www.editada.org

International Journal of Combinatorial Optimization Problems and Informatics, 16(2), April 2025, 112-122. ISSN: 2007-1558.
https://doi.org/10.61467/2007.1558.2025.v16i2.603

Comparative study of Convolutional Neural Networks performance and efficiency with YOLOv8 models applied for pest detection purposes in bean plants

Nantai Nava Nolazco [1,2] *, Marlon Zamora Galván* [1,2] *, Víctor Adrián Macias Martínez* [2] *, Hugo Enrique Orozco García* [2] *, Julio Cesar De Dios García* [1,2]*, Ernesto Monroy Cruz* [3,4]*, Luis Rodolfo García Carrillo* [5]*, Noe Ordaz Rodríguez* [2]

[1] *Universidad Politécnica de Pachuca*
[2] *Centro de Bachillerato Tecnológico industrial y de servicios No. 222*
[3] *Tecnológico Nacional de México Campus Atitalaquia*
[4] *Centro de Estudios Tecnológicos industrial y de servicios No. 26*
[5] *Klipsch School of Electrical and Computer Engineering, New Mexico State University*

E-mails: equipomeca742@gmail.com, jcesarupp93@gmail.com, ernesto.mz@atitalaquia.tecnm.mx, luisillo@nmsu.edu

Abstract. Neural Networks have significantly evolved, particularly in their application to computer vision. This paper presents a comprehensive comparison of different versions of YOLOv8, such as YOLOv8n, YOLOv8s, YOLOv8m, YOLOv8l, and YOLOv8x for the detection of pests in bean plants, leveraging the capabilities of Convolutional Neural Networks. To train the neural network using different versions of YOLOv8, identical conditions were applied, such as the amount of environment light, the number of labeled images, epochs, and batch size. The results indicate that, as the complexity of the YOLO model increases, the training time escalates significantly. This increase corresponds to a more detailed data processing approach in advanced models. The results also provide insight on which model emerges as the most balanced option, offering the highest precision without compromising too much on speed. One of the models achieves the highest precision, making it reliable for accurate object detection but the speed is slow compared with other models. Otherwise, exceptional precision makes it ideal for tasks where accurate identification is critical. The slight reduction in speed does not significantly hinder its overall performance in contexts where precision and detection distance are prioritized.
Keywords: Neural Network, Convolutional Neural Networks, YOLO, Pest detection

Article Info
Received January *17, 2025*
Accepted Mar *26, 2025*

1 Introduction

Agricultural productivity is critical for global food security, and pest management plays a vital role in ensuring healthy crop yields. According to studies conducted in Mexico, there are a series of phytosanitary problems, including a complex group of pests such as the bean conchuela, the whitefly, and the leafhopper (Morales Galvez,2023). Additionally, there are other pests like Sclerotinia sclerotiorum de Bary, a pathogenic organism also known as Whetzelinia sclerotiorum Korf and Bumont, which is known in Mexico as white mold (Santos, A. M.,2023).

At present, diverse technologies are used in precision agriculture for pest control tasks. Among them we can mention, for example, the creation of a protocol for extracting geometric and structural information from the foliar canopy of fruit plantations using 3D point clouds generated by LiDAR sensors and images acquired from unmanned aircraft systems (UASs) (Sandonís-Pozo, 2022). Other companies, such as DJI (DJI, 2020) have developed customized models for crop protection against insects or weeds by carrying out the integration of specific sensors such as high precision RTK GPS, multispectral, and high-resolution cameras. The Parrot Bluegrass Fields is a complete ready to fly UAS solution (Parrot, 2020) equipped with multispectral sensor processing

software, which is suitable for the entire crop analysis workflow. The Parrot Bluegrass Fields software provides farmers with information to maximize their yield and improve the quality of their crops (Del Cerro,2021).

Technologies available for pest control tasks based on detection techniques use various computer vision methods. On one hand, traditional approaches involve the use of a series of image pre-processing operations, such as threshold segmentation, edge detection, and region growth, to extract features such as color, shape, texture, and size from an image. These features are used as a priori knowledge inputs in artificial intelligence algorithms, such as K-nearest neighbor and K-means clustering. In (Lee, 2020) all of the aforementioned studies were adopted for fruit detection of various applications following a pixel-level segmentation approach.

