EDUCACIÓN TECNOLÓGICO NACIONAL DE MÉXICO
IPSUMTEC ISSN: 2594 - 2905

LOCATION OF LEAVES AND FRUITS IN TOMATO PLANTS USING YOLO8VN

LOCATION OF LEAVES AND FRUITS IN TOMATO PLANTS USING YOLO8VN

Carranza Flores José Luis[1], Magadán Salazar Andrea[1], Cristóbal Alejo Jairo[2], Fuentes-Pacheco Jorge[1]

[1] National Technological Institute of Mexico/National Center for Research and Technological Development (CENIDET),d23ce167@cenidet.tecnm.mx , https://orcid.org/0000-0002-0235-8685.
[1] National Technological Institute of Mexico/National Center for Research and Technological Development (CENIDET),andrea.ms@cenidet.tecnm.mx , https://orcid.org/0000-0001-5474-4150.
[2] National Technological Institute of Mexico/CONKAL,jairo.ca@conkal.tecnm.mx , https://orcid.org/0000-0001-9354-1129. ([1]) National Technological Institute of Mexico/National Center for Research a n d Technological Development (CENIDET),jorge.fp@cenidet.tecnm.mx , https://orcid.org/0000-0002-9060-8276.

https://doi.org/10.61117/ipsumtec.v8i2.383

Abstract -- The localization of objects of interest using deep learning is an area of computer vision that focuses on identifying and delimiting the location of specific objects within an image. This article presents a proposal to automatically locate the minimum rectangle of tomato leaves and fruits by analyzing images with Yolo8vn. The results obtained show good performance even in the face of factors such as changes in perspective, scale, lighting, dust, pathologies, etc.

Keywords: deep learning, CNN, YOLO8vn, tomato leaves, tomato fruit.

Abstract -- Locating objects of interest with deep learning is an area of computer vision that focuses on identifying and delineating the location of specific objects within an image. This article presents a proposal to locate the minimum rectangle of tomato leaves and fruits automatically, through image analysis with Yolo8vn. The results obtained show good performance even in the face of factors such as the presence of changes in perspective, scale, lighting, presence of dust, pathologies, etc.

Key words – deep learning, CNN, YOLO8vn, tomato leaf, tomato fruit.

INTRODUCTION

According to the United Nations, by 2050 there will be 9.7 billion people inhabiting the planet [1], which means that the demand for food will increase dramatically. Therefore, the agricultural sector must make use of technologies to support decision-making in the care of agricultural crops, giving rise to a new area called precision agriculture [2]. Precision agriculture relies on advances in artificial intelligence in the areas of computer vision, machine learning de and deep learning , through which the design and development of intelligent systems that automate processes that are time-consuming for humans is carried out. Tomatoes are the most widely consumed vegetable in the world. In Mexico, it is one of the most important vegetables due to the number of direct and indirect jobs generated by its cultivation and the amount of foreign currency that enters the country from its commercialization [3]. In addition to its economic importance, tomatoes are also a source of vitamins, minerals, and antioxidants, which are essential for human nutrition and health. The presence of diseases in their cultivation causes large losses or malformations of the fruit. That is why it is proposed to speed up the process of recognizing pathologies from early stages to support farmers in making decisions to prevent their spread and the indiscriminate use of pesticides and fungicides.

In the literature on this subject, there is a particular interest in tomato cultivation due to its importance in the global market. The main activities carried out are the identification of symptoms on the leaves to determine the disease and/or the causative pathogen (virus, bacteria, fungi), as in [4-13]. It is noteworthy that few studies consider the recognition of pathologies in the fruit, as in [14-17]. This is because viruses, bacteria, fungi, and pests first affect the leaves of plants and, once they have caused damage, they also appear on stems and fruits.

