), along with a relatively low FPR (0.031). However, when the background image ratio was increased to 15%, despite achieving the lowest FPR (0.012), the overall performance of the model declined significantly (Table 2). Considering that real-world applications prioritize a balance in overall performance and minimal false positives, we concluded that the model trained with an 8% background image ratio is the optimal choice.

For a single class, Encyrtidae outperforms Ceratopogonidae and Formicidae in most metrics across different background image ratios (Fig. 4). For Encyrtidae, the model's performance is best when the background image ratio is 8%, achieving an F1 Score of 0.86 and an mAP50 of 0.89. For Ceratopogonidae and Formicidae, the model's F1 Score and mAP50 reach their highest values at a background image ratio of 10% (F1 Score: 0.75 and 0.65; mAP50: 0.80 and 0.66, respectively). This may indicate that the model has different sensitivity to background changes when processing different types of insects.

Analysis based on model size. Model size is mostly a trade-off between detection accuracy and computational demand, which must be evaluated for each application. In some cases, using a very large model may not necessarily result in higher accuracy. This phenomenon, known as underfitting, occurs when the model is too complex for the available data. To determine the optimal accuracy and model size, we conducted a series of tests on our dataset using different sizes of the YOLO model (for a detailed description of the YOLO model refer to¹³). Specifically, we evaluated the performance of YOLOv8 models with increasing complexity and size from YOLOv8n, YOLOv8s, YOLOv8m, and YOLOv8l. Our experiments aimed to identify the model that provides the best balance between accuracy and computational efficiency.

