Foods and Raw Materials. 2025;13(2)

ISSN 2308-4057 (Print)
ISSN 2310-9599 (Online)

Research Article
Open Access

Available online at http://jfrm.ru/en
https://doi.org/10.21603/2308-4057-2025-2-650
https://elibrary.ru/OVLNHL

UAV imagery, advanced deep learning, and YOLOv7 object detection model in enhancing citrus yield estimation

Mohamed Jibril Daiaeddine*, Sara Badrouss, Abderrazak El Harti, El Mostafa Bachaoui, Mohamed Biniz, Hicham Mouncif

Sultan Moulay Slimane University, Beni Mellal, Morocco

* e-mail: mohamedjibril.daiaeddine@usms.ma

Received 04.01.2024; Revised 14.04.2024; Accepted 07.05.2024; Published online 18.10.2024

Abstract:
Accurate citrus fruit yield and estimation is of utmost importance for precise agricultural management. Unmanned aerial vehicle (UAV) remote-sensing systems present a compelling solution to this problem. These systems capture remote-sensing imagery with both high temporal and spatial resolution, thus empowering farmers with valuable insights for better decision-making. This research assessed the potential application of UAV imagery combined with the YOLOv7 object detection model for the precise estimation of citrus yield.

Images of citrus trees were captured in their natural field setting using a quadcopter-mounted UAV camera. Data augmentation techniques were applied to enhance the dataset diversity; the original YOLOv7 architecture and training parameters were modified to improve the model's accuracy in detecting citrus fruits.

The test results demonstrated commendable performance, with a precision of 96%, a recall of 100%, and an *F*1-score of 97.95%. The correlation between the fruit numbers recognized by the algorithm and the actual fruit numbers from 20 sample trees provided the coefficient R^2 of 0.98.

The strong positive correlation confirmed both the accuracy of the algorithm and the validity of the approach in identifying and quantifying citrus fruits on sample trees.

Keywords: Agricultural management, unmanned aerial vehicle (UAV), remote-sensing systems, YOLOv7 object detection model, crop yield estimation

Please cite this article in press as: Daiaeddine MJ, Badrouss S, El Harti A, Bachaoui EM, Biniz M, Mouncif H. UAV imagery, advanced deep learning, and YOLOv7 object detection model in enhancing citrus yield estimation. Foods and Raw Materials. 2025;13(2):242–253. https://doi.org/10.21603/2308-4057-2025-2-650

INTRODUCTION

Crop yield estimation plays a crucial role in effective crop management, enabling farmers to make informed decisions regarding harvesting, transportation, storage, and marketing of their produce. Traditional fruit counting methods, while commonly used, are inherently labor-intensive, time-consuming, and prone to human error; moreover, they often give a higher margin of error than expected [1]. Consequently, fruit farming needs new efficient and automated approaches to crop yield estimation. Automated methods reduce the burden of manual labor while enhancing the accuracy and reliability of yield forecasts.

The recent progress in computer technology, camera capabilities, and image analysis have given rise to a diverse array of fruit count methods [2].

Numerous studies feature image processing techniques and machine learning algorithms in the domain of fruit detection and recognition. Sengupta & Lee harnessed a combination of support vector machines, Canny edge detection, Hough transform, and scale-invariant feature transform, along with the majority voting algorithm, to effectively discern citrus fruits from the background [3]. Maldonado & Barbosa based their approach on the extraction of relevant features from green fruits [4]. The method consisted of a series of steps including color model conversion, thresholding, histogram equalization, spatial filtering with Laplace and Sobel operators, and Gaussian blur. Zhao *et al.* contributed to the field by applying the sum of the absolute transformed difference method to the detection of immature green citrus fruits [5]. The proposed technique effectively identified fruit pixels through the transfor-

mative process. A subsequent support vector machine classifier discerned and eliminated false positives, thereby refining the detection accuracy.

Dorj *et al.* introduced a novel algorithm aimed at automating fruit detection [6]. This algorithm encompassed a series of pivotal steps including the conversion of the RGB (red, green, blue) color space to the hue-saturation-value color space, threshold color detection, fruit segmentation, noise reduction, morphological operations, labeling, feature extraction, and classification.