It is also observed that the reported systems detect various diseases in agricultural crops, improving their yield through the use of deep learning architectures: AlexNet [18], GoogleNet [19-20], ResNet50 [21], ResNet18 [22], Xception [23], VGG16 [24-27]. In most of them

, pathogen detection is carried out by analyzing images under controlled conditions with advanced damage levels. That is, they are images of a cropped (segmented) leaf, aligned and contrasting with the background, as in [22], [24], [28-32]. However, these conditions do not always occur in a real environment, where it is important to identify symptoms at an early stage so that measures can be taken to eliminate the pathogens. In order to recognize the presence of the first symptoms in plants, the first thing that is needed is a system that can locate or segment the leaves and fruits of tomato plants under real conditions.

This paper describes the methodology for locating tomato leaves and fruit, considering that images acquired under uncontrolled conditions may contain several leaves at the same time with different perspectives and sizes, occlusion between them, shadows, dust, water, changes in light intensity, changes in leaf color due to disease symptoms, etc. These factors increase the complexity of the segmentation process and are therefore rarely considered. For its development, the proposed system uses a deep learning approach through the implementation, training, and optimization of the YOLO8vn network [34].

The article is organized as follows: the theoretical framework of the YOLO8vn architecture and the metrics used to evaluate the system's performance are presented. The methodology followed for the development of the proposed system is detailed. In addition, the experimentation carried out is presented and the results are discussed, and finally, the conclusions of the work are given.

YOLO ARCHITECTURE

Since its creation in 2015, the YOLO (*You Only Look Once*) variant of object detectors has grown rapidly, with YOLO8vn being released in January 2023. YOLO variants are based on the principle of real-time performance and high classification, based on limited but efficient computational parameters [35]. This principle has been found in the DNA of all versions of YOLO with increasing intensity, as the variants evolve to address automated quality inspection requirements within of domain of detection of defects of industrial surfaces, such as the need for rapid detection, high precision, and deployment in restricted areas.

The literature shows that the YOLO family reports good results in object localization.

various challenges. However, these architectures have been considered for leaf localization in different crops, as can be seen in [6], [11], [27], [33], [34]. However, it is noteworthy that most articles report performing localization by analyzing images under controlled conditions. For example, in works [4-26], they report satisfactory performance under controlled conditions; that is, it is necessary to have images in which the main object (leaf or fruit) is cropped, aligned, located, and even has a contrasting background. YOLO8vn [36] is a state-of-the-art computer vision model created by *ultralytics.* It is specifically designed to be lightweight and efficient, ideal for devices with limited resources such as mobile phones, IoT devices, and other embedded applications. In addition, it is optimized to be much smaller than its predecessors. The model contains ready-to-use support for object detection, classification, and segmentation tasks. YOLO8vn comes with both architectural and developer experience improvements. Compared to previous versions, YOLO8vn comes with:

1. New anchor detection system
2. Changes to the convolutional blocks used in the model
3. Mosaic augmentation applied during training, disabled before the last 10 epochs

In addition, YOLO8vn comes with changes to improve the developer experience with the model, and the model is packaged as a library that you can install in Python code.

Metrics

Evaluation metrics allow you to measure the performance of a predictive model. In other words, their purpose is to provide an objective picture of how the model is performing. The classic metrics for evaluating model performance are:

a) *Recall* (Sensitivity or true positive rate): this is the proportion of true positives among all cases that are actually positive. It measures the proportion of positive instances that were correctly identified by the model, see expression 1.

$$Recall = \frac{TP}{TP+FN} \quad \text{Ec. (1)}$$

b) *Precision*: The proportion of true positives among all instances that have been predicted as positive. Expression 2 shows the calculation of *precision*.

$$Precision = \frac{TP}{TP+FP} \quad \text{Ec. (2)}$$

c) *F1 Score*: this is the harmonic mean of *precision* and *recall*. It provides a balance between both metrics and is useful when there is an imbalance in the classes. Expression 3 shows the calculation of the aforementioned metric.

$$(F1)Score = 2 * \frac{Precision * Recall}{Precision + Recall} \quad \text{Ec. (3)}$$

DEVELOPMENT

Methodology

In general, the proposal to locate leaves and fruits in images of tomato plants acquired in real conditions using the YOLO8vn network consists of the following steps, which are detailed in the following subsections (see Figure 1).

a) The set of images used to train and validate the system is acquired and defined.
b) The images are normalized to a single format and size.
c) *Ground truth* is required for training, which is created through a labeling process.
d) The operations for data augmentation are defined and applied.
e) The images are randomly distributed in an 80:10:10 ratio for the subsets to be used in training, validation, and testing.
f) During network training, the parameters are optimized, resulting in predictions of bounding boxes at different scales and positions. Subsequently, a confidence threshold is applied to refine the bounding boxes. To do this, the model is trained with the parameters described in the training section, see Table 4.
g) Finally, the network provides the final bounding boxes with labels and scores, see the experimentation section.

