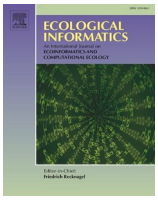
Ecological Informatics 80 (2024) 102543

Contents lists available at ScienceDirect

Ecological Informatics

journal homepage: www.elsevier.com/locate/ecolinf



A lightweight and enhanced model for detecting the Neotropical brown stink bug, *Euschistus heros* (Hemiptera: Pentatomidae) based on YOLOv8 for soybean fields

Bruno Pinheiro de Melo Lima [a], Lurdineide de Araújo Barbosa Borges [b], Edson Hirose [c], Díbio Leandro Borges [a,d,*]

[a] University of Brasilia, Department of Mechanical Engineering, Brasilia, DF, Brazil
[b] EMBRAPA Cerrados, BR-020, km 18, Planaltina, DF, Brazil
[c] EMBRAPA Soybean, Rodovia Carlos Joao Strass, s/n, Londrina, PR, Brazil
[d] University of Brasilia, Department of Computer Science, Brasilia, DF, Brazil

ARTICLE INFO

Keywords:
Heteroptera
Insect pest detection
Improved YOLO model
Image-based detection and counting
Deep learning
Soybean field images

ABSTRACT

Insect pest detection and monitoring are vital in an agricultural crop to help prevent losses and be more precise and sustainable regarding the consequent actions to be taken. Deep learning (DL) approaches have attracted attention, showing triumphant performance in many image-based applications. In the adult stage, this research considers detecting a vital insect pest in soybean crops, the Neotropical brown stink bug (*Euschistus heros*), from field images acquired by drones and cellphones. We develop and test an improved YOLO-model convolutional neural network (CNN) with fewer parameters than other state-of-the-art models and demonstrate its superior generalization and average precision on public image datasets and the new field data provided here. Considering the proposal's precision and time of response, the possibility of deploying this technology for automatic monitoring and pest management in the near future is promising. We provide open code and data for all the experiments performed.

1. Introduction

Smartly monitoring insects in a cropping system is highly significant due to economic and sustainability issues. More than a hundred insect species might be present in a particular crop cycle, especially in tropical areas, and only a few of them could be considered pests to a specific crop, with some being neutral, and a considerable amount is even beneficial. Nowadays, with the advent of precise cameras and powerful computers at affordable prices, artificial intelligence techniques such as deep learning (Butera et al., 2021) provide a spectrum of possibilities to approach insect pest detection for real-time applications. (Li et al., 2021) and (Kasinathan et al., 2021) provide two interesting and timely reviews of machine and deep learning techniques for intelligent management of insects with field images, where relevant successful approaches are shown, and challenges to be pursued are commented on. Most of the recent approaches are designed for a particular crop or even for a particular insect pest, since the relevance of controlling an insect pest in a more sustainable way poses economic and practical issues. This type of technology is undoubtedly changing the way we can identify insects in the wild. Moreover, new developments are needed (Høye et al., 2021).

Despite its economic significance, the excessive and indiscriminate use of pesticides poses a significant economic and environmental challenge since insecticides contribute more than 20% of production costs (Bueno et al., 2011). Precision farming techniques have emerged, aiming to precisely locate pests, diseases, and deficiencies, enabling targeted interventions and reducing waste.

Accordingly, computer vision models have recently been employed in diverse platforms to identify insects and weeds. In general, the approach aims to identify the pest as quickly and accurately as possible in the early stages of the crop. The challenges are considerable, since there are visual similarities in insects morphology between species, camouflage strategies, habits and preferences depending on cropping environment. Some successful approaches rely on sticky traps to capture insect individuals, and then apply algorithms to identify and count the trapped insects (Tang et al., 2023), (Ciampi et al., 2023). Trying to detect and identify insect species in open field images and video streams

* Corresponding author at: University of Brasilia, Department of Computer Science, Brasilia, DF, Brazil.
E-mail address: dibio@unb.br (D.L. Borges).

https://doi.org/10.1016/j.ecoinf.2024.102543
Received 12 November 2023; Received in revised form 22 February 2024; Accepted 22 February 2024
Available online 27 February 2024

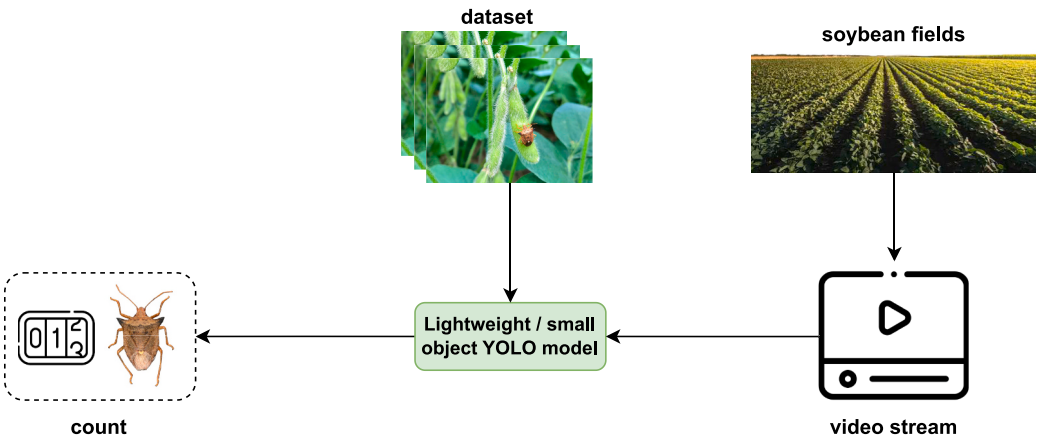


Fig. 1. General schematics of the proposed method showing it can receive different input from real fields and produce detection and counting of NBSB.

poses more difficulties than using sticky traps (Li et al., 2021). A trend lately is to encourage collecting field images with cameras on drones, labeling them, and design more efficient algorithms to improve performance on classification. Integrated platforms, onboard drones with recognition software, would aim to map areas for pest management (Kern et al., 2020).

Soybean is a globally significant crop, valued for its high protein and oil content, making it a crucial source of vegetable oil and animal protein feed (Masuda and Goldsmith, 2009). Among the pests most commonly found in soybean crops in tropical regions, the Neotropical brown stink bug (NBSB) *Euschistus heros* (Hemiptera: Pentatomidae) is one of the most critical (Bortolotto et al., 2015). Adults have an average length of 10 mm. So, detecting NBSB in soybean fields is of utmost importance in agricultural settings due to its potential to cause significant crop damage. Traditional detection methods, such as using the beat cloth, can be time-consuming and labor-intensive, in addition to demanding the hiring of qualified and expensive professionals to carry out the operation. Therefore, there is a need to explore efficient and real-time solutions to detect and count these pests accurately in the early stages of crop infestation. This paper addresses this research question: How can a fast and precise enough classification model for NBSB be developed using field images and a state-of-the-art algorithm?

Some recent works have diligently approached the problem of classifying insect pest images in soybean crops. Tetila et al. (2020b) evaluated three CNN models, DenseNet-201, Inception-Resnet-v2, and Resnet-50, in soybean field images after SLIC superpixel segmentation. A follow-up of that work was published by (Tetila et al., 2020a) to evaluate the fine-tuning strategies of Inception-v3, Resnet-50, VGG-16, VGG − 19, and Xception, where accuracy classifications, over 90% were reported. Those models need to be faster for real-time applications.

YOLO (You Only Look Once) (Redmon et al., 2016) models are widely used for object detection but may struggle with accurately detecting small objects and have high computational demands. Despite these limitations, YOLO's real-time processing and overall performance make them favored in computer vision applications. Researchers are continually improving YOLO models to enhance their versatility and efficiency. Silveira et al. (2021) assessed YOLOv3 for real-time insect pest detection in soybean fields and failed in some cases. Verma et al. (2021) compared YOLOv4 and YOLOv5, with YOLOv5 achieving the highest insect detection accuracy. Also, (Önler, 2021), used YOLOv5 to identify thistle caterpillars in sunflower cultivation. In turn, (Ahmad et al., 2022) developed an object recognition system for different insects based on various YOLO architectures, with YOLOv5x outperforming others. And more recently (Khalid et al., 2023), YOLOv8 achieved high-precision figures for early pest detection when compared with other YOLO architectures.

Researchers have made significant efforts to address the challenges YOLO models face in accurately detecting small objects and their computational demands. Various architectural modifications have been proposed to enhance the performance and efficiency of YOLO-based object detection systems. These advancements aim to improve small object detection and optimize computational resources for real-world applications.

One approach to improving small object detection from YOLOv5 is the YOLO-Z model proposed by (Benjumea et al., 2021). Enhancing YOLOv5 with the incorporation of attention mechanisms, specifically channel and spatial attention modules, has shown significant refinement of the focus of feature maps (Yuan et al., 2021). Yuan et al. (2022) introduced YOLOv5-tiny, a miniature aggregate detection and classification model that outperformed other object detection algorithms regarding precision. An MD-YOLO model was proposed by (Tian et al., 2023) with three key components: an image feature extraction part, a feature fusion network, and a prediction module for detecting some pests in field images. Zhan et al. (2022) proposed four design variations in a CNN model to improve small object detection in drone-captured scenes.

One lightweight object detection method, YOLOLite-CSG, was proposed by (Cheng et al., 2022) and designed for low-performance devices in agricultural environments. Based on YOLOLite with optimized precision, reduced parameters, and enhanced spatial information via k-means++, sandglass blocks, and coordinate attention. Liu et al. (2023) developed an improved YOLOv5 algorithm for UAV capture scenes to enhance feature extraction capacity and detection performance for medium- and long-range objects. Li et al. (2023) proposed an algorithm with a new point-line distance loss function, attention module, and mixup online data augmentation to achieve high mean average precision while retaining the lightweight characteristics of traditional YOLOv5. Huang et al. (2023) and Mahaur and Mishra (2023) have proposed interesting modifications to basic YOLO architectures to detect small objects well.