Liu *et al.* devised a distinctive approach centered on the Cr-Cb color coordinates [7]. They established a multi-elliptical boundary model capable of detecting both citrus fruits and tree trunks in natural light settings. Another contribution by Liu *et al.* introduced a recognition methodology based on regional specifics [8]. The approach hinged on a feature mapping table, which effectively reduced the dimensionality of feature vectors while concurrently enabling the segmentation of citrus fruits, branches, and leaves.

Xu *et al.* pursued the segmentation of target citrus regions within the YUV color space by applying the Otsu adaptive threshold algorithm [9]. Their study incorporated a distinctive *random ring* method which used a greedy algorithm to recognize multiple citrus targets.

In a recent study, Zhang *et al.* introduced a pioneering algorithm that enabled the detection and quantification of citrus fruits within orchards [10]. The proposed methodology leveraged the LAB color space in tandem with the Hough circle transform. While image processing methods demonstrated proficiency in various fruit detection tasks, they encountered challenges when dealing with complex situations, such as occlusion, overlapping objects, and varied illumination [11].

Furthermore, the application of machine learning techniques to large-scale yield estimation often leads to suboptimal outcomes due to their constrained ability to generalize [12].

In recent years, there has been a growing interest in using object detection algorithms based on deep learning as promising tools for fruit detection and yield estimation. These algorithms have remarkable advanced generalization capabilities, which are categorized into one-stage and two-stage algorithms [13–15]. Typically, a one-stage algorithm offers faster inference speeds while a two-stage algorithm achieves better accuracy despite its relatively slower processing pace. The one-stage approach to target detection involves a convolutional neural network (CNN) to directly extract predictions for both the target class and its corresponding location within the input image. Instances of this methodology include the Single Shot MultiBox Detector (SSD) and the You Only Look Once (YOLO) algorithm represented by YOLOv1, YOLOv2, YOLOv3, YOLOv4, YOLOv5, YOLOx, and YOLOv7 [16–23]. As for the two-stage detection approach, the initial step employs a region proposal mechanism to sift through potential candidate regions. This process facilitates the acquisition of the region of interest, thus enabling the subsequent stages to engage in precise object localization and border regression prediction within the chosen region. Prominent exemplars of the two-stage detection strategy encompass such methods as Fast R-CNN, Faster R-CNN, and Mask R-CNN [24–26].

A cohort of researchers have contributed to the domain of fruit detection by employing deep learning models for object detection. The following researchers focused on the one-stage algorithm, e.g., improved YOLO models. Xu *et al.* introduced HPL-YOLOv4, an innovative approach for detecting citrus fruits [27]. This method employed GhostNet as its foundational backbone network and incorporated a DBM module with depthwise separable convolution and the Mish activation function, replacing the CBL module in the neck segment. The enhancements included integrating the ECA channel attention mechanism and using the soft DIoU-NMS technique to improve detection in overlapping or occluded situations. Yang *et al.* presented BCoYOLOv5, a novel network model for identifying and detecting fruit targets in orchards [28]. The model was based on YOLOv5s architecture and integrated a bidirectional cross attention mechanism for enhanced performance. Lai *et al.* introduced a target detection model based on an enhanced YOLOv7 variant, specifically designed for accurate pineapple detection in field environments [29]. The model incorporated the SimAM attention mechanism, refined the max-pooling convolution architecture, and replaced the conventional NMS with the soft-NMS variant to address detection challenges posed by occlusion and overlapping. Chen *et al.* proposed Citrus-YOLOv7 for citrus detection in orchards [30]. This model enhanced the YOLOv7 architecture with a specialized small object detection layer, lightweight convolution operations, and a convolutional block attention module. Yang *et al.* improved YOLOv7 to enhance apple fruit target recognition in scenarios with dense fruit clusters, occlusion, and overlapping [31]. They integrated a MobileOne module for backbone network establishment and used an altered image fusion strategy; this novel recognition algorithm also had an auxiliary detection head.