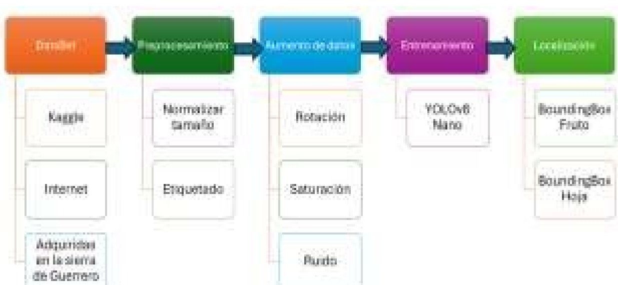

Figure 1. *Schematic of the localization process.*

DataSet

There are several public datasets that are
consist of images of healthy and diseased leaves from various crops. In particular, the most widely used datasets containing tomato leaves are: Plantvillage [37], ImageNet [38], and Kaggle [39]. It is important to note that most of the images in the first two datasets are taken under controlled conditions; that is, the object of interest is
displayed on a white background or a uniform color so that there is no noise obstructing the objects of study. In addition, two of the aforementioned datasets only contain images of leaves from different crops and do not feature fruit

The Kaggle tomato leaf disease dataset contains some images with a white background, but most of the photographs were taken in an uncontrolled environment, as shown in Figure 2. For this reason, we chose to use the Kaggle dataset and selected the images necessary for the development of the project. The dataset contains 10,000 images of various sizes, showing different types of plant diseases, including bacterial infections, fungal diseases, pest infestations, and viral infections.

Figure 2. *Images from the Kaggle dataset [4].*

In order to obtain a broader and more diverse set of images, a visit was made to tomato greenhouses in the state of Guerrero, where photographs were taken (as shown in Figure 3) using three different devices.

Figure 3. *Example of images acquired in the mountains of Guerrero, Mexico.*

IPSUMTEC
ISSN: 2594 - 2905

Finally, the dataset was supplemented with images downloaded from the internet, both of healthy fruits and those with diseases. Some examples are shown in Figure 4.

Figure 4. *Images downloaded from the internet.*

Table 1 shows the number of images captured in greenhouses in Guerrero, from the Kaggle dataset, and those downloaded from the internet.

Table 1. *Number of images in the dataset created.*

Own		*Kaggle*		Internet	
Leaf	Fruit	Leaf	Fruit	Leaf	Fruit
400	400	800	-	100	100
Total: 1800 [1300 leaves and 500 fruits]					

Preprocessing

a) Normalization:

Since images come from various sources, it is necessary to normalize them. To do this, the image format was set to jpg, and the size of the input image was modified. Some of them were 1266 x 1390, others 4224 x 5632, and others 3468 x 4624. All of them were resized to 640 x 640 pixels.

b) Labeling:

The labeling process consists of locating the minimum box containing the object of interest, that is, assigning the position of the upper left point and the lower right point so that the network learns to identify the objects. This process is essential to obtain results that ensure greater accuracy in locating objects. There are different ways to do this. Manual labeling involves humans observing the images and assigning the positions manually.

Semi-automatic labeling involves the use of algorithms to suggest positions, which are then reviewed and corrected by humans. Automatic labeling uses image processing algorithms to locate objects.

Roboflow [40] was used for the labeling process, and manual adjustments were made to some images where the minimum rectangle was not correct. In addition, labels are assigned to images of leaves and fruits, in this particular case of the tomato plant, as shown in Figure 5. During the assignment process, two categories were created: class 0, which refers to fruits, and class 1, which refers to leaves. Subsequently, the three subsets are generated randomly: training, validation, and testing.