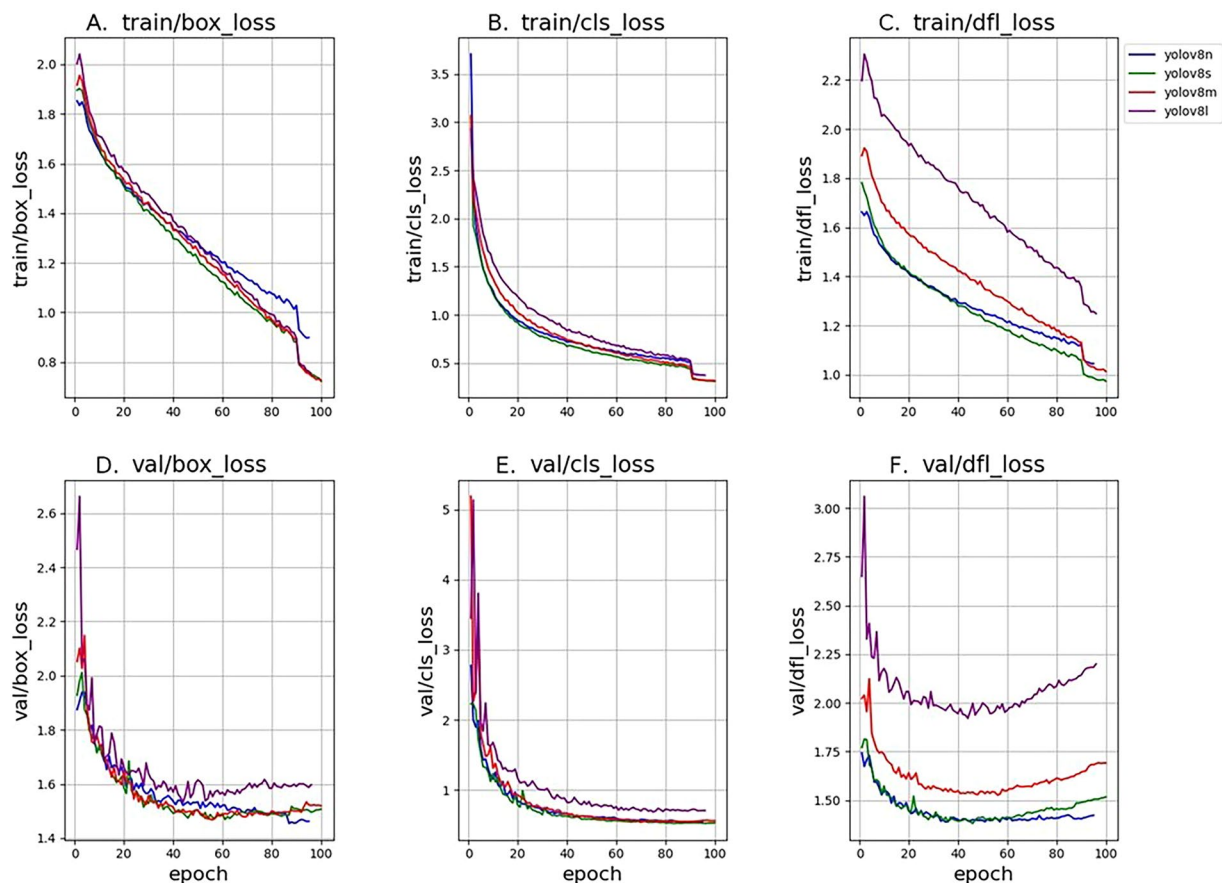


Fig. 5 Comparison of Loss Curves during training for models of different sizes. (A) train/box loss = Localization error for predicted vs. ground truth boxes during training; (B) train/cls_loss = Classification error during training; (C) train/dfl_loss = Focal loss optimizing bounding box regression during training; (D) val/box loss = Localization error during validation; (E) val/cls_loss = Classification error during validation; (F) val/dfl_loss = Focal loss for bounding box regression during validation.

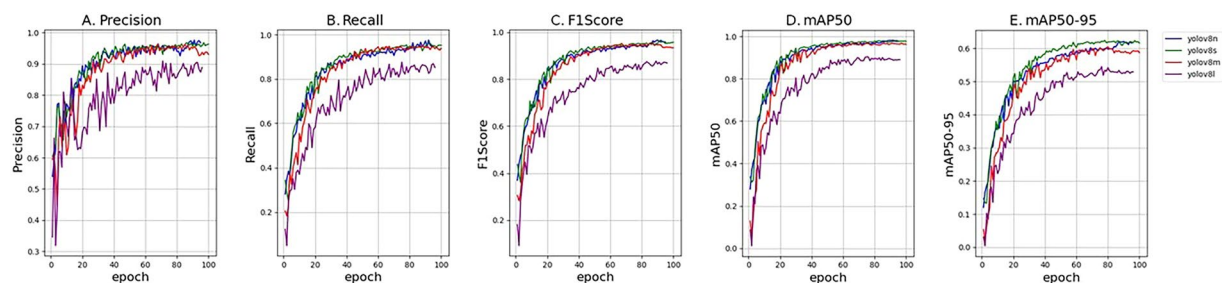


Fig. 6 Comparison of performance metrics. (A) Precision, (B) Recall, (C) F1 Score, and (D,E) Mean Average Precision (mAP) for YOLOv8 models of different sizes during training.

Performance and training effectiveness improved in all models as the number of training epochs increased (Fig. 5). Among them, the smaller models, YOLOv8n and YOLOv8s, converged faster, while the larger models, YOLOv8m and YOLOv8l, had a slower convergence rate but ultimately achieved lower loss values. From the perspective of training performance, the performance of YOLOv8n, YOLOv8s, and YOLOv8m are similar (Fig. 6). This suggests that a larger model size does not necessarily lead to better training results with our dataset.

When evaluating pre-trained YOLOv8 models on test images that were unseen during training, YOLOv8m demonstrated superior overall performance. It achieved the highest Precision (0.78), Recall (0.65), F1 Score (0.71), and mAP50 (0.70) among all tested models (Table 3).

As the model size increases, the false positive rate (FPR) for background images shows a gradual upward trend. This is attributed to the increased model complexity and the higher number of network parameters in YOLOv8m and YOLOv8l, which enhances the models' ability to extract detailed features—especially for small

Model	Params(M)	Precision (overall)	Recall (overall)	F1score (overall)	mAP50 (overall)	mAP50-95 (overall)	FPR
YOLOv8n	3.2	0.72	0.55	0.61	0.63	0.27	0.016
YOLOv8s	11.2	0.77	0.60	0.67	0.68	0.29	0.022
YOLOv8m	25.9	0.78	0.65	0.71	0.70	0.31	0.026
YOLOv8l	43.7	0.75	0.63	0.68	0.65	0.28	0.027

Table 3. Performance evaluation of YOLOv8 models of different size.