Therefore, the literature showcases ongoing efforts to address the challenge of detecting small objects, like insects, indicating that the problem remains relevant and unsolved. Recent work on YOLOv8 includes optimization using the simulated annealing (SA) algorithm for crop pest detection (Kang et al., 2023), as well as an improved feature fusion network and new network structure for improved detection accuracy (Lou et al., 2023). However, a good compromise between speed and accuracy remains to be dealt with in practical scenarios such as field crops.

Given the architectural affinities between YOLOv5 and YOLOv8, specific proposed adaptations for YOLOv5 hold promise in fashioning an improved YOLOv8 model tailored for the discernment of NBSB in soybean fields. In light of this, this research proposes modifications within the YOLOv8 bottleneck, representing a strategic step forward. This

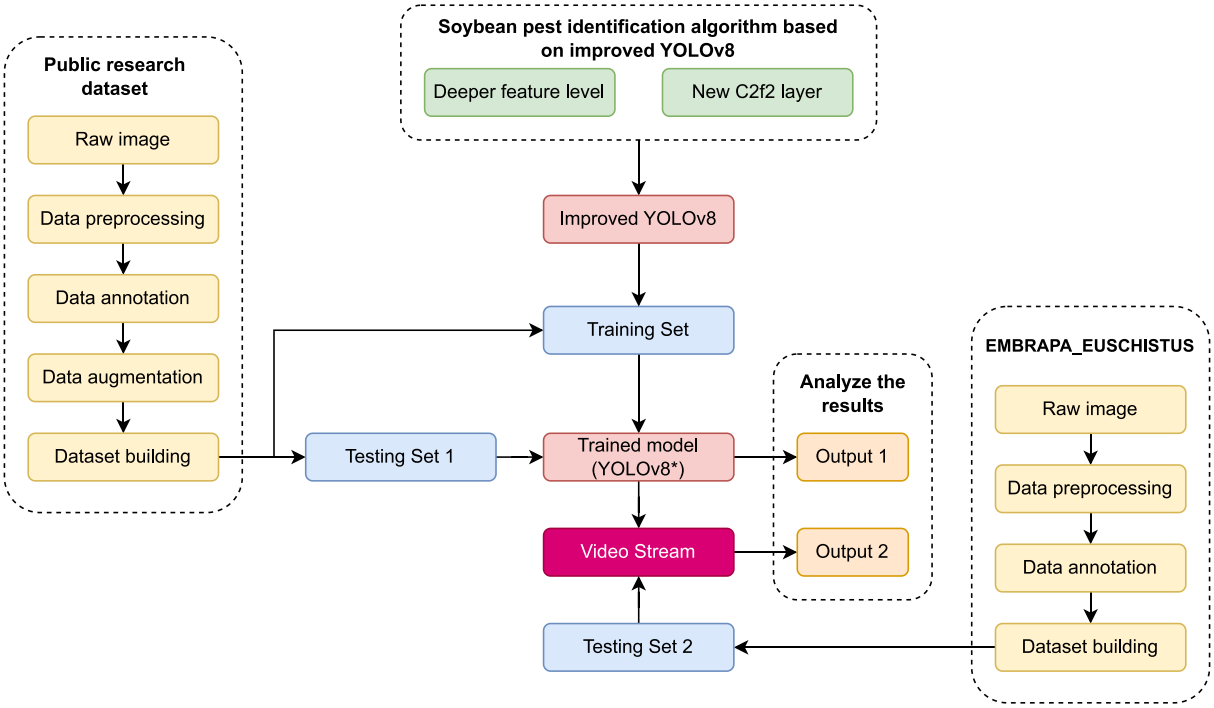


Fig. 2. Schematic diagram about the construction of the model to be embedded into the pipeline.

proposal is especially crucial in insect detection, where precision and efficiency are paramount.

This study approached the problem of detecting and counting NBSB through images and videos acquired with drones. The state-of-the-art methods nowadays are based on deep learning algorithms, and we have developed an improved YOLOv8 model with architecture modifications that have allowed a good compromise between accuracy and speed of processing after it has been trained. We tested the model with a public image dataset of NBSB in soybean fields and added a novel one with more images. The results have shown that the system can truly help automate this task, which will allow better pest management control, and it can be extended to other pests and crops in the future. Fig. 1 brings general schematics of the proposal, where the model receives images or videos from real field conditions and produces detection and counting of NBSB.

This research addresses those main issues with the following highlights as contributions:

- an improved YOLOv8 model with an adapted P2 level and C2f2 layer, well tailored for small object detection, such as insects in field images;
- ablation experiments with the new modules showing the effectiveness of their inclusion;
- tests with a benchmark public dataset and with novel field-collected images of soybean fields;
- a novel dataset of NBSB (ground truth) images in soybean fields for testing;

The remainder of the paper describes the materials and methods in Section 2, and it shows results and discussion in Section 3, finishing with conclusions in Section 4.

2. Materials and methods

In order to develop the proposed model based on YOLOv8, we evaluated the impact of different modules, namely the P2 feature level and C2f2 layer, on the performance of the NBSB object detection algorithm. We evaluated ablation experiments under the same experimental conditions, where the new algorithm was trained and tested on the INSECT10K7C640_SAT dataset (Lima, 2023). The new algorithm's performance is compared against three models: A) YOLOv8n, B) YOLOv8n with C2f2 only, and C) YOLOv8n with P2 only. Fig. 2 summarizes the complete proposed process.

2.1. YOLOv8

YOLOv8 (Ultralytics, 2023) is the latest version released by Ultralytics of the popular real-time object detector YOLO (You Only Look Once) (Redmon et al., 2016). It is designed to combine the advantages of many other real-time object detectors, including a lightweight network architecture, effective feature fusion methods, and more accurate detection results.

YOLOv8 introduces a state-of-the-art model with advanced object detection networks and instance segmentation capabilities, adaptable to diverse project requirements through scalable models akin to YOLOv5. YOLOv8 uses Anchor-Free (instead of Anchor-based) (Ultralytics, 2023), a method where object detection models directly predict the object's center without using anchor boxes. Anchor boxes are pre-defined boxes with specific heights and widths that detect object classes with the desired scale and aspect ratio. During detection, these anchor boxes are tiled across the image, and the network outputs probability and attributes for each box, which are then used to adjust the anchor boxes. However, anchor-free detection is more flexible and efficient than the previous YOLO models, as it does not require manual specification of anchor boxes, which can lead to suboptimal results.

The C2f module is another improvement of YOLOv8. This module was designed by referring to the ELAN structure in YOLOv7 and incorporating it while retaining the original idea of YOLOv5 (Lou et al., 2023). The Bottleneck in YOLOv8 is similar to that in YOLOv5 but with a 3×3 kernel size for the first convolution instead of the 1×1 kernel size in YOLOv5. YOLOv8's C2f module differs in how it handles the bottleneck outputs, consisting of two 3×3 convolutions with residual connections. In C2f, all the outputs from the bottleneck are concatenated, while in C3, only the output of the last bottleneck is used.

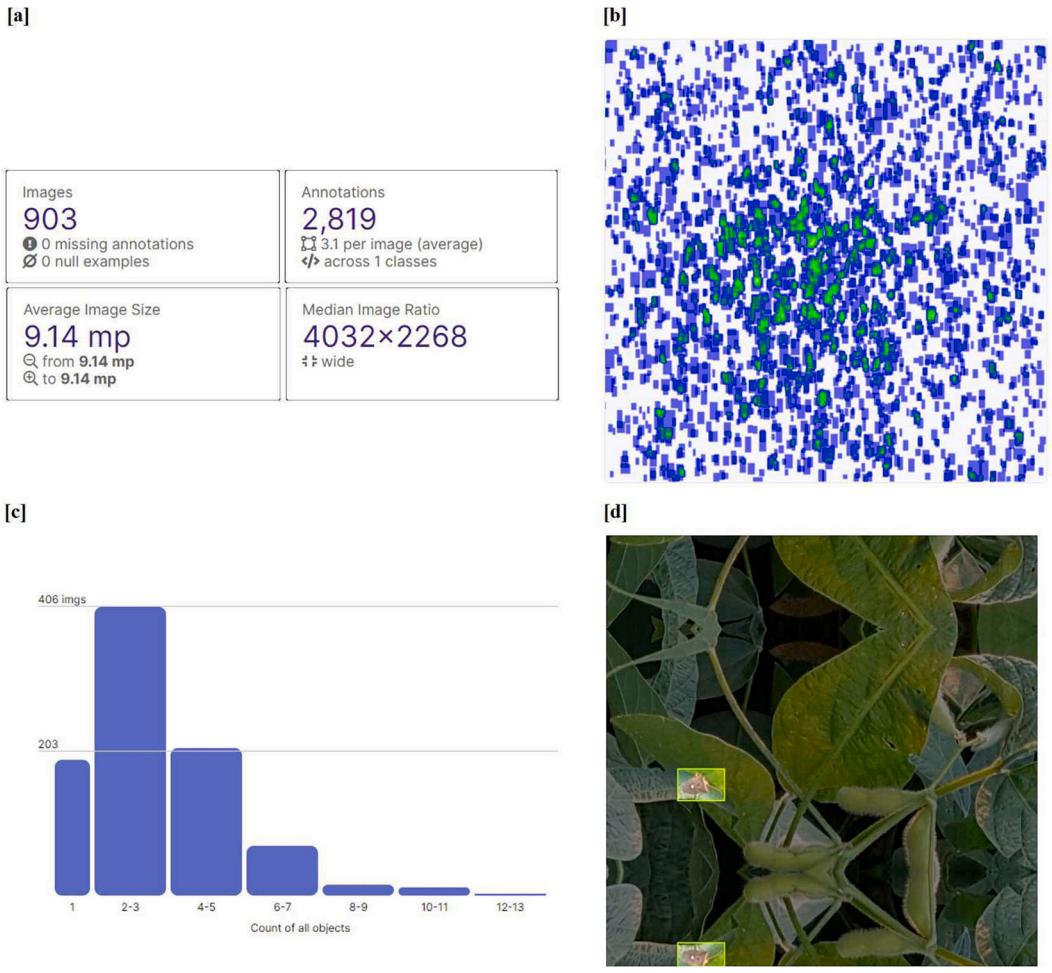


Fig. 3. (a) INSECT10K7C640_SAT dataset main data; (b) NBSB distribution over scene; (c) Count of insects over images and (d) Sample image from our dataset, comprising scene with NBSB annotated and background only after preprocessing.