Our main objective was to investigate the potential application of unmanned aerial vehicle (UAV) remote-sensing technology and the YOLOv7 object detection model for citrus fruit yield estimation. Our solution will provide farmers with a precise and efficient alternative to traditional manual fruit count, leveraging cutting-edge technology to enhance decision-making in crop management.

STUDY OBJECTS AND METHODS

Study area. The research centered on the use of the *Maroc Late* variety of citrus trees in orchard environment. We obtained the original images of these citrus trees from an orchard located in the Beni Mellal-Khenifra region, Morocco. This region significantly contributes to citrus cultivation, accounting for 14% of

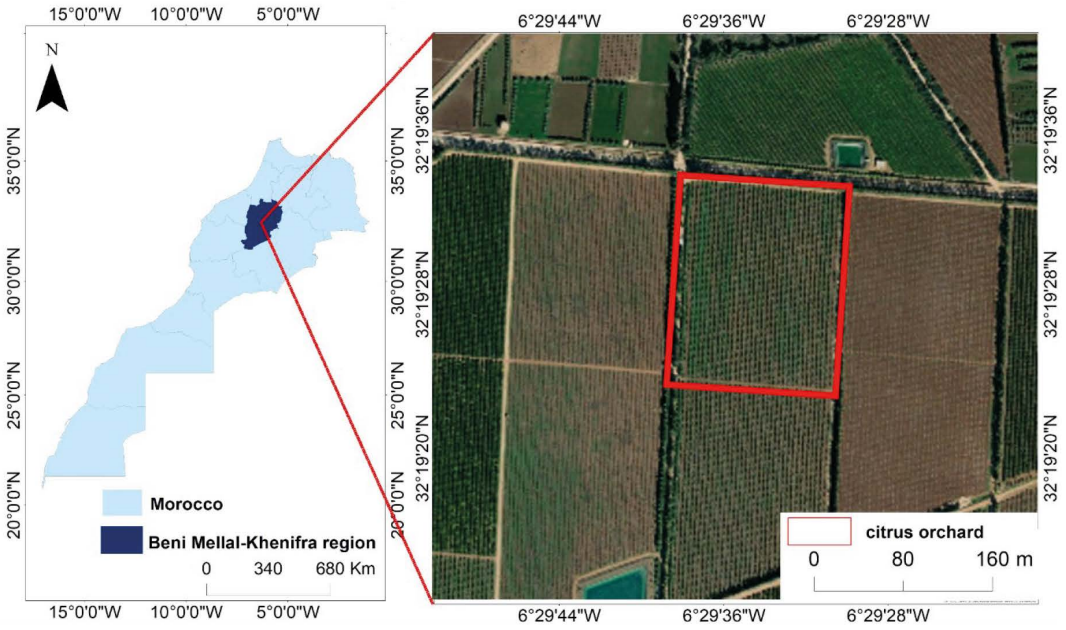

Figure 1 Citrus orchard: geographic location

the country's total citrus farming. The region allocates a substantial 17 426 ha to citrus cultivation, with the Maroc Late variety responsible for 23% of the citrus crops [32]. The central coordinates of the orchard under study are 32°19'28''N and 6°29'36''W (Fig. 1). The orchard, identified by parcel number 20 050, was established on August 18, 2018: it covered an area of 6.11 ha and included a total of 2546 citrus plants. In the standard planting configuration, the rows of citrus trees were separated by a 6-m gap, with each tree spaced approximately 4 m apart.

Sampling citrus trees. The initial tree sampling is crucial before capturing images in the orchard. This process involves selecting a few representative samples from the entire population. These samples should be robust and comprehensive enough to represent the entire orchard. This approach provides a more accurate estimation of the overall yield, thus securing a more precise assessment of the productivity of a particular orchard. This study employed two sampling methods.