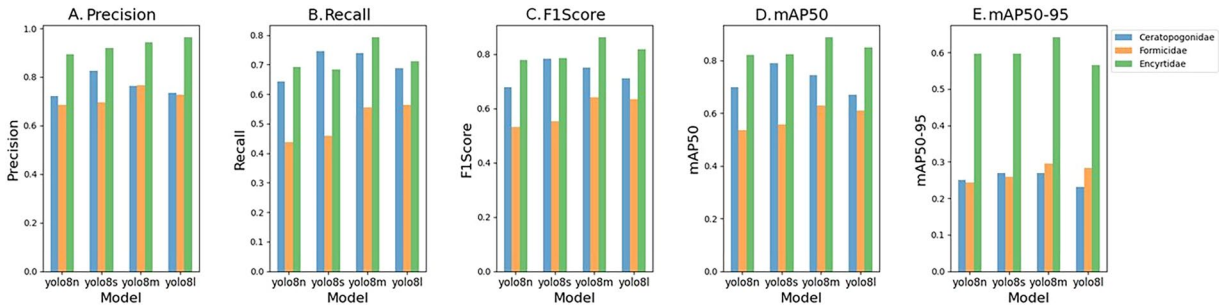


Fig. 7 Performance comparison of YOLOv8 models of different sizes on the test dataset.

objects and complex scenes. However, this improvement comes at the cost of a higher propensity to generate false alarms from intricate background features, leading to an elevated FPR. Despite this trade-off, YOLOv8m strikes a better balance between accuracy and false positive rate, making it a strong candidate for applications requiring both precision and robustness in complex environments.

In single-category detection, YOLOv8m performs better in the detection of Formicidae and Encyrtidae. Ceratopogonidae exhibited the best performance on YOLOv8s (Fig. 7). This indicates that the YOLOv8m model offers the most optimal performance in detecting visitors to cocoa flowers based on our dataset. Despite the smaller number of Encyrtidae images, the insects in these images were clearly visible and displayed distinct features. This clarity allowed the model to perform well even on images it had not seen during training. In contrast, the images of Ceratopogonidae were more numerous but often suffered from poor focus, leading to incomplete or indistinct insect shapes, sometimes appearing as mere black dots. This made it difficult for the model to distinguish the insects from the background. For Formicidae, the available images contained different species of varying sizes, which made it harder for the model to generalize effectively.

This work shows that parameter choice, and the percentage of background image inclusion was critical to enhance model performance to detect economically viable cocoa flower visitors. A medium-sized YOLOv8 model with 25.9 million parameters, trained on a dataset with 8% background images, achieved the best performance in recognizing three categories. The model attained a Precision of 0.78, Recall of 0.65, F1 Score of 0.71, and mAP50 of 0.70, with a false positive rate of 2.6%. These results suggest that to identify cocoa flower visitors under challenging field conditions, it is critical to increase the amount of data for each class to be detected and, therefore the total number of field deployments. To further enhance detection accuracy across different environments, future efforts could focus on optimizing detection algorithms or increasing the diversity of insect images. This work provides a foundational basis for advancing AI-driven solutions in the cocoa farming industry.

Code availability

The code used in this study is based on the open-source Ultralytics YOLO repository (<https://github.com/ultralytics/ultralytics>). Only minor parameter modifications (e.g., training epochs, image size, confidence thresholds) were made to adapt the model to our dataset. All scripts used to run training and inference, along with the configuration files, are openly available at Zenodo³². These materials allow full reproduction of the dataset generation and model outputs using the same tools.

Received: 6 January 2025; Accepted: 16 July 2025;
Published online: 28 July 2025

References

1. FAO. The State of Food and Agriculture 2023: Revealing the true cost of food to transform agrifood systems. FAO. <https://doi.org/10.4060/cc7724en> (2023).
2. Willett, W. *et al.* Food in the Anthropocene: the EAT–Lancet Commission on healthy diets from sustainable food systems. *The Lancet* **393**, 447–492, [https://doi.org/10.1016/S0140-6736\(18\)31788-4](https://doi.org/10.1016/S0140-6736(18)31788-4) (2019).
3. Wanger, T. C. *et al.* Integrating agroecological production in a robust post-2020 Global Biodiversity Framework. *Nat. Ecol. Evol.* **4**, 9, <https://doi.org/10.1038/s41559-020-1262-y> (2020).
4. Wanger, T. C. *et al.* Co-benefits of agricultural diversification and technology for the environment and food security in China. *Nat. Food* **1**, 1–4, <https://doi.org/10.1038/s43016-024-01075-x> (2024).
5. Díaz, S. & Malhi, Y. Biodiversity: Concepts, patterns, trends, and perspectives. *Annu. Rev. Environ. Resour.* **47**, 31–63, <https://doi.org/10.1146/annurev-environ-120120-054300> (2022).