2.1.1. P2 feature level

As the preceding YOLOv5 network, YOLOv8 uses a feature pyramid architecture. It extracts features from different scales of the input image and combines them to generate a set of detection predictions. As part of this YOLOv5 architecture, the P2 layer is a CNN layer that is part of the object detection process and is responsible for receiving outputs from previous layers and generating the necessary outputs for the next layer. Specifically, the P2 layer is part of a section of the network called the "neck" that combines low and high-level features extracted from previous layers to obtain richer information about the image.

The P2 layer receives the output tensor from the previous layer as input. It applies convolution, normalization, and activation operations to generate a new tensor representing image features at different scales. The next layer then uses this output to generate the final outputs, representing object detections in the image. The P2 layers are specifically designed to extract features at a smaller scale, which can help identify small objects that other layers in the network may miss. On the other hand, by adding the P2 layer, the network becomes more profound, with more layers of computation. This increased depth can help the network to learn complex representations of the input image better and extract more informative features, leading to improved performance in object detection tasks, particularly for small objects. As part of this work, a P2 layer with some corrections was integrated into this architecture.

2.1.2. Changing the number of filters and its effects

Reducing the number of filters in the YOLOv8 object detection model can have several expected effects on its performance and characteristics. The number of filters in the model's convolutional layers directly impacts its ability to learn and represent features from the input data.

One of the most immediate effects of reducing the number of filters is a decrease in model complexity. The number of parameters and gradients will be reduced, resulting in a more lightweight model. That can be beneficial for scenarios where computational resources are limited, making the model more efficient for deployment on resource-constrained devices Zhang et al. (2019). While a smaller model may require less computational power, it is also likely to decrease detection accuracy. Because of that, it may restrict the model's capacity to represent a wide range of object features effectively. That could lead to difficulties in detecting objects with varying scales, shapes, or orientations, which does not seem to be an immediate problem, considering the low variability of NBSB, if we limit ourselves to detecting only more advanced stages of the insect, as is the case.

On the positive side, a smaller model might generalize better on data from previously unseen domains or categories, with a reduced capacity to memorize training data; the model may focus on learning more generic features that can be useful across different datasets (Seemakurthy et al., 2022). As we will see later, this is a goal we seek in our model.

A reduction in the number of filters can lead to faster inference times. Using smartphone applications, the model may process images more quickly, making it suitable for real-time applications or scenarios that demand rapid detection, such as NBSB detection and counting (Diwan et al., 2023). However, care must be taken, as if the number of filters is decreased, the model might become more prone to overfitting the

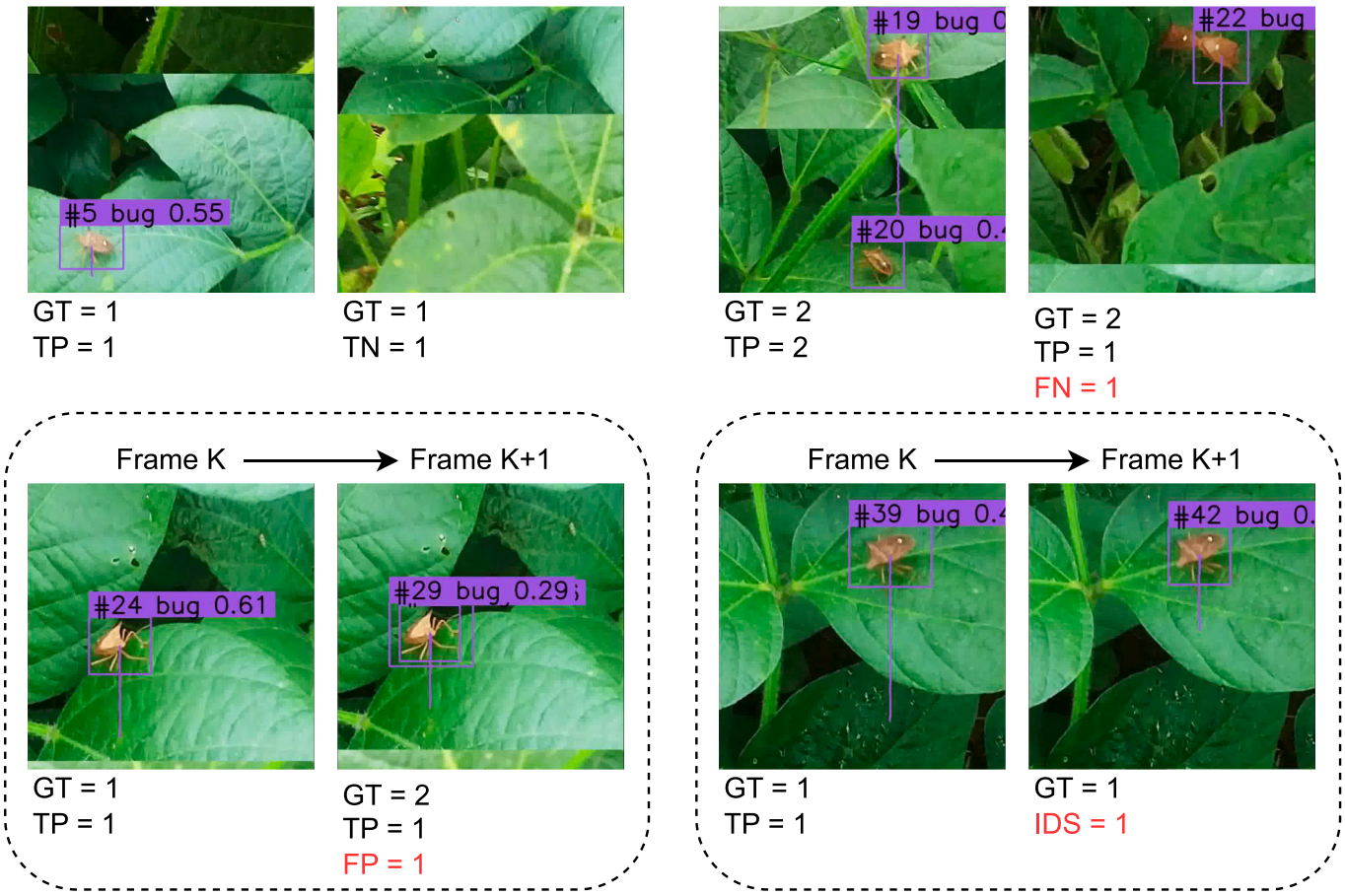


Fig. 4. Summary of metrics and their use cases in the tracking problem for later application on insect counting.

training data, and it could struggle to capture complex patterns, resulting in a loss of performance when dealing with new, unseen examples (Diwan et al., 2023).

2.2. Datasets

Our study used a dataset of plantation images collected from a soybean agricultural area in Dourados-MS, Brazil, called INSECT10K7C and available (Tetila, 2019). The dataset consists of 1000 digital images captured locally by the researchers using a digital camera equipped with a 12.2-Mpixel resolution 1/2.5" Samsung SM-G930F sensor. The images were recorded at a 1-m altitude above the plantation, using an angle of approximately 45° from the camera to the ground. The images were collected during the reproductive phenological stages R1–R6 of the soybean reproductive phase in the 2018/19 crop, on different days and in varied climatic conditions between 6 P.M. and 7:30 P.M.

The original dataset used in this study consisted of 903 images with at least one annotation. These raw images were first manually annotated using Make Sense, with a single "bug" class, fully containing the visible outline of the NBSB insects. We then applied several preprocessing and augmentation steps to the annotated images using Roboflow tools to enhance the dataset's diversity and increase its size for more effective model training. These steps generated additional data points, ultimately expanding the dataset to 4000 images. The dataset was obtained at the end of these steps, as shown in Fig. 3.

During the preprocessing phase, techniques like auto-orientation, resizing, tiling, and filtering were utilized to standardize and enrich the dataset. These steps ensured that the images were consistently oriented, had a uniform size of 640 × 640 pixels, and contained a substantial amount of annotated data, meeting the 90% annotation threshold. Augmentation played a crucial role in further diversifying the dataset. By introducing random saturation adjustments to each training example, the number of outputs per example was increased to two. This augmentation strategy introduced variations in color intensity, contributing to a more comprehensive and robust training dataset.

The training set, comprising the majority of the data (82%), containing 4000 images, is used to train the model and adjust its parameters, allowing it to learn from a diverse range of examples and patterns in the data. The validation set (13%), consisting of 607 images, is utilized during training to fine-tune hyperparameters and assess the model's performance on unseen data, helping to prevent overfitting and ensuring generalization. Lastly, the testing set (5%), which included 260 images, serves as an independent evaluation of the model's performance on completely unseen data, providing a reliable measure of its real-world effectiveness and ability to generalize.