On the other hand, the alternative to achieve detection is to apply recent techniques in artificial vision such as YOLO, which is a real-time object detection and image segmentation model using Artificial Neural Networks (ANNs) (Ultralytics, 2024), which are defined as mathematical models that try to emulate the natural behavior of biological Neural Networks (NNs) (Pascasio, J. 2022). This method allowed the development of more complex NNs with multiple layers of neurons (Jia, W.,2022). The NN architecture can have different configurations, such as Convolutional Neural Networks (CNNs) used for image processing, or Recurrent Neural Networks (RNNs) that are used for sequential tasks. The Convolutional Neural Support Vector Machines Hybrid Classifier (CNSVMHC) is a heterogeneous combination of CNNs and the Support Vector Machines (SVM), where the output layer of the CNN is replaced by an SVM. In (Anguraj, 2021) a scheme based on smart data mining is presented, which shows a solution of an automatic irrigation in agriculture area for water management. Unfortunately, this study does not propose a solution against pests and plant detection. In (Abdullahi, 2017) CNNs are applied for the recognition and classification of plant images, specifically focusing on corn plants. However, it is important to note that YOLO models were not employed, limiting the approach to conventional CNNs techniques for identifying and categorizing corn plants.

This paper focuses on computer vision through implementing CNNs using YOLOv8, in order to perform real-time pest detection in bean plants. YOLOv8 is the latest iteration in the YOLO series of real-time object detectors, offering cutting-edge performance in terms of accuracy and speed. Building upon the advancements of previous YOLO versions, YOLOv8 introduces new features and optimizations that make it an ideal choice for various object detection tasks in a wide range of applications (Ultralytics, 2024). The main contributions of this article are twofold. On one side, we introduce the training methodology description of various CNNs models such as YOLOv8n, YOLOv8s, YOLOv8m, YOLOv8l and YOLOv8x. On the other side, we provide a performance and efficiency comparative analysis of models through obtaining parameters such as training time, frame rate, detection accuracy, and PC resource consumption. As a result, we are able to suggest the CNN model that best fits to solve the described problem.

2 Convolutional Neural Networks (CNNs)

A CNN or ConvNet is a network architecture for Deep Learning (DL) that learns directly from data. In a NN, see Fig. 1, the input layer receives data and has one neuron for each component of the data. This data is passed to one or more hidden layers, which are those layers that are neither the input nor the output layers in the network. It is in the hidden layers that all the processing happens, through a connection system characterized by weights (w) and biases (b).

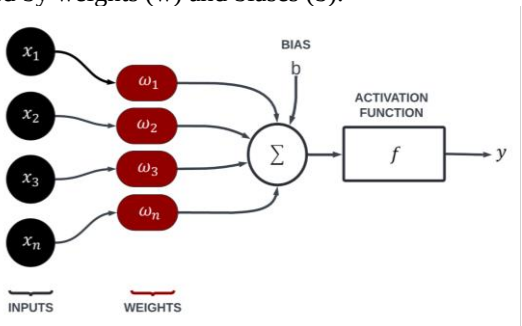

Fig. 1. Diagram to understand the structure of a CNN, showing the input layers, their specific weights, and how they link to the activation function.

With the input value received by the neuron, a weighted sum is calculated by also adding the bias and according to the result and a preset activation function, as follows:

$$\sum \omega_i x_i + b \tag{1}$$

such expression would be the body of the neuron, and then release the activation function as follows:

$$f\left(\sum_{\square}^{\square} \omega_i x_I + b\right) \tag{2}$$

Equation (2) is a mathematical function of the form f(x), and then is added to an ANN to help the network learn complex patterns as follows:

$$y = f\left(\sum_1^n \omega_n x_n + b\right) \tag{3}$$

The result's value, if not constrained, can reach substantial magnitudes, especially in deep neural networks with millions of parameters. This scenario will lead to calculation problems. Therefore, to solve this a sigmoid function is used:

$$f(x) = \frac{1}{1 + e^{-x}} \tag{4}$$

This feature works especially for models where there is a need to predict probability as an outcome. Since the probability of anything exists only between the range of 0 and 1 as shown in Fig. 2, the detection would be faster and the data would be easier to read to collect data.

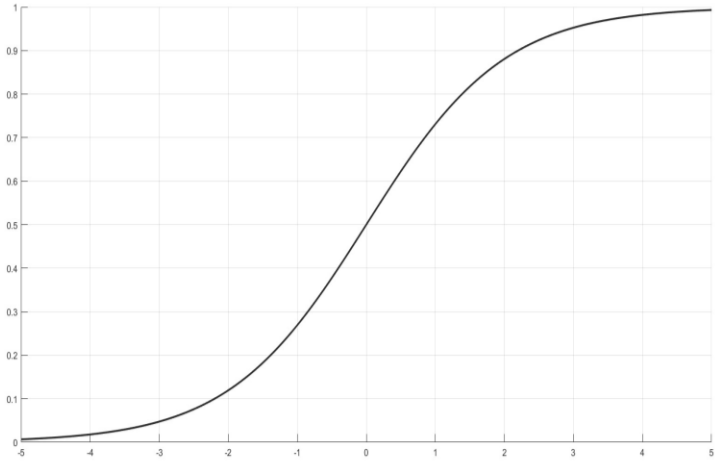

Fig. 2. Plot of the sigmoid function applied as an activation function within the neural network for data reduction, taken values from 0 to 1.