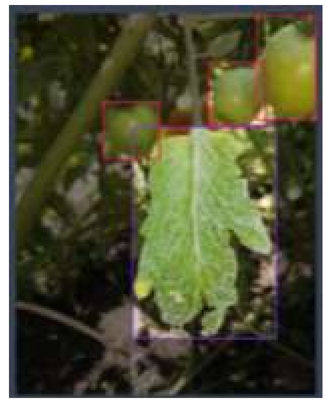

Figure 5. *Image labeled in RoboFlow.*

- The first set provides the input data for training the YOLO8vn network.
- The second group of images allows the parameters to be adjusted during each epoch of the algorithm training
  algorithm, in order to optimize its accuracy.
- The third group allows the model to be evaluated once training is complete. This set is used to
  the final results of the automatic detection reported.

The data is exported for training into three folders containing the images and labels.

Data Augmentation.

Considering that convolutional neural networks require large amounts of images, it was decided to apply some transformations to obtain a greater amount of information and avoid overfitting. Table 2 shows the techniques implemented. Finally, 3,119 images were obtained.

Table 2. *Techniques applied for data augmentation.*

Technique	Value
Rotation	90° upside down
Saturation	34
Gaussian noise	1.96
Resizing	640 x 640

After data augmentation. The dataset is distributed with 80% for training, 10% for validation, and 10% for testing, as shown in Table 3.

Table 3. *Distribution of the image set.*

Number of images	Training 80%	Validation 10%	Testing 10%
3119	2730	150	150

Training

The implementation was carried out using the Google Colab tool for network training and Pycharm 2024.2.3 for testing. The training process consists of iterative learning and evaluation, since the objective of the model is to locate the object of interest within the image. To start training, it is necessary to configure some basic parameters. After a series of training sessions, it was observed that the model performed well with the following values, see Table 4.

Table 4. *Hyperparameters used.*

Hyperparameters	Value
Batch size	64
Ephocs	20
Width shape and height shape	640 x 640

DISCUSSION AND ANALYSIS OF RESULTS

The objective of the experiment was to evaluate the performance of the model in automatically locating the minimum bounding box containing the object of interest, in this case, classifying whether it is a leaf or a fruit, as they subsequently require different analysis.

Figure 6 visually shows the results obtained by enclosing the objects of interest in a *bounding box*, despite the fact that the images, having been acquired in uncontrolled environments, show both healthy and diseased leaves and fruits. It can even be seen in these images that the system is capable of segmenting leaves and fruit despite the scale, perspective, age of the leaves, overlap between them, even in the presence of dust, and problems with tone and shape.

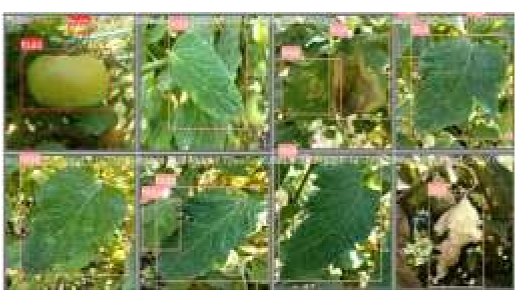

Figure 6. *Results obtained.*

The model performs well in the validation phase, achieving an accuracy of 97.6% (see Figure 7), a recall of 97% (see Figure 8), and an F1 score of 84.28% (see Figure 9). With regard to accuracy, this means that all the fruits and leaves detected by the model are indeed true.

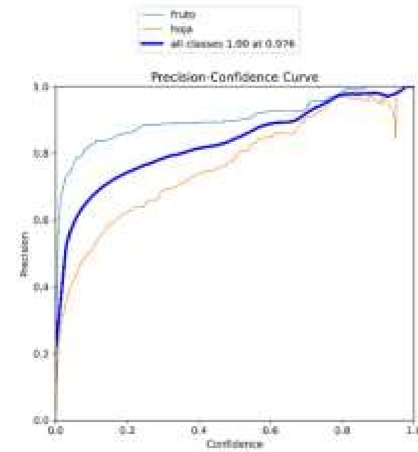

Figure 7. *Graph of the confidence curve in accuracy.*

The recall results tell us that, of all the existing fruits and leaves, the model identifies most of the instances in the image.