2.3. Performance evaluation

To evaluate the ablated model results, we use five metrics, namely Precision (Eq. (1)), Recall (Eq. (2)), mAP0.5 and mAP0.5:0.95, related to Eq. (3), Params(M), Flops(G), Inference(ms) and Time(h).

$$Precision = \frac{TP}{TP + FP} \tag{1}$$

$$Recall = \frac{TP}{TP + FN} \tag{2}$$

The evaluation of insect detection is performed using detection scores. A detection score of at least 0.5 is required to classify the insect as a true positive (TP). Incorrect identification of an object, such as leaves

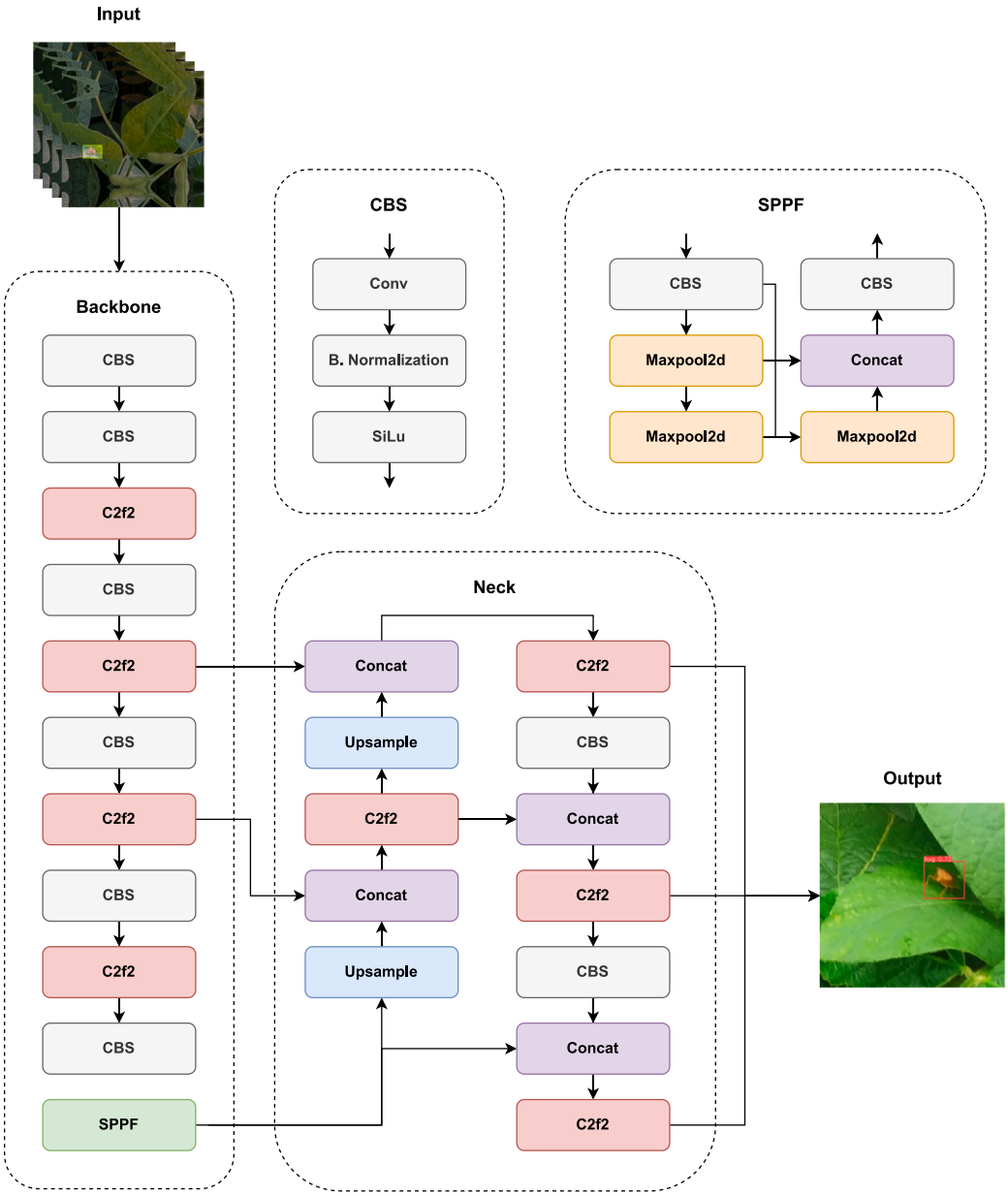


Fig. 5. Representation of the structure of the improved YOLO model.

or background, as an insect is considered a false positive (FP). Failure to detect an insect or incorrectly detect it in a different class is considered a false negative (FN). In cases where no insect is present in the image, a true negative (TN) is recorded.

The mean average precision (mAP) is the primary evaluation index used to measure network detection performance, which considers both precision and recall, defined in Eqs. (1) and (2), respectively. The mAP is calculated by averaging the precision at various recall values obtained from the precision-recall (PR) curve, as in Eq. (3). This evaluation metric comprises mAP0.5 and mAP0.5:0.95.

$$mAP = \frac{1}{k}\sum_{i=1}^{k} AP_i \tag{3}$$

We also consider Params and FLOPs to evaluate the model. The "Params" metric in a neural network model refers to the number of learned variables used for making predictions. It is an essential metric for evaluating the model's complexity and computational efficiency. Models with more parameters generally require more resources for training and inference, but they may also have higher accuracy. FLOPS stands for "Floating Point Operations per Second", it measures how many floating point arithmetic operations a model can perform per second, being a hardware-dependent measure.

Performance parameters such as inference time (in ms) and time taken for training (in hours) are critical metrics for evaluating the efficiency of YOLO models. Inference time measures the speed at which the model processes input data and produces output predictions. It is essential to minimize inference time for real-time applications where speed is crucial, such as real-time object detection in videos. On the other hand, time taken for training measures how long it takes for the model to learn from the training data and improve its accuracy. This metric is essential for measuring the overall training efficiency of the YOLO model. Considering these two parameters when selecting a YOLO model for a particular use case is essential, as they can impact the model's performance and computational cost.

These concepts change slightly for insect counting in video streams, based on differences recorded between each video frame. A true positive

Table 1
The detailed structure of YOLOv8n + P2 feature level.

From	Repeats	Module	Arguments
−1	1	Conv	[64, 3, 2]
−1	1	Conv	[128, 3, 2]
−1	3	**C2f2**	[128, True]
−1	1	Conv	[256, 3, 2]
−1	6	**C2f2**	[256, True]
−1	1	Conv	[512, 3, 2]
−1	6	**C2f2**	[512, True]
−1	1	Conv	[1024, 3, 2]
−1	3	**C2f2**	[1024, True]
−1	1	SPPF	[1024, 5]
−1	1	Upsample	[None, 2, "nearest"]
(−1, 6)	1	Concat	[1]
−1	3	**C2f2**	[512]
−1	1	Upsample	[None, 2, "nearest"]
(−1, 4)	1	Concat	[1]
−1	3	**C2f2**	[256]
−1	1	Upsample	[None, 2, "nearest"]
(−1,2)	1	Concat	[1]
−1	3	**C2f2**	[128]
−1	1	Conv	[128, 3, 2]
(−1, 15)	1	Concat	[1]
−1	3	**C2f2**	[256]
−1	1	Conv	[256, 3, 2]
(−1,12)	1	Concat	[1]
−1	3	**C2f2**	[512]
−1	1	Conv	[512, 3, 2]
(−1, 9)	1	Concat	[1]
−1	3	**C2f2**	[1024]
(18, 21, 24, 27)	1	Detect	[nc]

(TP) is considered when a new-appearing insect (from frame k to k + 1) receives a new ID. A false negative (FN) occurs when a new insect in the video does not receive a new ID or its track is interrupted between frames k and k + 1. A false positive (FP) is computed when an insect previously identified receives another ID simultaneously with the previously assigned ID. An ID switch(IDS) occurs when an insect changes its ID between frame k and the subsequent frame k + 1. Finally, a true negative (TN) occurs when the model correctly does not perform a new insect count when there is no insect in the frame. It is also worth noting that each of these events on the scene is recorded as Ground Truth (GT) so that when there are two insects on the scene, we have a GT equal to 2. The same occurs when there is a duplicate ID, an FP and a TP simultaneously. These predicted situations are summarized in Fig. 4.

With these primary metrics, we calculate the multi-object tracking accuracy (MOTA), Bernardin et al. (2006) of the experiment, the primary evaluation metrics for the tracking performance, according to Eq. (4). MOTA ranges from $-\infty$ to 1 and can be multiplied by 100 to get MOTA in percentage. The tracking quality is better when MOTA value is closer to 1, being deficient when this value is 0 or less.

$$MOTA = 1 - \frac{FN + FP + IDS}{GT} \tag{4}$$

2.4. Proposed approach

The original YOLOv8 model is highly effective; however, it still struggles to accurately detect small targets in complex scenes; therefore, it is a problem that still needs to be solved entirely. The issue lies mainly in feature extraction, where more extensive features often overshadow small targets. Extracted features lack small-target information, leading to poor detection results. Furthermore, small targets are more likely to overlap with other objects, making them harder to distinguish and locate in the image (Lou et al., 2023). To solve the mentioned problems, we proposed an improved detection algorithm, the structure is shown in Fig. 5, that detects small-size targets, such as *Euschistus Heros*, and it is also a lighter and faster model than the original YOLOv8.

The YOLOv8 algorithm offers a range of network structures, including YOLOv8n, YOLOv8s, YOLOv8m, YOLOv8l, and YOLOv8x. While they differ in width and depth, they follow the same principles and can be chosen according to specific needs. The deeper the structure, the higher the precision, but the slower the training and inference speed. YOLOv8n was chosen as the base structure to prioritize speed without compromising accuracy, with further enhancements to improve performance.

Adding the P2 feature level to the YOLOv8 architecture makes the network deeper because an additional layer is added to the overall network structure. The YOLOv8 architecture already includes a series of convolutional layers and a neck section that combines features from

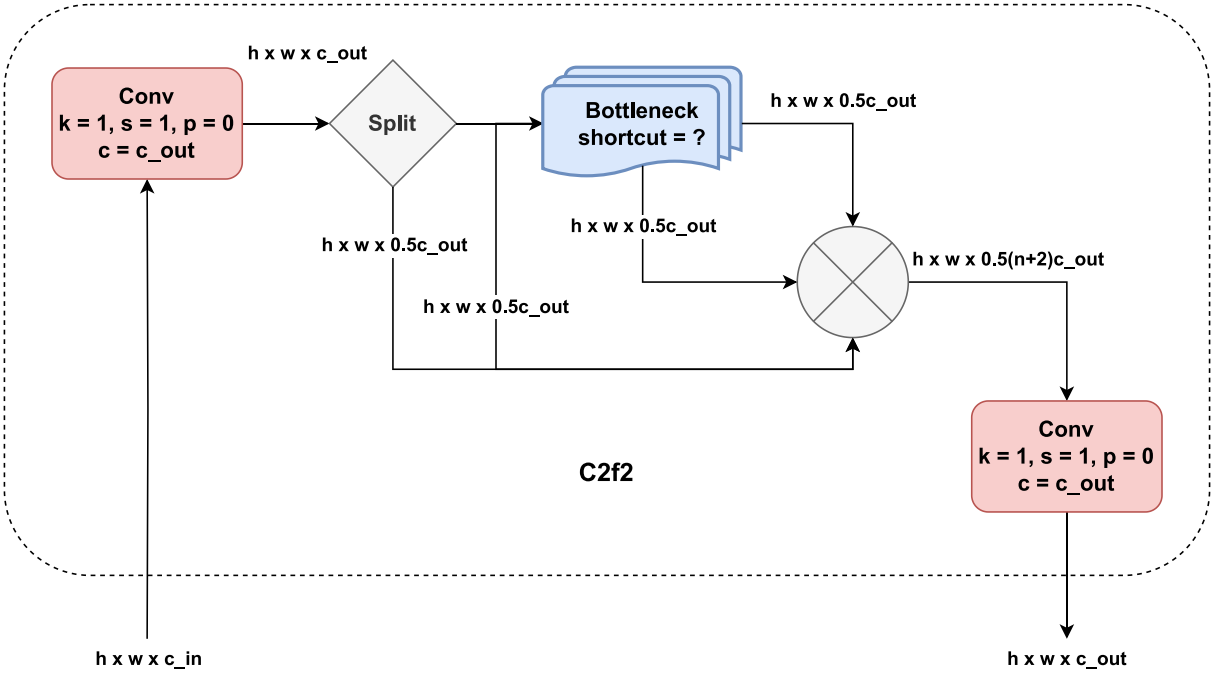


Fig. 6. Proposed C2f2 layer structure showing the number and order of filters.