3 Experimental implementations

In (Ultralytics, 2024) several YOLO models are shown, ranging from YOLOv3 to YOLOv10. The models 9 and 10 are models that are still in process, so this work focuses on implementing and analyzing the last one functional version, i.e., the YOLOv8 model. The YOLOv8 series offers a wide range of models, each specialized for specific tasks in machine vision. These models are designed to meet a variety of requirements, from object detection to more complex tasks such as instance segmentation, pose/keypoint detection, oriented object detection, and classification. These models are denoted as follows:

- YOLOv8: Deteccion
- YOLOv8-seg: Instance Segmentation
- YOLOv8-pose: Pose/Keypoints
- YOLOv8-obb: Oriented Detection
- YOLOv8-cls: Classification

In this work, the YOLOv8 model for detection is addressed. For this model there are five subversion files, which are differentiated by the termination and they will be compared in terms of performance for pest detection in bean plants.

- **YOLOv8n**
- **YOLOv8s**
- **YOLOv8m**
- **YOLOv8l**
- **YOLOv8x**

The methodology carried out in this research starts from the selection of images and their labeling, and ends with the performance comparison of each CNN model under equal conditions. In the following sections the stages involved are described.

3.1 Selection and Labeling

At this stage, photos were taken and images were selected, with exactly 800 RGB images of 640 x 480 pixels to be labeled according to the plant conditions. Fig. 3 presents some examples of these acquired images.

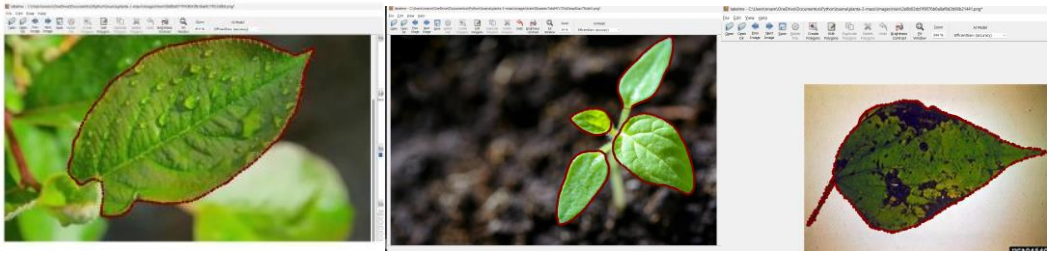

Fig. 3. Example of the type of images that were selected and labeled ROI (Region Of Interest) for neural network training.

3.2 Training Models

All networks were trained with the same rules and under equal conditions. The following hyperparameters were used for training stage:

- Data labeled: It is the number of labeled objects of a class that will be processed in training.
- Number of epochs: An epoch is a complete pass back and forth of all training examples.
- Batch: Number of images processed simultaneously in a forward pass.

For all the models, i.e., YOLOv8n, YOLOv8m, YOLO8s, YOLOv8l, and YOLOv8x, 800 labeled data were specified with a single class, a training of 100 epochs, and a batch of 2 images.

In this stage, the training time parameter of each model was taken in order to initialize the model's comparison. Fig. 4 shows all the obtained results. At this point, the training time parameter of each network gives us an approach about how complex each model is, and an expectation about its operation. It is possible to observe a shorter time for the YOLOv8n model and a longer time for the YOLOv8x model.

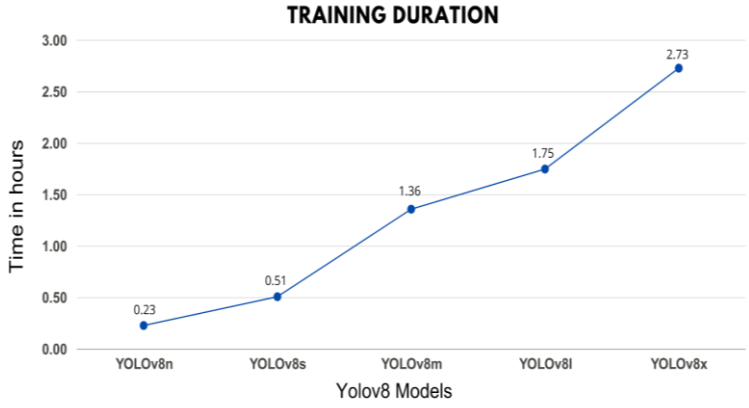

Fig. 4. Training time required in hours for different YOLOv8 models which increase progressively with the complexity of the model.