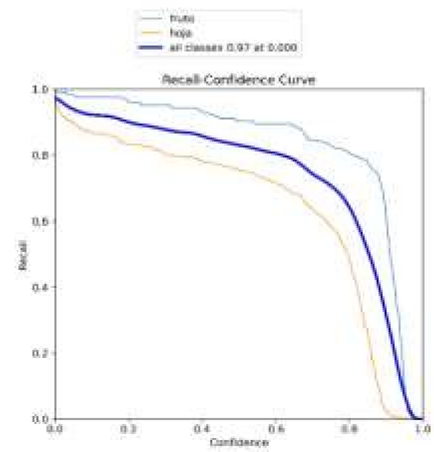

Figure 8. *Graph of the recall confidence curve.*

The F1 results show that the model classifies correctly, but there are some objects (leaves and fruits) that are not located because most of the images were captured in uncontrolled environments and some of them have some occlusions.

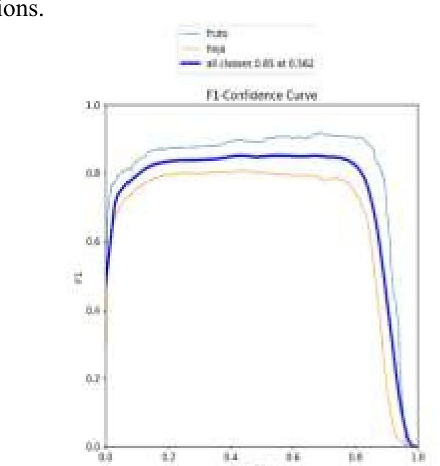

Figure **9**. *F1 confidence curve graph.*

As can be seen in the network learning, there are some localization problems in images containing leaves and fruits in the background. In addition, some images in the test set were out of focus.

CONCLUSIONS

Based on the scores obtained, it can be concluded that the model obtained with YOLO8vn allows for accurate localization in images captured in uncontrolled environments. Furthermore, the use of labeling tools is effective in this particular case because the *bounding box* completely encloses the object of interest. The experiment shows that the YOLO8vn network is a good option for locating leaves and fruits. Furthermore, applying data augmentation provides a greater quantity and diversity of images, which will help to achieve better performance.

This work has presented initial results that demonstrate the feasibility of using computer vision and deep learning techniques for the classification of fruits and leaves. However, the findings represent only the first step toward a more comprehensive system for recognizing plant pathologies. Therefore, the following is proposed as future work:

1. Expand the Data Set: Include a greater number of images covering light conditions and field scenarios, as well as various stages of development and levels of infection.

2. Incorporate Specific Pathologies: Extend the methodology to address the diagnosis of common diseases in the plants analyzed, so that the model can differentiate between healthy leaves and leaves with specific pathologies.

These proposals seek not only to improve the accuracy and efficiency of the model presented, but also to advance toward practical applications that contribute to the control and prevention of plant pathologies, benefiting the agricultural sector and crop sustainability.

BIBLIOGRAPHY

[1] United Nations, 2022. [Accessed August 8, 2024] Available at: https://www.un.org/es/global-issues/population.

[2] Hernández, R. R. (2021). Precision agriculture. A current necessity. *Agricultural Engineering Magazine*, *11*(1), 67-74.

[3] Institute for Agricultural, Aquaculture, and Forestry Research and Training ICAMEX, (2023). Tomato Cultivation.

[4] Jia, W., Xu, Y., Lu, Y., Yin, X., Pan, N., Jiang, R., & Ge, X. (2023). An accurate green fruits detection method based on optimized YOLOX-m. *Frontiers in Plant Science*, *14*, 1187734.

[5] Singh, A. K., Sreenivasu, S. V. N., Mahalaxmi, U. S. B. K., Sharma, H., Patil, D. D., & Asenso, E. (2022). Hybrid feature-based disease detection in plant leaf using convolutional neural network, Bayesian optimized SVM, and random- -forest classifier. *Journal of Food Quality*, *2022*, 1-16.