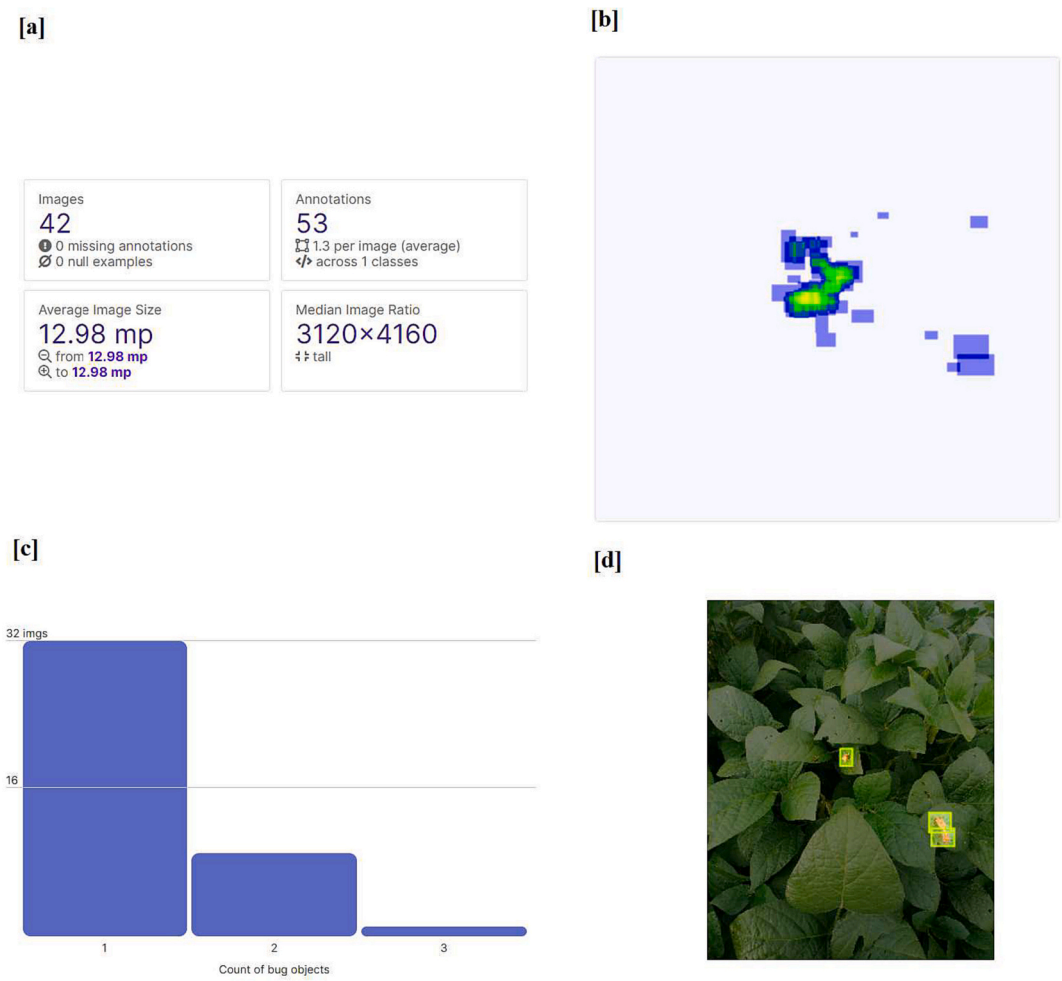


Fig. 7. (a) EMBRAPA_EUSCHISTUS dataset main data; (b) NBSB distribution over scene; (c) Count of insects over images and (d) Sample image from our dataset, comprising scene with NBSB annotated and background only after preprocessing.

different scales. By adding P2, as in Table 1, in a similar way as in YOLOv5 (Benjumea et al., 2021; Zhan et al., 2022), the network is expected to become more profound with more layers of computation. This increased depth can help the network to learn complex representations of the input image better and extract more informative features, which can lead to improved performance in object detection tasks, particularly for small objects.

It is worth noting, however, that increasing the network depth also comes with some potential downsides, such as increased computational complexity and a greater risk of overfitting the model to the training data. So, adding the P2 layer to the YOLOv8 architecture is a tradeoff that needs to be carefully balanced with other considerations, such as model size and performance requirements. In this sense, we proposed in this work a modification in the C2f layer due to its importance in the general architecture of YOLOv8, in the sense of making it lighter and counterbalancing the increase obtained with the addition of the P2 feature level layer. In our work, we call this new layer C2f2.

C2f2 diverges from C2f insofar as they have a different number of filters in the bottleneck blocks but have a similar network structure, Fig. 6. Both implementations utilize the CSP bottleneck block, which commonly incorporates two convolutions found in convolutional neural network architectures. The block consists of a 1×1 convolution layer, a 3×3 depthwise convolution layer, and another 1×1 convolution layer. The input tensor is split into two equal parts, with the first part going through the first 1×1 convolution and being split again. The second part goes through bottleneck modules, which consist of two separable convolution layers and an optional shortcut connection. The two split parts and the outputs of the bottleneck modules are concatenated and passed through the second 1×1 convolution layer to produce the block output. An 'n' parameter determines the number of bottleneck modules; in this case, it is 1.

In this study, ablation experiments are conducted to evaluate the impact of different modules (P2 feature level and C2f2 layer) on the performance of the NBSB object detection algorithm under the same experimental conditions. For that purpose, the new algorithm was trained and tested on the INSECT10K7C640_SAT dataset and compared with A) YOLOv8n, B) YOLOv8n with C2f2 only, and C) YOLOv8n with P2 only. We chose YOLOv8n version 8.0.99 as the baseline model for the ablation experiments. The input image resolution was set to 640×640, and 100 epochs were trained under a batch size 150.

This work used Google Colab, a browser-based coding platform that provides free GPU resources. Specifically, we used the Google Colab Pro+ version, which offers priority access to more powerful GPUs and high-memory virtual machines compared to the free version. We used the NVIDIA A100-SXM4-40GB GPU, specifically, a high-performance GPU with 40GB memory.

2.5. *Testing generalization of the model*

We have also used an unpublished set of NBSB images to test whether the proposed object detection model, trained with images from the INSECT10K7C640_SAT dataset, could generalize under different conditions. Moreover, to test the counting capabilities of the proposed system. This set comprises a total of 42 images of *Euschistus Heros* in soybean

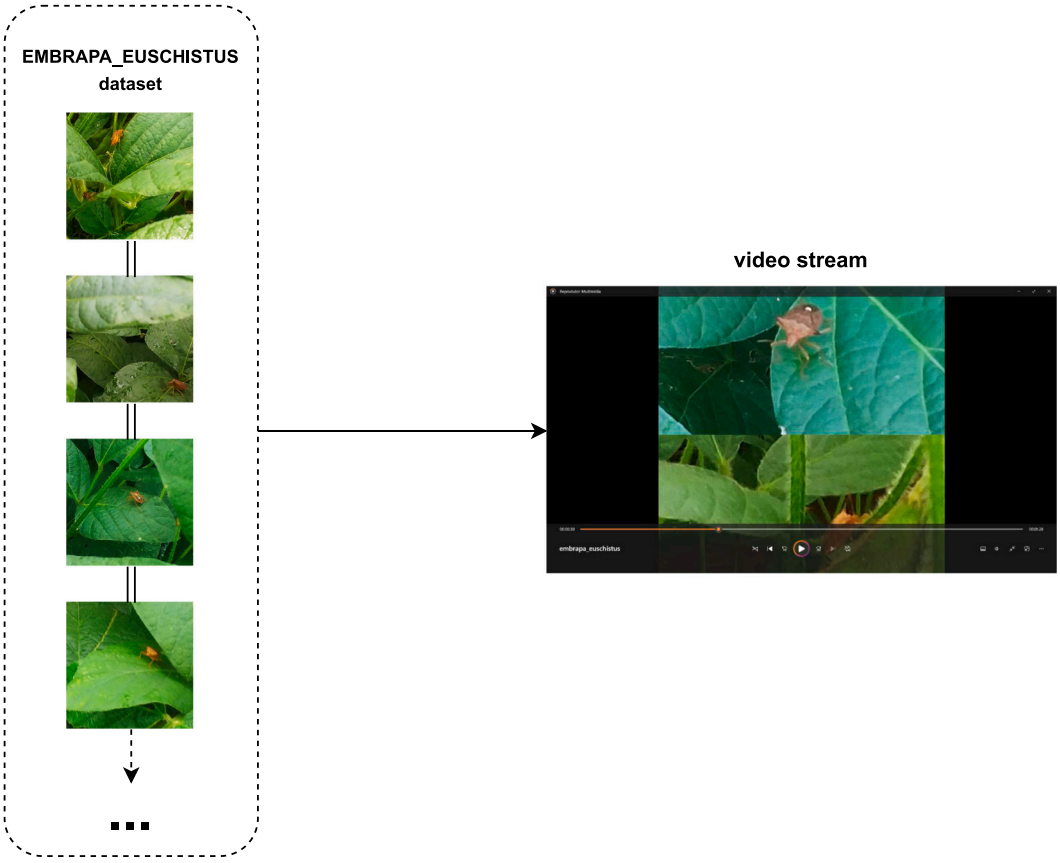


Fig. 8. General scheme of the process of transforming a series of images into a dynamic video mimicking drone footage.

obtained between February and March 2021 in the experimental area of Embrapa in Santo Antônio de Goiás, Goiás, Brazil. The images, which we call in this study EMBRAPA_EUSCHISTUS, Fig. 7, were obtained using the rear camera of a smartphone model G9 - LG. The bugs were found in soybean plants at a 1.2 to 1.5 m height. We used Google Colab NVIDIA A100-SXM4-40GB GPU for object detection on the images. The image size was 640 px, and the models were trained on it.