It is important to mention that for these and the following measurements of NN models parameters, a mid-range laptop with the following features was used:

- Model: Dell G15 5510
- Graphics card: Nvidia 3050 RTX
- Processor: 1ntel(R) Core(TM) i7-10870H CPU @ 2.20GHz[Cores 8] [Logical processors 16]
- RAM: 8Gb
- Display Memory (Vram): 8Gb
- DirectX 12

3.3 Running Models Metrics

The same video of a plant was used for analyzing all the CNNs models. The first aspect to measure is about performance, i.e., the Frames per second (FPS) achieved when the network is running and the resources used by the computer within the video. The specific parameters to be analyzed are the consumption of the CPU, GPU, Vram, and RAM. The second aspect to measure is the detection operation of the networks through the following parameters: Precision, Accuracy, Completeness, Confidence Detection, and the True Detection Distance before the network starts to fail.

A confusion matrix was used to analyze detection effectiveness, this matrix is shown in Table 1 and is defined as follows:

TP: When it detects and if it is correct
FP: When it detects and should not detect
TN: When it does not detect and if it is correct
FN: When it does not detect and should detect

Table 1. Definition of a confusion matrix for detection effectiveness, where the rows represent predicted values and the columns represent actual values.

	POSITIVE	**NEGATIVE**
POSITIVE	TRUE POSITIVE: **TP**	FALSE POSITIVE: **FP**
NEGATIVE	FALSE NEGATIVE: **FN**	TRUE NEGATIVE: **TN**

To extract the above information, the video was analyzed every two frames, counting the detections, TP, TN, FP and FN, and also the precision, accuracy and completeness were calculated with the following formulas:

Accuracy: Metric that allows to calculate the overall performance of the class:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (5)$$

Precision: Metric that quantifies the number of correct predictions made:

$$Precision = \frac{TP}{TP+FP} \quad (6)$$

Completeness: Metric that quantifies the number of correct predictions that could have made:

$$Completeness = \frac{TP}{TP+FN} \quad (7)$$

The last two parameters were measured with the same video, taking them out of the same network and measuring the distance at which the network was still detecting in an efficient way.

Confidence Detection: An amount between 0 and 1 that represents the security of the same network when detecting the desired object as can be seen in Fig. 5.

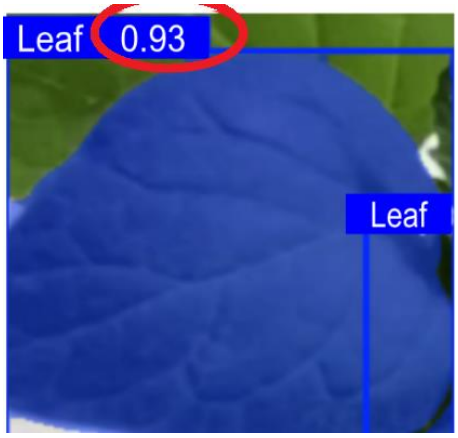

Fig.5. Confidence detection score of 0.93, indicating the model's high confidence in its prediction.

4 Results and Comparative Analysis

4.1 Results about performance aspect

To see the fluidity of the video, FPS were measured during the test. This parameter can be associated with the detection of each model. The higher the FPS are, the faster the detection. This helps to have a faster and more efficient future diagnosis. Results in Fig. 6. show that YOLOv8n is faster than all of the others.