[6] Jia, W., Xu, Y., Lu, Y., Yin, X., Pan, N., Jiang, R., & Ge, X. (2023). An accurate green fruits detection method based on optimized YOLOX-m. *Frontiers in Plant Science*, *14*, 1187734.

[7] Peng, D., Li, W., Zhao, H., Zhou, G., & Cai, C. (2023). Recognition of tomato leaf diseases based on DIMPCNET. *Agronomy*, *13*(7), 1812.

[8] Liang, J., & Jiang, W. (2023). A ResNet50-DPA model- -for- -tomato- -leaf- -disease-identification. *Frontiers in Plant Science*, *14*, 1258658.

[9] Chen, H., Wang, Y., Jiang, P., Zhang, R., & Peng, J. (2023). LBFNet: A Tomato Leaf Disease Identification Model Based on Three-Channel Attention Mechanism and Quantitative Pruning. *Applied Sciences*, *13*(9), 5589.

[10] Debnath, A., Hasan, M. M., Raihan, M., Samrat, N., Alsulami, M. M., Masud, M., & Bairagi, A. K. (2023). A Smartphone-Based Detection System for Tomato Leaf Disease Using EfficientNetV2B2 and Its Explainability with Artificial Intelligence (AI). *Sensors, 23(*21), 8685.

[11] Umar, M., Altaf, S., Sattar, K., Somroo, M. W., & Sivakumar, S. (2023). Multi-Disease Recognition in Tomato Plants: Evaluating the Performance of CNN and Improved YOLOv7 Models for Accurate Detection and Classification.

[12] Flores Colorado, O. E., Cervantes Canales, J., García-Lamont, F., & Ruiz Castilla, J. S. (2023). Identification of the main diseases of the coffee plant (*Coffea arabica*) through artificial vision. CIENCIA ergo-sum, 30(3).

[13] Fadhilla, M., & Suryani, D. (2023). Android Application for Tomato Leaf Disease Prediction Based on MobileNet Fine-tuning. *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, *7*(6), 1260-1267.

[14] Islam, M. M., Talukder, M. A., Sarker, M. R. A., Uddin, M. A., Akhter, A., Sharmin, S., ... & Debnath, S. K. (2023). A deep learning model for cotton disease prediction using fine-tuning with smart web application in agriculture. *Intelligent Systems with Applications*, *20*, 200278.

[15] Khatoon, S., Hasan, M. M., Asif, A., Alshmari, M., & Yap, Y. (2021). Image-based automatic diagnostic system for tomato plants using deep learning. *Comput. Mater. Contin*, *67*(1), 595-612.

[16] Khattak, A., Asghar, M. U., Batool, U., Asghar, M. Z., Ullah, H., Al-Rakhami, M., & Gumaei, A. (2021). Automatic detection of citrus fruit and leaves diseases using deep neural network model. *IEEE access*, *9*, 112942-112954.

[17] Zeng, Q., Sun, J., & Wang, S. (2023). DIC-Transformer: interpretation of plant disease classification results using image caption generation technology. *Frontiers in Plant Science*, *14*.

[18] Da Silva Abade, A., de Almeida, A. P. G., & de Barros Vidal, F. (2019). Plant Diseases Recognition from Digital Images using Multichannel Convolutional Neural Networks. In *VISIGRAPP (5: VISAPP)* (pp. 450-458).

[19] López, J. A. M., De la Torre Gutiérrez, H., & López, F. J. H. Detection of urban anti-spaces using YOLO: Mexicali case study.

[20] Mohanty, S. P., Hughes, D. P., & Salathé, M. (2016). Using deep learning for image-based plant disease detection. *Frontiers in plant science*, *7*, 1419.

[21] Sagar, A., & Dheeba, J. (2020). On using transfer learning for plant disease detection. *BioRxiv*, 2020-05.

[22] Zhao, S., Peng, Y., Liu, J., & Wu, S. (2021). Tomato leaf disease diagnosis based on improved convolution neural network by attention module. *Agriculture*, *11*(7), 651.