This new dataset was animated in a video where the counting capacity of the proposed method could also be tested. This approach has several advantages over traditional methods like the beat cloth. It can help farmers target specific areas for pest control measures, optimizing the use of resources and reducing environmental impact.

The compilation of images into a video was facilitated by PowerPoint's slideshow feature, as depicted in Fig. 8. We adjusted the slide transition mode using the "push" effect so that each transition lasted 3 s and, once the transition was complete, it would automatically advance a new slide. The images occupied the entire 1:1 video screen, thus obtaining a video lasting 2 min and 7 s. By leveraging presentation software like PowerPoint, precise control over transition effects and timing allows for seamless, continuous transitions between images, mimicking the fluidity of drone footage. While this approach may not encompass the entirety of drone capabilities, it presents a creative and cost-effective alternative. The video was recorded using the Windows 11 Snipping Tool and saved in mp4 format. With this, the generated video can be easily uploaded to our framework. Finally, the video was divided into frames at a rate of 5 FPS (compatible with the analysis we want to make of each frame) using the ASPOSE web application (ASPOSE, 2024).

Table 2
Comparing algorithm performance in terms of precision, mean average precision at IoU 0.5, mean average precision at IoU 0.95, model parameters, floating-point operations, inference time, and total processing time.

Model	Prec.	mAP0.5	mAP0.95	Par. (M)	Flops (G)	Inf. (ms)	Time (h)
YOLOv8n	84.4	61.5	34.4	3.01	8.1	0.4	0.33
YOLOv8n + C2f2	62.5	56.5	35	1.81	5.1	0.3	0.32
YOLOv8n + P2	77.1	65	39.2	2.92	12.2	0.7	0.33
(Proposed model)	78.3	**71.1**	**38.8**	**1.69**	8.6	0.6	0.33

2.6. Counting method

In the context of counting insects in soybean crops using video streams, the ByteTrack algorithm, introduced by (Zhang et al., 2022), emerges as a valuable tool. Specifically designed for video sequences, ByteTrack efficiently categorizes detection boxes into high and low-score classifications, retaining comprehensive information (Zhang et al., 2022).

Initially, ByteTrack, as covered by (Zhang et al., 2022), establishes connections between tracks and high-scoring boxes, although occasional mismatches can occur, often due to factors such as motion blur or occlusion. In order to address these challenges, the algorithm leverages a pivotal component: the Kalman filter. This filter extrapolates the current state of frames based on prior estimations and continually refines them using real-time observations. This dynamic process guarantees precise tracking of objects over time.

Moreover, ByteTrack's attributes render it particularly advantageous in this scenario. The algorithm achieves high tracking speeds by efficiently distributing computational resources, ensuring real-time monitoring capabilities, according to (Zhang et al., 2022). Consequently, ByteTrack enables rapid and highly accurate enumeration of insects

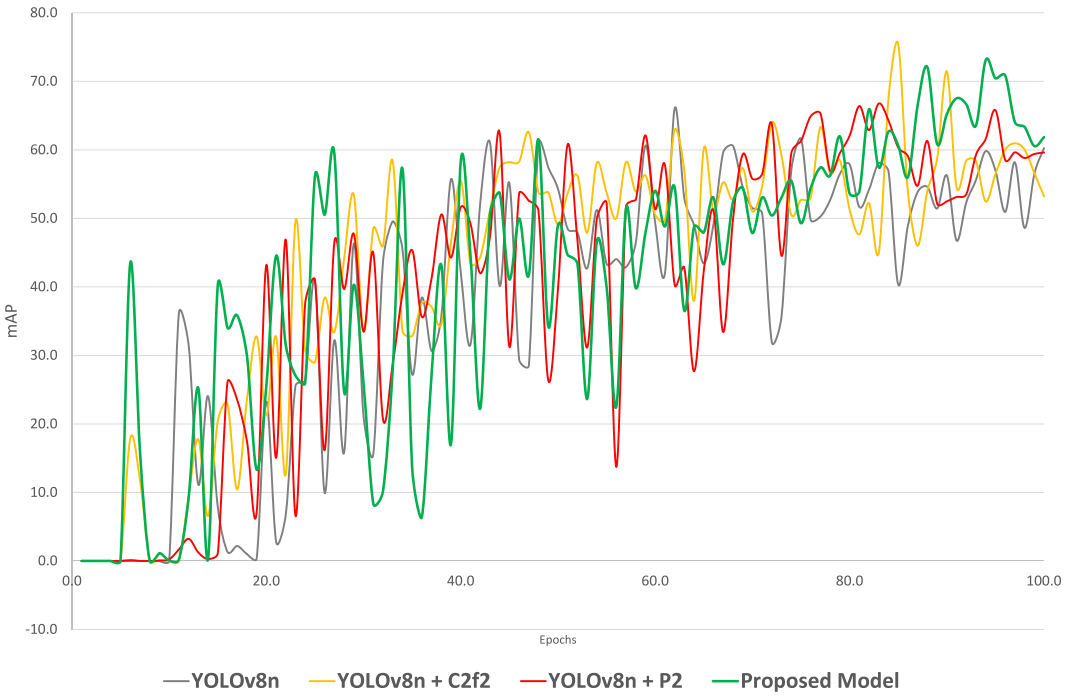


Fig. 9. Mean average precision (mAP) report for the ablation experiment from YOLOv8.

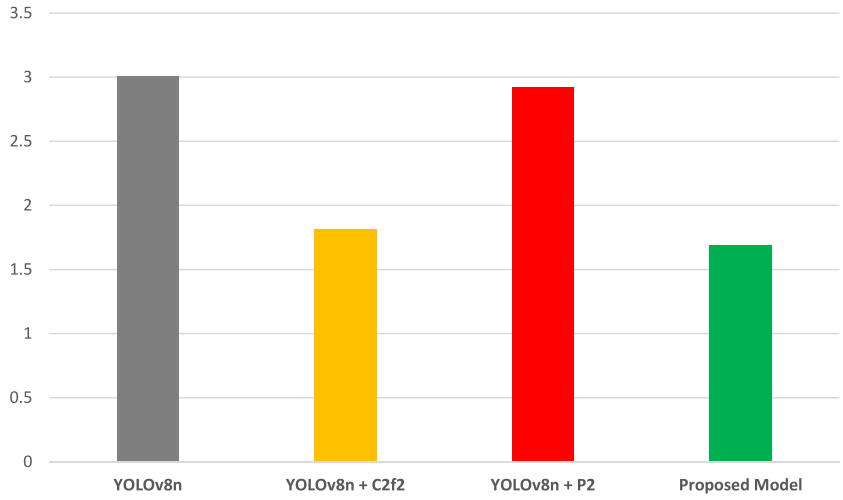


Fig. 10. Number of parameters (Par.) report, in millions, for the ablation experiment from YOLOv8.

within the soybean crop environment using a moving camera video.

Encouraging future research, we readily share our datasets and codes on GitHub.

3. Results and discussion

3.1. Proposed model analysis

Our study compared, as in Table 2, the performance of different object detection algorithms for the given dataset, shown in Fig. 3. Among the algorithms evaluated, YOLOv8n, YOLOv8n + C2f2, YOLOv8n + P2, and the proposed model (YOLOv8n + P2 + C2f2). Adding P2 and C2f2 to YOLOv8n resulted in the highest overall performance among YOLOv8 models.

Compared to YOLOv8n, adding P2 improved the precision and mAP values, especially at higher IoU thresholds. The addition of C2f2, on the other hand, did not significantly improve the performance of YOLOv8n. However, when we added both P2 and C2f2 to YOLOv8n, we observed an improvement in the algorithm's performance, as seen in Fig. 9. The precision and mAP values improved, especially at higher IoU thresholds, indicating that the addition of both features can help the algorithm to better localize and classify objects in images.

Furthermore, adding P2 and C2f2 did not considerably increase model complexity, as seen in the lower number of parameters in Fig. 10 and FLOPs compared to YOLOv8n + C2f2. We also observed that the addition of P2 and C2f2 to YOLOv8n did not result in a noticeable increase in inference time, as seen in the similar values of inference time between YOLOv8n and the proposed model (YOLOv8n + P2 + C2f2). This result suggests that adding P2 and C2f2 simultaneously to YOLOv8n can noticeably improve the algorithm's performance without compromising its speed or model complexity, indicating that the algorithm obtained can perform satisfactorily in real-time tasks, which is crucial for many applications, such as real-time object detection in videos.