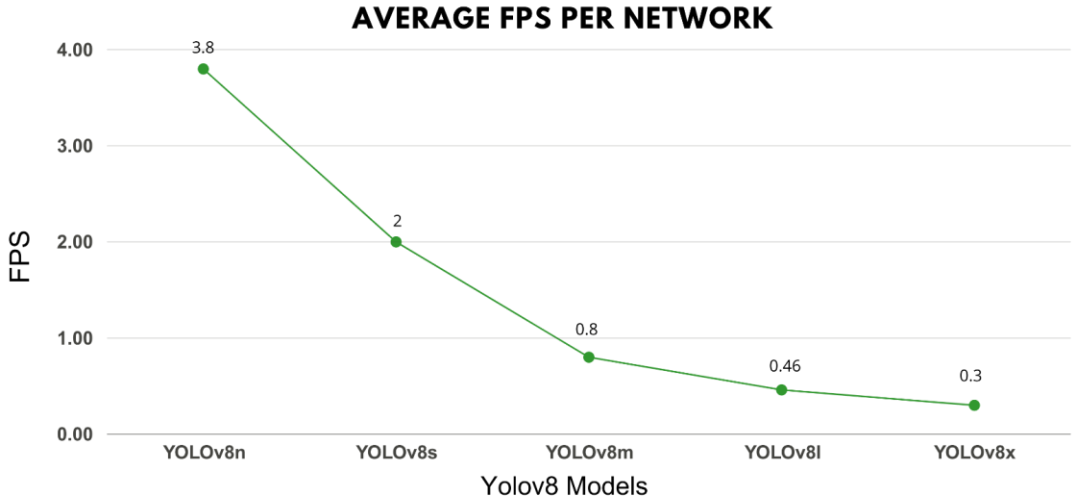

Fig. 6. Average frames per second (FPS) performance comparison which demonstrates the computational efficiency of each model variant.

The results of the complementary performance parameters can be seen in Fig. 7. These values show the consumption of computer resources while each model is running. As observed, the RAM, VRAM, and GPU usage increase with the model version. This indicates that as the network models advance, more video and graphics resources are required for detection, which is the reason for the decrease in FPS

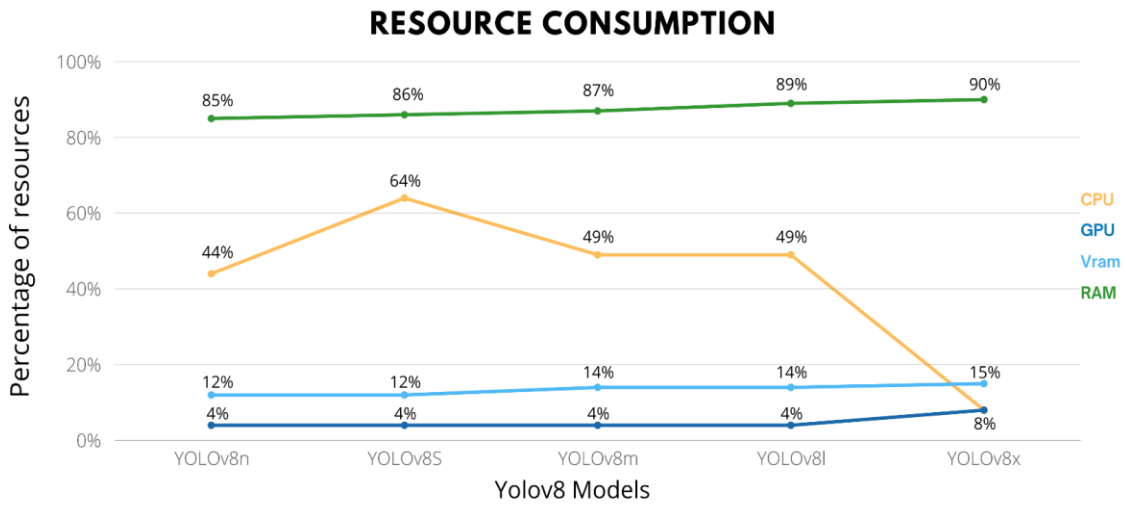

Fig. 7. Computer resources consumed (%) when testing each neural network during the test.

4.2 Results about detection operation aspect

The following results are about the detection efficiency of each NN. Table 2 shows the confusion matrix corresponding to each model.

YOLO	MODEL	POSITIVE	NEGATIVE
POSITIVE	**YOLOv8n**	1781	81
	YOLOv8s	2028	2
	YOLOv8m	1768	23
	YOLOv8l	1828	102
	YOLOv8x	1801	160
NEGATIVE	**YOLOv8n**	709	686
	YOLOv8s	1326	686
	YOLOv8m	876	686
	YOLOv8l	684	686
	YOLOv8x	594	636

Table 2. Confusion matrix showing the accuracy and reliability of each YOLOv8 model in detecting positive and negative cases, highlighting the differences in detection performance across the models.

The parameter quantified for detection is distance. This involves running the video and recording the data on the point at which the neural network begins to fail. The results are shown in meters and percentages, where one meter equals one hundred percent. Based on this, the parameter could be called the maximum prediction or detection distance. Figure 8 shows the results for distance, as well as accuracy, precision, and completeness.