6. Secretariat of the Convention on Biological Diversity. 2030 *Targets and Guidance Notes*. <https://www.cbd.int/gbif/targets/> (2023).
7. Darras, K. F. *et al.* Worldwide soundscapes: a synthesis of passive acoustic monitoring across realms. *bioRxiv*, <https://doi.org/10.1101/2024.04.10.588860> (2024).
8. van Klink, R. *et al.* Emerging technologies revolutionise insect ecology and monitoring. *Trends Ecol. Evol.* **37**, 872–885, <https://doi.org/10.1016/j.tree.2022.06.001> (2022).
9. Crall, J. D. *et al.* Neonicotinoid exposure disrupts bumblebee nest behavior, social networks, and thermoregulation. *Science* **362**, 683–686, <https://doi.org/10.1126/science.aat1598> (2018).
10. Zhao, W., Yamada, W., Li, T., Digman, M. & Runge, T. Augmenting crop detection for precision agriculture with deep visual transfer learning—A case study of bale detection. *Remote Sens.* **13**, 1, <https://doi.org/10.3390/rs13010023> (2021).
11. Wanger, T. C. & Frohn, P. Testing the Efficient Network TRaining (ENTR) Hypothesis: initially reducing training image size makes Convolutional Neural Network training for image recognition tasks more efficient. *Preprint at* <http://arxiv.org/abs/1807.11583> (2018).
12. Xia, D., Chen, P., Wang, B., Zhang, J. & Xie, C. Insect detection and classification based on an improved Convolutional Neural Network. *Sensors* **18**, 12, <https://doi.org/10.3390/s18124169> (2018).
13. Muppala, C. & Guruviah, V. Paddy pest identification with deep convolutional neural networks. *Eng. Agric. Environ. Food* **14**, 54–60, https://doi.org/10.37221/eaef.14.2_54 (2021).
14. Kasinathan, T., Singaraju, D. & Uyyala, S. R. Insect classification and detection in field crops using modern machine learning techniques. *Inf. Process. Agric.* **8**, 446–457, <https://doi.org/10.1016/j.inpa.2020.09.006> (2021).
15. Lander, T. A. *et al.* Global chocolate supply is limited by low pollination and high temperatures. *Commun. Earth Environ.* **6**, 97, <https://doi.org/10.1038/s43247-025-02072-z> (2025).
16. Toledo-Hernández, M., Tschardtke, T., Giannini, T. C., Solé, M. & Wanger, T. C. Hand pollination under shade trees triples cocoa yield in Brazil's agroforests. *Agric. Ecosyst. Environ.* **355**, 108612, <https://doi.org/10.1016/j.agee.2023.108612> (2023).
17. Toledo-Hernández, M. *et al.* Hand pollination, not pesticides or fertilizers, increases cocoa yields and farmer income. *Agric. Ecosyst. Environ.* **304**, 107160, <https://doi.org/10.1016/j.agee.2020.107160> (2020).
18. Wanger, T. C., Hölscher, D., Veldkamp, E. & Tschardtke, T. Cocoa production: Monocultures are not the solution to climate adaptation—Response to Abdulai *et al.* 2017. *Global Change Biol.* **24**, 561–562, <https://doi.org/10.1111/gcb.14005> (2018).
19. Toledo-Hernández, M., Wanger, T. C. & Tschardtke, T. Neglected pollinators: Can enhanced pollination services improve cocoa yields? *A review. Agric. Ecosyst. Environ.* **247**, 137–148, <https://doi.org/10.1016/j.agee.2017.05.021> (2017).
20. Darras, K. F. A. *et al.* Eyes on nature: embedded vision cameras for terrestrial biodiversity monitoring. *Methods Ecol. Evol.* **15**, 2262–2275, <https://doi.org/10.1111/2041-210X.14436> (2024).
21. Terven, J. & Cordova-Esparza, D. A comprehensive review of YOLO architectures in computer vision: From YOLOv1 to YOLOv8 and YOLO-NAS. *MAKE 5*, 1680–1716, <https://doi.org/10.3390/make5040083> (2023).
22. Girshick, R. Fast R-CNN. in *Proc. IEEE Int. Conf. Comput. Vis.* 1440–1448 http://openaccess.thecvf.com/content_iccv_2015/html/Girshick_Fast_R-CNN_ICCV_2015_paper.html (2015).
23. Nieuwenhuizen, A. T., Hemming, J. & Suh, H. K. Detection and classification of insects on stick-traps in a tomato crop using Faster R-CNN. in *The Netherlands Conf. Comput. Vis.* <https://library.wur.nl/WebQuery/wurpubs/542509> (2018).