Inference on the testing set of the INSECT dataset is covered in

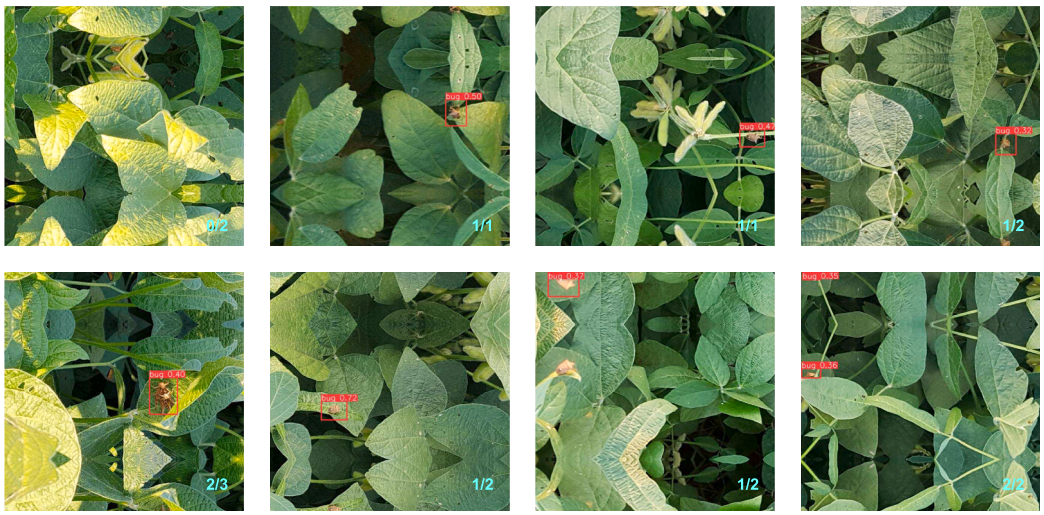


Fig. 11. Samples of NBSB detection with the proposed model on the INSECT10K7C640_SAT dataset (testing set). True positive cases are indicated when the confidence threshold is greater than or equal to 0.3. In the lower right corner, in blue, we present insects detected or present in the image in question. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

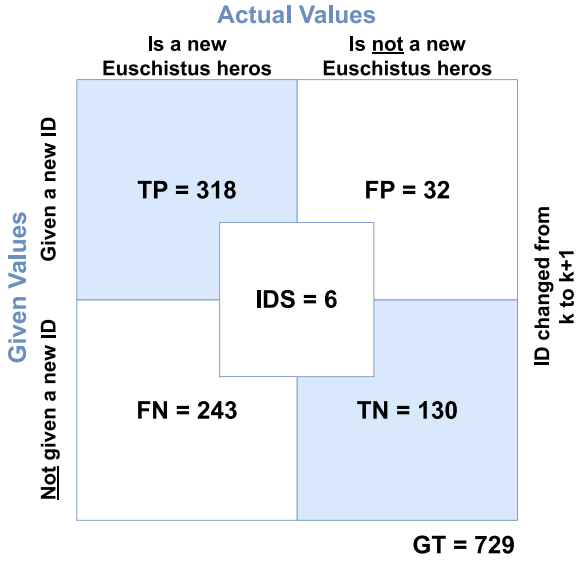


Fig. 12. Confusion matrix exposing the results obtained (TP, FN, FP, TN and IDS), absolute values, in tracking insects along the frames of the provided video stream.

Fig. 11.

3.2. Analyzing NBSB count via video stream

The results of the video insect detection and counting framework are summarized in Fig. 12 and exhibits several positive aspects that warrant recognition.

Regarding false negatives (FN), the proposed framework exhibited a combined FN count of 243, indicating instances where the model failed to detect NBSBs that were present in the ground truth, which can lead to an underestimation of insect populations. While this metric highlights areas for potential improvement in sensitivity, it is important to note that the model detected a significant portion of the ground truth, as evidenced by the true positive (TP) count of 318. In insect detection and counting, having a framework based on high True Positives (TPs) is advantageous for several reasons despite the presence of high False Negatives (FNs). High TPs indicate that the model effectively captures a significant portion of the insect population, providing valuable data for agricultural monitoring.

Firstly, a high TP count ensures that the model accurately represents the proper distribution and density of insects in the environment. This information is crucial for assessing the NBSB population, identifying hotspots of insect activity, and implementing targeted agricultural interventions to mitigate pest damage. Secondly, a system with high TPs instills confidence in the reliability of the model's output, enhancing its utility in decision-making processes as high TPs contribute to the overall reliability and credibility of the data collected.

False positives (FP) were minimized with a count of 32, indicating instances where the model incorrectly identified non-insect objects as such or falsely doubled the count in an NBSB. This low FP rate underscores the model's specificity in distinguishing insects from other environmental elements and in understanding an individual on a trajectory without guessing that it is new, contributing to the overall accuracy of insect counting. Identity switches (IDS), representing cases where the model incorrectly switched the identity of an object across frames, were limited to 6 instances. That demonstrates the model's ability to consistently track individual insects over time, which is crucial for accurately assessing population counting and subsequent mapping.

The multiple object tracking accuracy (MOTA) score, a comprehensive metric considering FN, FP, and IDS, was calculated at 61.45%. This significant MOTA score reflects the model's robustness in maintaining consistent object trajectories despite challenges such as occlusions and changes in lighting conditions. Such a high MOTA score underscores the model's reliability and efficacy in capturing the population of insects when captured on video stream, positioning the model as a promising tool for real-time insect monitoring applications.

Finally, it is worth noting that in the video stream, there were 38 unique adult NBSBs present and that the proposed model (improved YOLOv8 + ByteTrack) was able to track (assign unique IDs on) 40 individuals, demonstrating in the case in question a certain balance between summative and subtractive effects, reaching a value slightly close to the real one (increased by 5.3% of the actual value).

In summary, the results of the video insect detection with improved YOLOv8 and the counting model with ByteTrack, partially shown in Fig. 13, demonstrate the framework's effectiveness in accurately identifying and counting insects on video streams. While there are areas for improvement, such as reducing false negatives and enhancing sensitivity, the framework exhibits promising performance across various metrics, highlighting its potential for applications in agriculture and

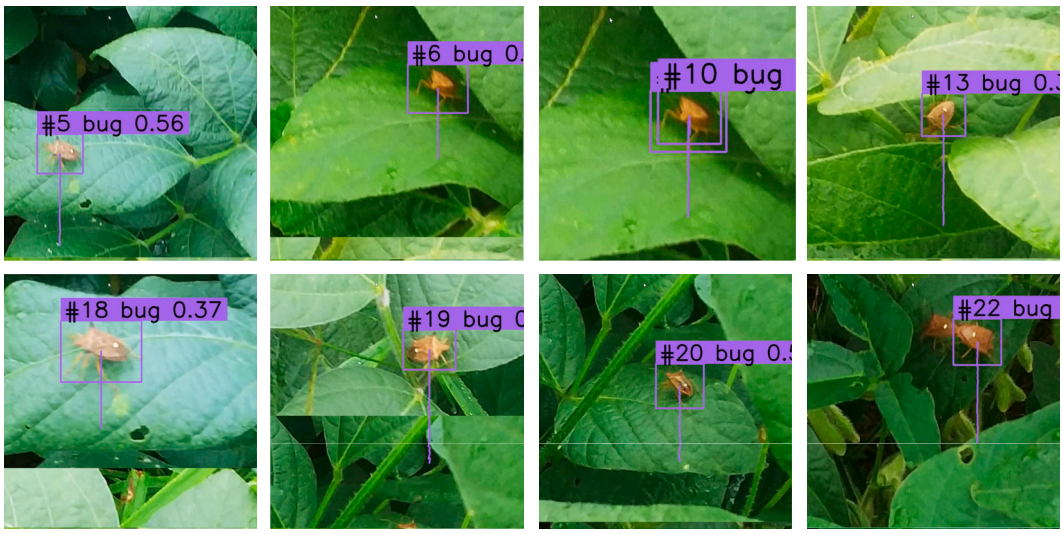


Fig. 13. Samples of NBSB counting using the proposed framework (Improved YOLOv8 model + ByteTrack).

ecological pest management. Continued research and development in this field are critical to further advancing automated insect monitoring techniques.

Insect pest management in cropping systems brings possibilities to improve automation techniques to avoid the undiscriminating application of pesticides. Knowing more precisely the insect species, and its distribution throughout a cultivated area can help more sustainable decisions to be taken in precision agriculture. That can be particularly true in crops cultivated in large areas such as soybeans. Monitoring crops for pest detection using aerial images and artificial intelligence techniques will be a reachable achievements shortly.

The lightweight and enhanced YOLOv8 model presented in this study goes in the direction of closing this gap. Detecting and counting NBSB automatically in soybean crops in tropical areas can be dealt with, and the research opens a path for further improvements and extension to other insect pests. The next steps of this research will collect more image field data and focus on parameter optimization for end devices.

3.3. Limitations of the approach

The specific problem of detecting and counting NBSB through drone-captured images addressed in this research is of great importance for lowering the use of pesticides in soybean crops since the next steps would be to control the pests only in areas with their significant presence. Compared with manual laboring, it would also be a step forward since precision and costs could be achieved satisfactorily in a short period.

Our solution proposed here has shown promising results for the conditions given. However, they should be further tested in more varied weather, lighting, and crop conditions to be fully tuned before deploying in an integrated platform. For the problem addressed, i.e., detecting and counting NBSB, through images with a state-of-the-art solution is a novelty contribution. Regarding the data used, it is one of the largest available, and the results are genuinely competitive. However, in the future, a larger dataset with varied conditions could even be put openly for other researchers to benchmark the newest advances in this type of technology. As hardware keeps advancing at processing capacity, we will see a real-time solution of this kind soon.

A model like this proposal should be trained and tested for each different crop and a set of significant pests for a particular crop, instead of trying to have a general pest detector for crops. Also, other critical sustainable and biological pest management should be considered, with more extensive benefits. The research model proposed here is a competitive solution in the scope of automation.

4. Conclusions

This study introduces a novel method tailored for real-time counting insect pests on soybean fields, made to the YOLOv8 detection model.

By conducting ablation experiments, we evaluated modifications upon YOLOv8 architecture's performance in detecting NBSB pests in soybean crops. Our results demonstrate that the modified YOLOv8 with P2 and C2f2 layers noticeably improved precision and mean Average Precision (mAP) without increasing model complexity or inference time. These enhancements excel at higher Intersection over Union (IoU) thresholds, indicating the algorithm's potential for real-time applications. True positive instances were reliably identified at a confidence threshold of 0.3 or higher.

The evaluation of the video insect detection and counting model (improved YOLOv8 + Bytetrack) reveals promising results for its application in insect control within soybean crops.