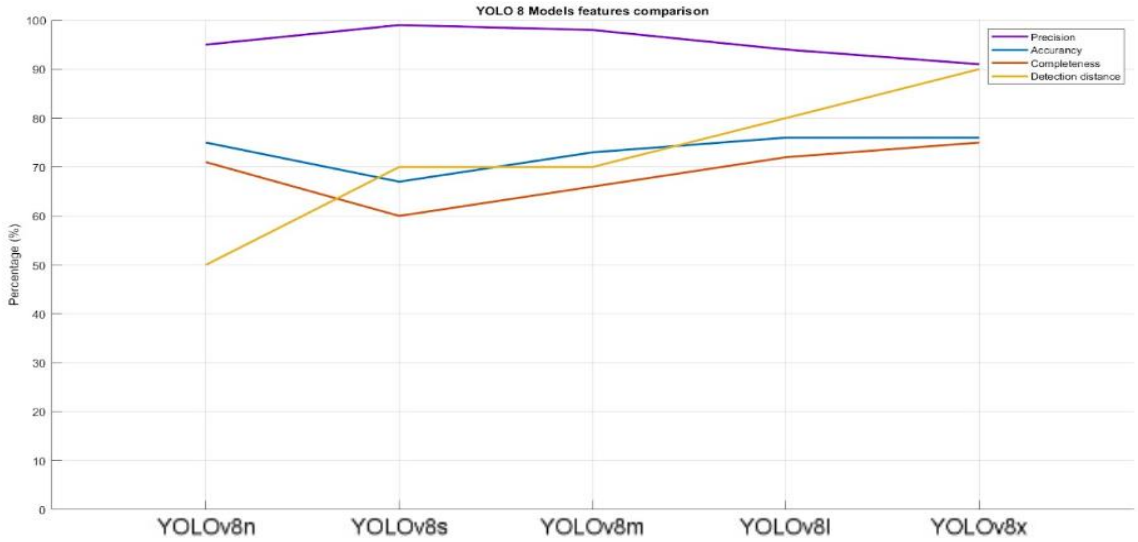

Fig. 8. Confusion Matrix Results, Precision, Accuracy, Completeness, and Detection distance (%) for each CNN model.

4.3 Comparative Analysis

To determine which model is the best in terms of confidence detection, several tests were performed using videos with healthy and infected leaves.

The parameter that indicates the certainty of the detected element is the Confidence. Results of confidence percentages when applying the different YOLOv8 models are presented in Figure 9. These tests were obtained from a video of two healthy bean plant leaves. The video used to collect these data is available at: https://drive.google.com/file/d/1f7_7JUG94NpE50xfb5bcYSk8ItKTcZMs/view?usp=sharing

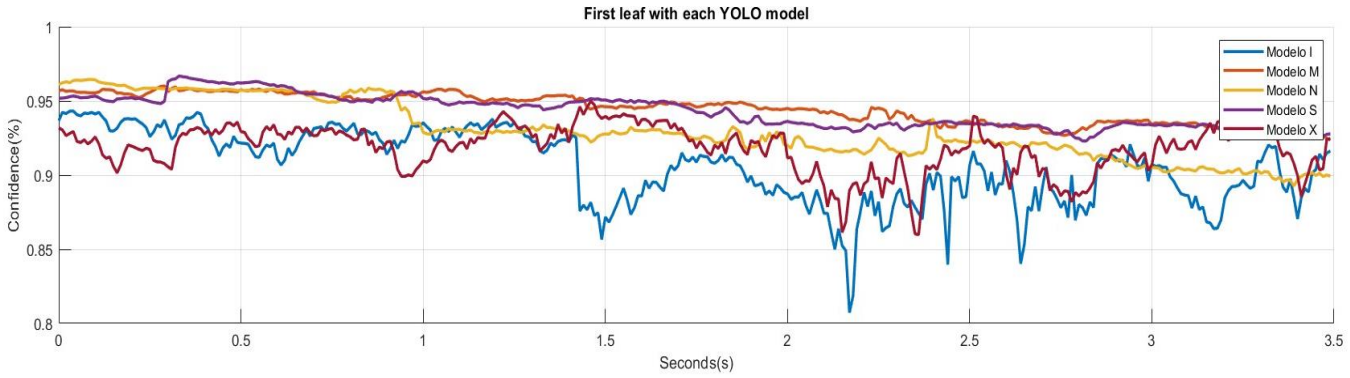

Fig. 9 .Results of the detection confidence parameter for each model when implementing YOLOv8 models.

As can be seen in Figure 9, all models have a confidence percentage between 70% and 96%. However, it is clear that the S model and the M model demonstrate greater stability in their parameters. Specifically, the S model in the graph for the first leaf shows accuracy between 94.6% and 96.6%.