24. Du, Y., Liu, Y. & Li, N. Insect detection research in natural environment based on Faster-R-CNN model. in *Proc. 2020 5th Int. Conf. Math. Artif. Intell. (ICMAI '20)* 182–186 <https://doi.org/10.1145/3395260.3395265> (2020).
25. Liu, W. *et al.* SSD: Single Shot MultiBox Detector. in *Comput. Vis.—ECCV 2016*, Leibe, B., Matas, J., Sebe, N. & Welling, M. (eds.) *Lect. Notes Comput. Sci.* **9905**, 21–37 https://doi.org/10.1007/978-3-319-46448-0_2 (Springer, Cham, 2016).
26. Lyu, Z., Jin, H., Zhen, T., Sun, F. & Xu, H. Small object recognition algorithm of grain pests based on SSD feature fusion. *IEEE Access* **9**, 43202–43213, <https://doi.org/10.1109/ACCESS.2021.3066510> (2021).
27. Garcia, R. G., Bicol, M. S. N., Cababat, A. M. E. & Pontigon, J. C. A Raspberry Pi microcontroller-based insect pests detection, counting and logging system in eggplants using SSD Lite MobileNetV2. in *2021 IEEE 13th Int. Conf. Humanoid, Nanotechnol., Inf. Technol., Commun. Control, Environ. Manage. (HNICEM)*, 1–6 <https://doi.org/10.1109/HNICEM54116.2021.9731906> (2021).
28. Redmon, J., Divvala, S., Girshick, R. & Farhadi, A. You only look once: Unified, real-time object detection. in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.* 779–788 https://www.cv-foundation.org/openaccess/content_cvpr_2016/html/Redmon_You_Only_Look_CVPR_2016_paper.html (2016).
29. Ratnayake, M. N., Dyer, A. G. & Dorin, A. Towards computer vision and deep learning facilitated pollination monitoring for agriculture. in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)* 2915–2924 <https://doi.org/10.1109/CVPRW53098.2021.00327> (2021).
30. Kumar, N., Nagarathna & Flammini, F. YOLO-based light-weight deep learning models for insect detection system with field adaptation. *Agriculture* **13**, 741, <https://doi.org/10.3390/agriculture13030741> (2023).
31. Vilar-Andreu, M., García, L., García-Sánchez, A.-J., Asorey-Cacheda, R. & García-Haro, J. Enhancing precision agriculture pest control: A generalized deep learning approach with YOLOv8-based insect detection. *IEEE Access* **12**, 84420–84434, <https://doi.org/10.1109/ACCESS.2024.3413979> (2024).
32. Xu, W. *et al.* Identifying Cocoa Flower Visitors: A Deep Learning Dataset. *Zenodo* <https://doi.org/10.5281/zenodo.15535674> (2025).
33. Torralba, A. & Efros, A. A. Unbiased look at dataset bias. in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.* 1521–1528 <https://doi.org/10.1109/CVPR.2011.5995347> (2011).

Acknowledgements

We would like to thank Professor Li Fupeng from the Spice Beverage Research Institute of the Chinese Academy of Tropical Agricultural Sciences for his support in data collection. We also thank students in Xinglong botanical garden for their help with data collection in 2023 and 2024. This work was funded through a Westlake University Startup Fund to T.C.W.

Author contributions

Conceptualization: W.X., S.G.B., Z.L., T.C.W.; Data collection: W.X., S.G.B., M.T.H.; Methodology and analyzes: W.X., S.G.B., Z.L., T.C.W.; Visualization: W.X., D.S., M.T.H.; Writing – original draft: W.X., S.G.B., D.S., T.C.W.; Writing – review & editing: all authors; Funding acquisition, Project administration & Supervision: T.C.W.

Competing interests

The authors declare no competing interests.

Additional information

Correspondence and requests for materials should be addressed to T.C.W.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Open Access This article is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License, which permits any non-commercial use, sharing, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if you modified the licensed material. You do not have permission under this licence to share adapted material derived from this article or parts of it. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by-nc-nd/4.0/>.

© The Author(s) 2025