The model exhibits promising insect detection and counting performance, as evidenced by the low false positive rate and the limited number of identity switches (IDS). While there is room for improvement in reducing false negatives and enhancing sensitivity, the high true positive (TP) underscores the model's capability to detect insects in the ground truth accurately. Furthermore, the multiple object tracking accuracy (MOTA) score of 61.45% reflects the model's overall accuracy in tracking multiple insects across frames. Furthermore, the proposed model could count (assign IDs on) a number very close to the actual presence in the scene, that is, 40 out of 38 insects (5.3% higher). These results collectively suggest that the model holds substantial potential as a valuable tool in soybean crop management, offering effective insect monitoring and control.

Extended work should aim to carry out tests with more drone video sequences, mainly covering other species of insects and developing an image processing methodology so that each collected frame in the video can be combined with the drone metadata so that the information collected can be mapped into a georeferenced orthomosaic capable of providing valuable information to farmers and researchers in the field.

CRediT authorship contribution statement

Bruno Pinheiro de Melo Lima: Writing – original draft, Visualization, Software, Investigation, Data curation, Conceptualization. **Lurdineide de Araújo Barbosa Borges:** Writing – original draft, Visualization, Validation, Investigation. **Edson Hirose:** Writing – original draft, Visualization, Validation, Investigation. **Díbio Leandro Borges:** Writing – review & editing, Writing – original draft,

Visualization, Validation, Supervision, Methodology, Investigation, Conceptualization.

Declaration of competing interest

None.

Data availability

Data will be made available on request.

Acknowledgments

Authors would like to thank partial support for this research from EMBRAPA (Brazilian Agricultural Research Corporation), CAPES (Coordination of Superior Level Staff Improvement), and UnB (University of Brasilia).

References

Ahmad, I., Yang, Y., Yue, Y., Ye, C., Hassan, M., Cheng, X., Wu, Y., Zhang, Y., 2022. Deep learning based detector yolov5 for identifying insect pests. Appl. Sci. 12, 10167.

ASPOSE, 2024. Convert Video File to Image Online app - Free Online Video File to Image Converter. https://products.aspose.app/video/video-to-image (Accessed on 02/07/2024).

Benjumea, A., Teeti, I., Cuzzolin, F., Bradley, A., 2021. Yolo-z: improving small object detection in yolov5 for autonomous vehicles. arXiv preprint arXiv:2112.11798.

Bernardin, K., Elbs, A., Stiefelhagen, R., 2006. Multiple object tracking performance metrics and evaluation in a smart room environment. In: Sixth IEEE International Workshop on Visual Surveillance, in Conjunction with ECCV. Citeseer.

Bortolotto, O.C., Pomari-Fernandes, A., de Freitas Bueno, R.C.O., de Freitas Bueno, A., da Cruz, Y.K., Sanzovo, A., Ferreira, R.B., 2015. The use of soybean integrated pest management in Brazil: a review. Agronomy Sci. Biotechnol. 1 (1), 25–32. https://doi.org/10.33158/ASB.2015v1i1p25.

Bueno, A., Batistela, M.J., Bueno, R.C.O., de Barros França-Neto, J., Nishikawa, M.A.N., Libério Filho, A., 2011. Effects of integrated pest management, biological control and prophylactic use of insecticides on the management and sustainability of soybean. Crop Prot. 30, 937–945.

Butera, L., Ferrante, A., Jermini, M., Prevostini, M., Alippi, C., 2021. Precise agriculture: effective deep learning strategies to detect pest insects. IEEE/CAA J. Autom. Sin. 9, 246–258.

Cheng, Z., Huang, R., Qian, R., Dong, W., Zhu, J., Liu, M., 2022. A lightweight crop pest detection method based on convolutional neural networks. Appl. Sci. 12, 7378.

Ciampi, L., Zeni, V., Incrocci, L., Canale, A., Benelli, G., Falchi, F., Amato, G., Chessa, S., 2023. A deep learning-based pipeline for whitefly pest abundance estimation on chromotropic sticky traps. Eco. Inform. 78, 102384.

Diwan, T., Anirudh, G., Tembhurne, J.V., 2023. Object detection using yolo: challenges, architectural successors, datasets and applications. Multimed. Tools Appl. 82, 9243–9275.

Høye, T.T., Ärje, J., Bjerge, K., Hansen, O.L., Iosifidis, A., Leese, F., Mann, H.M., Meissner, K., Melvad, C., Raitoharju, J., 2021. Deep learning and computer vision will transform entomology. Proc. Natl. Acad. Sci. 118, e2002545117.

Huang, Y., He, J., Liu, G., Li, D., Hu, R., Hu, X., Bian, D., 2023. Yolo-ep: a detection algorithm to detect eggs of pomacea canaliculata in rice fields. Eco. Inform. 77, 102211.

Kang, J., Zhao, L., Wang, K., Zhang, K., et al., 2023. Research on an improved yolov8 image segmentation model for crop pests. Adv. Comp. Sign. Syst. 7, 1–8.

Kasinathan, T., Singaraju, D., Uyyala, S.R., 2021. Insect classification and detection in field crops using modern machine learning techniques. Inform. Proces. Agric. 8, 446–457.

Kern, A., Bobbe, M., Khedar, Y., Bestmann, U., 2020. Openrealm: Real-time mapping for unmanned aerial vehicles. In: 2020 International Conference on Unmanned Aircraft Systems (ICUAS). IEEE, pp. 902–911.

Khalid, S., Oqaibi, H.M., Aqib, M., Hafeez, Y., 2023. Small pests detection in field crops using deep learning object detection. Sustainability 15, 6815.

Li, W., Zheng, T., Yang, Z., Li, M., Sun, C., Yang, X., 2021. Classification and detection of insects from field images using deep learning for smart pest management: a systematic review. Eco. Inform. 66, 101460.

Li, K., Wang, J., Jalil, H., Wang, H., 2023. A fast and lightweight detection algorithm for passion fruit pests based on improved yolov5. Comput. Electron. Agric. 204, 107534.

Lima, B.P.M., 2023. Insect10k7c640 Sat Dataset. URL. https://universe.roboflow.com/percevejo-marrom/insect10k7c640_sat.

Liu, Z., Gao, X., Wan, Y., Wang, J., Lyu, H., 2023. An improved yolov5 method for small object detection in uav capture scenes. IEEE Access 11, 14365–14374.

Lou, H., Duan, X., Guo, J., Liu, H., Gu, J., Bi, L., Chen, H., 2023. Dc-yolov8: Small Size Object Detection Algorithm Based on Camera Sensor.

Mahaur, B., Mishra, K., 2023. Small-object detection based on yolov5 in autonomous driving systems. Pattern Recogn. Lett. 168, 115–122.

Masuda, T., Goldsmith, P.D., 2009. World soybean production: area harvested, yield, and long-term projections. Int. Food Agribusiness Managem. Rev. 12, 1–20.

Önler, E., 2021. Real time pest detection using yolov5. Int. J. Agric. Nat. Sci. 14, 232–246.

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

Seemakurthy, K., Fox, C., Aptoula, E., Bosilj, P., 2022. Domain generalisation for object detection. arXiv preprint arXiv:2203.05294.

Silveira, F.A.G.D., Tetila, E.C., Astolfi, G., Costa, A.B.D., Amorim, W.P., 2021. Performance analysis of yolov3 for real-time detection of pests in soybeans. In: Intelligent Systems: 10th Brazilian Conference, BRACIS 2021, Virtual Event, November 29–December 3, 2021, Proceedings, Part II. Springer, pp. 265–279.

Tang, Z., Lu, J., Chen, Z., Qi, F., Zhang, L., 2023. Improved pest-yolo: real-time pest detection based on efficient channel attention mechanism and transformer encoder. Eco. Inform. 78, 102340.

Tetila, E.C., 2019. Insect10k7c—Insects Pests Dataset of Soybean Crop. https://bit.ly/2XK1XXv.

Tetila, E.C., Machado, B.B., Astolfi, G., de Souza Belete, N.A., Amorim, W.P., Roel, A.R., Pistori, H., 2020a. Detection and classification of soybean pests using deep learning with uav images. Comput. Electron. Agric. 179, 105836.

Tetila, E.C., Machado, B.B., Menezes, G.V., de Souza Belete, N.A., Astolfi, G., Pistori, H., 2020b. A deep-learning approach for automatic counting of soybean insect pests. IEEE Geosci. Remote Sens. Lett. 17, 1837–1841.

Tian, Y., Wang, S., Li, E., Yang, G., Liang, Z., Tan, M., 2023. Md-yolo: Multi-scale dense yolo for small target pest detection. Comput. Electron. Agric. 213, 108233.

Ultralytics, 2023. New - Yolov8 Multi-Object Tracking Issue #1429 Ultralytics/Ultralytics. https://github.com/ultralytics/ultralytics/issues/1429 (Accessed on 07/21/2023).

Verma, S., Tripathi, S., Singh, A., Ojha, M., Saxena, R.R., 2021. Insect detection and identification using yolo algorithms on soybean crop. In: TENCON 2021-2021 IEEE Region 10 Conference (TENCON). IEEE, pp. 272–277.

Yuan, Z., Fang, W., Zhao, Y., Sheng, V.S., 2021. Research of insect recognition based on improved yolov5. J. Artif. Intell. 3, 145.

Yuan, S., Du, Y., Liu, M., Yue, S., Li, B., Zhang, H., 2022. Yolov5-ytiny: a miniature aggregate detection and classification model. Electronics 11, 1743.

Zhan, W., Sun, C., Wang, M., She, J., Zhang, Y., Zhang, Z., Sun, Y., 2022. An improved yolov5 real-time detection method for small objects captured by uav. Soft. Comput. 26, 361–373.

Zhang, P., Zhong, Y., Li, X., 2019. Slimyolov3: narrower, faster and better for real-time uav applications. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops.

Zhang, Y., Sun, P., Jiang, Y., Yu, D., Weng, F., Yuan, Z., Luo, P., Liu, W., Wang, X., 2022. Bytetrack: multi-object tracking by associating every detection box. In: European Conference on Computer Vision. Springer, pp. 1–21.