On the other hand, Figure 10 presents the confidence percentages when the models are implemented for detecting a leaf infested with the red spider mite, where necrosis is observed. The video used to perform this test is available at: https://drive.google.com/file/d/16MCeMD07TO4jwJAGYNDgcbmFBuLu8dE0/view?usp=sharing

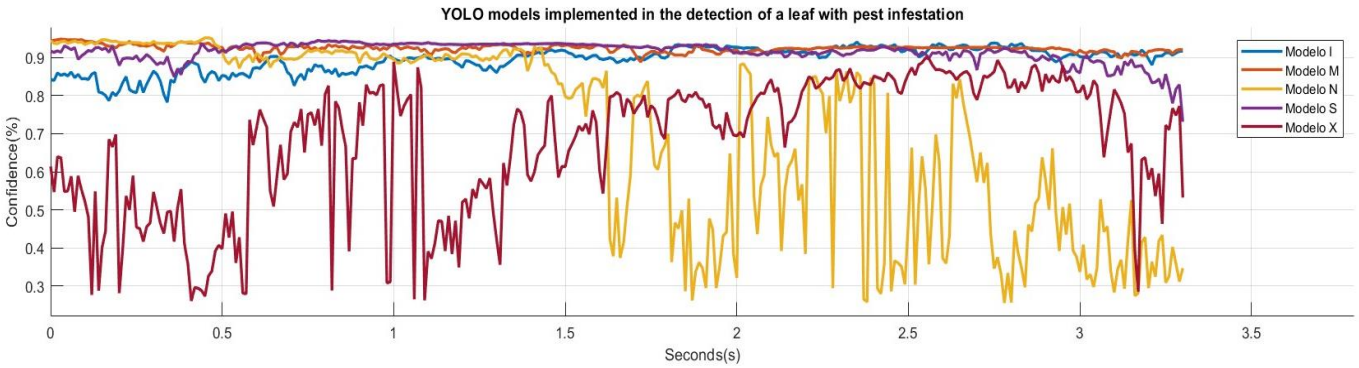

Fig. 10.Results of the detection confidence parameter for each model on an infested leaf.

According to Figure 10, the S and M models are more stable. The S model shows values between 84% and 94%, and the M model shows values between 88% and 93%. In contrast, the X and N models have the lowest average confidence parameters, although they do show some stability in parts of the graph.

5 Discussions

To determine which model is the best in terms of video speed and detection accuracy, we will evaluate the models using the following parameter summary:

Frames per Second (FPS): Measures the model's processing speed.
Precision: Indicates the proportion of true positives over the total number of positive predictions.
Accuracy: Reflects the proportion of true positives and true negatives over the total number of cases.
Completeness: Indicates the proportion of true positives over the total number of actual positive cases.
Maximum Detection Distance: Indicates the model's ability to detect objects at a certain distance.
Highlighting the following points:

Model n has a higher FPS (3.8) and also provides good precision (95%), accuracy (75%) and completeness (71%), making it a strong contender for scenarios where speed is more critical, though it has slightly lower precision and a shorter detection range (0.5 m).

Model m offers high precision (98%) and a decent detection range (0.7 m) with an accuracy (73%) and completeness (66%) but has a lower FPS (0.8), making it less suitable for applications where both high speed and precision are needed.

Model l has a respectable precision (94%) and the highest accuracy (76%) and recall (72%), with a good detection distance (0.8 m). However, its FPS (0.46) is lower, which might impact its performance in real-time applications.

Model x has the lowest FPS (0.3) and precision (91%), but excels in accuracy (76%) and recall (75%), along with the best detection range (0.9 m). Despite its strong performance in other areas, its low FPS may limit its effectiveness in scenarios requiring faster processing.

Models thus emerges as the most balanced option, offering the highest precision (99%), without compromising too much on speed FPS (2), accuracy (67%) and completeness (60%), making it ideal for applications that require both high detection accuracy and reasonable processing speed, although it has a good detection distance (0.7 m).

On the other hand, the most stable models in terms of confidence parameters are the "m" and "s" models. These models also have the best average parameters, making them important to consider. It is worth noting that the remaining models produced confidence parameters that are still workable, even though they are lower.

Finally, considering the most balanced option in terms of performance and detection aspects, the CNN model suggested for pest detection in bean plants is YOLOv8s.

6 Conclusions and future work

Determining which neural network is the most efficient based on speed, precision, accuracy, completeness, and detection distance is crucial to optimizing its implementation in specific applications. Evaluating these factors helps identify the model that offers the best performance where one or more of these criteria are a priority.

In this paper, the comparison of parameters between YOLOv8 versions proves that YOLOv8 models achieves the highest precision at 99%, making it reliable for accurate object detection. Its speed is relatively low at 2 FPS, which is slower compared to other models, such as model n (3.8 FPS) so models is 47.2% slower than model n. The Accuracy of YOLOv8s is moderate at 67%, and its Completeness is slightly lower at 60%. However, it compensates for these limitations with a strong detection distance of 0.7 meters, which is noteworthy for applications requiring reliable range detection.