[23] Chipuli, J. P., & Luwemba, G. W. (2023). Tomato Plant Leaf Disease Detection Using Image Recognition: A Case Study of Mlali in Morogoro Region, Tanzania. *European Journal of Information Technologies and Computer Science*, *3*(4), 18-25.

[24] Rangarajan, A. K., Purushothaman, R., & Ramesh, A. (2018). Tomato crop disease classification using pre-trained deep learning algorithm. *Procedia computer science*, *133*, 1040-1047.

[25] Latif, G., Abdelhamid, S. E., Mallouhy, R. E., Alghazo, J., & Kazimi, Z. A. (2022). Deep learning utilization in agriculture: Detection of rice plant diseases using an improved CNN model. *Plants*, *11*(17), 2230.

[26] Jung, M., Song, J. S., Shin, A. Y., Choi, B., Go, S., Kwon, S. Y., ... & Kim, Y. M. (2023). Construction of deep learning-based disease detection model in plants. *Scientific Reports*, *13*(1), 7331.

[27] Liu, Y., & Yu, Q. (2024). Real-time and lightweight detection of grape diseases based on Fusion Transformer YOLO. *Frontiers in Plant Science*, *15*, 1269423.

[28] Fuentes, A., Yoon, S., Kim, S. C., & Park, D. S. (2017). A robust deep-learning-based detector for real-time tomato plant diseases and pests recognition. *Sensors*, *17*(9), 2022.

[29] Fuentes, A. F., Yoon, S., Lee, J., & Park, D. S. (2018). High-performance deep neural network-based tomato plant diseases and pests diagnosis system with refinement filter bank. *Frontiers in plant science*, *9*, 1162.

[30] Gonzalez-Huitron, V., León-Borges, J. A., Rodríguez-Mata, A. E., Amabilis-Sosa, L. E., Ramírez-Pereda, B., & Rodríguez, H. (2021). Disease detection in tomato leaves via CNN with lightweight architectures implemented in Raspberry Pi 4. *Computers and Electronics in Agriculture 181, 105951.*

[31] Bhujel, A., Kim, N. E., Arulmozhi, E., Basak, J. K., & Kim, H. T. (2022). A lightweight Attention-based convolutional neural networks for tomato leaf disease classification. *Agriculture*, *12*(2), 228.

[32] Ullah, Z., Alsubaie, N., Jamjoom, M., Alajmani, S. H., & Saleem, F. (2023). EffiMob-Net: A Deep Learning-Based Hybrid Model for Detection and Identification of Tomato Diseases Using Leaf Images. *Agriculture*, *13*(3), 737.

[33] Mbouembe, P. L. T., Liu, G., Park, S., & Kim, J. H. (2023). Accurate and fast detection of tomatoes based on improved YOLOv5s in natural environments. *Frontiers in Plant Science*, *14*.

[34] Bochkovskiy, A., Wang, C. Y., & Liao, H. Y. M. (2020). Yolov4: Optimal speed and accuracy of object detection. *arXiv preprint arXiv:2004.10934*.

[35] Yolo. [Accessed on 12 from May from 2024]
Available at: docs.ultralytics.com/es/models/yolov8/

[36] Yolo8vn. [Accessed May 12, 2024]
https://github.com/ultralytics/ultralytics?tab=readm e-ov-file

[37] Plantvillage. [Accessed October 12, 2024]
Available at: https://plantvillage.psu.edu/

[38] ImageNet. [Accessed October 12, 2024]
Available at: https://www.image-net.org/

[39] Kaggle. [Accessed May 12, 2024]
Available at: https://www.kaggle.com/datasets

[40] Roboflow. [Accessed May 12, 2024] Available at: https://roboflow.com/

Contribution Role Table

Role	Author(s)
Conceptualization	Jairo Cristóbal Alejo
Methodology	Andrea Magadán Salazar
Administration of the project	Andrea Magadán Salazar, Jorge Fuentes-Pacheco

Software	Jose Luis Carranza Flores
Validation	Jorge Fuentes-Pacheco
Writing	Jose Luis Carranza Flores, Andrea Magadán Salazar