

Mosquito Larvae Detection Model with YOLOv8

Yong-Sung Masuda

May 11, 2024

Purpose:

Counting mosquito larvae in an aliquot is a tedious task faced by PhD students researching mosquito population control. This paper presents a machine learning approach to this problem, leveraging a minimal dataset of only 96 images. The goal is to develop a model that can accurately detect and count mosquito larvae in images taken in a laboratory environment.

Dataset:

The model's inputs are images of mosquito larvae and mosquito eggs in an aliquot, and the output is the location, dimensions, and classifications of the objects detected within these two classes. Initially, the model was trained to detect live larvae only, as this is the targeted class to be identified. This resulted in the eggs occasionally being misclassified as larvae, so the model is now trained to distinguish between live larvae and eggs to reduce this error. Due to the limited nature of the dataset, a transfer learning method is implemented using the generic object detection model YOLOv8.

The dataset consists of 96 images of size 2592 by 1944 pixels. The images have a uniform background and lighting and contain three classes of objects: live larvae, dead larvae, and eggs. Because there are very few samples of dead larvae in the dataset, this class is excluded from the training process. Each object within the images is labeled by class with bounding boxes tightly surrounding them. The data is randomly permuted and split as follows:

- Training set: 67 images (70%)
- Validation set: 19 images (20%)
- Test set: 10 images (10%)

The model is trained on the training set and the validation set was used for hyperparameter selection and early stopping. The test set is used to evaluate the final model.

Preprocessing & Tuning:

Preprocessing includes auto-orienting the images to ensure the bounding boxes match the image orientation. The images are not resized due to the small nature of the dataset, as well as to avoid requiring preprocessing during the usage of the trained model. The bounding boxes

are encoded using the YOLO format, which consists of its class number and four values between 0 and 1 describing its center coordinate and height and width relative to the dimensions.

Hyperparameter tuning for this model was conducted using the YOLO model tune method to optimize the model's performance. YOLO tune method uses the mutation algorithm which searches the hyperparameter space by applying small random changes to existing hyperparameters. Tuning was conducted with AdamW optimization for 93 iterations for 30 epochs each. The hyperparameters which were included in the tuning process and their final values are as follows:

Hyperparameter	Value
lr0	0.00838
lrf	0.01346
momentum	0.86808
weight_decay	0.00041
warmup_epochs	3.37112
warmup_momentum	0.88987
box	9.86137
cls	0.44448
df	1.59169
hsv_h	0.01633
hsv_s	0.54486
hsv_v	0.55232
degrees	0
translate	0.08194
scale	0.45391
shear	0
perspective	0
flipud	0
fliplr	0.53219
bgr	0
mosaic	0.94871
mixup	0
copy_paste	0

Training & Evaluation:

The base model used is “yolov8n” and is trained using the best hyperparameters from the tuning process. Training was set to be conducted for 500 epochs with early stopping and a patience of 100. Training ended after epoch 376 and the best results were observed at 276. Figure 1 shows the labels manually added by humans alongside the model's predictions and their corresponding confidence levels. These predictions were generated by the final model on the test data that was not included in the training or validation processes.

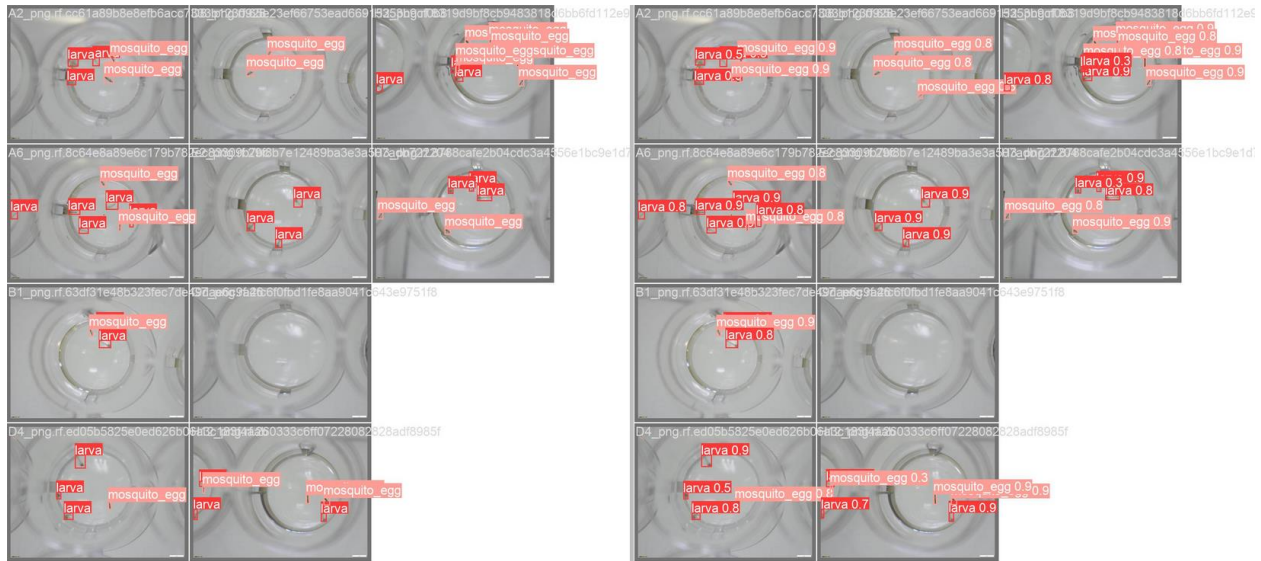


Figure 1: Human-added labels (left) vs model predictions (right)

The performance of this model is evaluated using the following metrics:

- **Precision-Recall:** A plot of the trade-off between:
 - Precision: the ratio of true positive predictions to the total number of positive predictions.
 - Recall: the ratio of true positive predictions to the total number of actual positive instances in the data.
- **Precision-Confidence:** A plot of precision at different confidence thresholds.
- **Recall-Confidence:** A plot of recall at different confidence thresholds.
- **F1-Confidence:** A plot of the F1 score at different confidence thresholds. The F1 score is the harmonic mean of precision and recall.

The F1 score for this model has a peak value of 0.94 at a confidence threshold of 0.32.

Distribution Shift:

While this model has been highly evaluated for its performance, its ability to generalize is not thoroughly proven due to the minuscule size of the test set. Since all training was conducted using images from the same source and in the same setting, it is likely that the model will not generalize well to photos taken under different conditions, such as those with a different background or lighting. Any variation in these factors could significantly impact the model's ability to accurately detect and classify mosquito larvae.