Despite its lower speed, YOLOv8s's exceptional precision makes it ideal for tasks where accurate identification is critical. The slight reduction in speed does not significantly hinder its overall performance in contexts where precision and detection distance are prioritized.

YOLOv8s model offers a well-rounded performance when balancing speed, precision, accuracy, completeness, and detection distance. Its selection as the ideal model will ultimately depend on the specific requirements of the application, prioritizing one or more of these evaluation criteria.

As future work, evaluation of the model's performance under various environmental conditions, including different lighting scenarios and weather conditions (rain, fog, snow), is proposed to ensure robust operation across diverse operational settings. The study will also explore the adaptation of the best performing model through transfer learning techniques, enabling detection over different types of ground vehicles in order to obtain major mobility of the system.

References

Abdullahi, H. S., Sheriff, R., & Mahieddine, F. (2017, August). Convolution neural network in precision agriculture for plant image recognition and classification. In 2017 Seventh International Conference on Innovative Computing Technology (INTECH) (Vol. 10, pp. 256-272). New York: Ieee.

Anguraj, D. K., Mandhala, V. N., Bhattacharyya, D., & Kim, T. H. (2021). Hybrid neural network classification for irrigation control in WSN based precision agriculture. Journal of Ambient Intelligence and Humanized Computing, 1-12.

Badillo, F. L., Hernández, C. A. R., Narváez, B. M., & Trillos, Y. E. A. (2021). Redes neuronales convolucionales: un modelo de Deep Learning en imágenes diagnósticas. Revisión de tema. Revista colombiana de radiología, 32(3), 5591-5599.

Del Cerro, J., Cruz Ulloa, C., Barrientos, A., & de León Rivas, J. (2021). Unmanned aerial vehicles in agriculture: A survey. Agronomy, 11(2), 203.
DJI. Drones for Agriculture. Available online: https://ag.dji.com/es?site=brandsite&from=nav (accessed on 10 October 2020).

Jia, W., Sun, M., Lian, J., & Hou, S. (2022). Feature dimensionality reduction: a review. Complex & Intelligent Systems, 8(3), 2663-2693.

Juan, R. Q., Mario, C. M. (2011). Redes neuronales artificiales para el procesamiento de imágenes, una revisión de la última década. RIEE & C, Revista de Ingeniería Eléctrica, Electrónica y Computación, 9(1), 8-9.

Lee, J., Nazki, H., Baek, J., Hong, Y., & Lee, M. (2020). Artificial intelligence approach for tomato detection and mass estimation in precision agriculture. Sustainability, 12(21), 9138.

Martínez Llamas, J. (2018). Reconocimiento de imágenes mediante redes neuronales convolucionales.

Morales Galvez, G. M. (2023). Identidad y actividad patogénica del hongo entomopatógeno aschersonia hypocreoidea (cooke y massee) petch contra trialeurodes vaporariorum Westwood en Phaseolus vulgaris L.

Parrot. Parrot Bluegrass Fields. Available online: https://atyges.es/tienda/en/parrot-bluegrass-fields/ (accessed on 10 October 2020).
Pascasio, J. J., Mela, R. H., Vélez, M. L., & Rangel, J. C. (2022). Implementación de redes neuronales para la clasificación de desechos dentro de un cesto inteligente. Centros: Revista Científica Universitaria, 11(1), 229-245.

PlanetScope Vegetation Indices to Estimate UAV and LiDAR– derived Canopy Parameters in a Super–Intensive Almond Orchard. Frutic 14th International Symposium, 29 Jun – 1 Jul, Valencia. http://hdl.handle.net/10459.1/84009

Ultralytics. (2024, 7 January). UltralyticsRetrieved May 26, 2024, from https://docs.ultralytics.com/es.

Sandonís Pozo, L., Plata Moreno, J. M., Llorens Calveras, J., Escolà i Agustí, A., Pascual Roca, M., & Martínez Casasnovas, J. A. (2022). PlanetScope Vegetation Indices to Estimate UAV and LiDAR-derived Canopy Parameters in a Super-Intensive Almond Orchard.

Santos, A. M., González, A. M., De Dios Alche, J., & Santalla, M. (2023). Microscopical Analysis of Autofluorescence as a Complementary and Useful Method to Assess Differences in Anatomy and Structural Distribution Underlying Evolutive Variation in Loss of Seed Dispersal in Common Bean. Plants, 12(11), 2212.