The first method was random sampling: it provided a dataset to develop a deep learning model specifically targeting citrus fruit detection. The importance of employing random sampling comes from its ability to unbiasedly select samples from a diverse population. For this research, we selected 200 trees at random to be included in the dataset. The size of the dataset plays a pivotal role, as larger datasets enhance the capacity of deep learning models to recognize more complex patterns, thereby improving their generalization capabilities.

The second sampling method estimated the number of fruits on each citrus tree. We used the traditional method of manual counting to determine the total fruit count across a sample of 20 citrus trees. This sampling relied on the geographical location of the trees in the orchard: it involved four trees in each of the four directions (east, west, north, and south) plus another group of four trees in the middle of the orchard. The actual fruit count and the recognized fruit count generated by the algorithm were then paired for each of these trees. Utilizing a linear fitting method, we established a direct correlation between the observed fruit counts and the fruit counts identified by the algorithm created.

Table 1 Imaging system specifications

Parameter	Value
Sensor size, mm	4.87×3.96
Image dimensions, pixels	1600×1300
Focal length, mm	5.74 mm
Shutter type	Global 2 MP shutter

Data acquisition and UAV flights. The citrus trees designated for sampling were photographed on March 10 and 15, 2023, during their ripening season. We took the images at various times throughout the day – morning, noon, and afternoon – in the field under natural lighting conditions. The weather conditions during the image capture were ideal for UAV flights, with a wind speed of 9 km/h and a clear, cloudless sky.

This procedure involved the DJI Phantom 4 Multispectral (P4M) Unmanned aerial vehicle (UAV) equipped with a suite of imaging sensors. These sensors included five multispectral sensors representing the blue, green, red, red-edge, and near-infrared bands, along with one RGB sensor. Table 1 demonstrates the parameters of the sensors.

The DJI Pilot application served as the control interface for the P4M-UAV during the data collection process (Fig. 2). We employed manual control mode to navigate the UAV, with the camera angle adjusted to 45°. For each sampling tree, both right-side and left-side images were captured from a consistent distance of 4 m. The flight

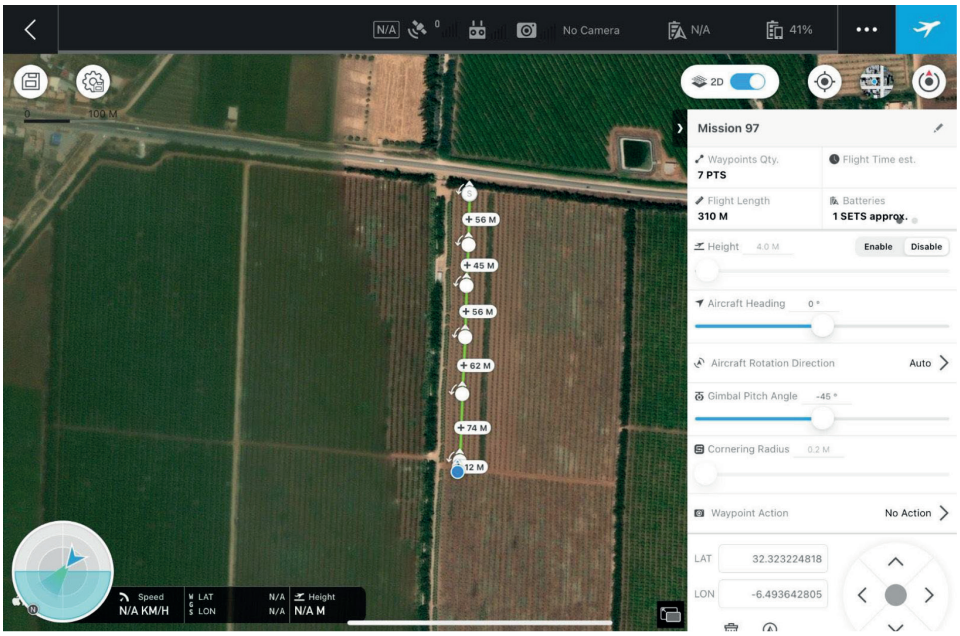

Figure 2 DJI Pilot application interface: examples of image capture locations (white points) and flight path (green line)

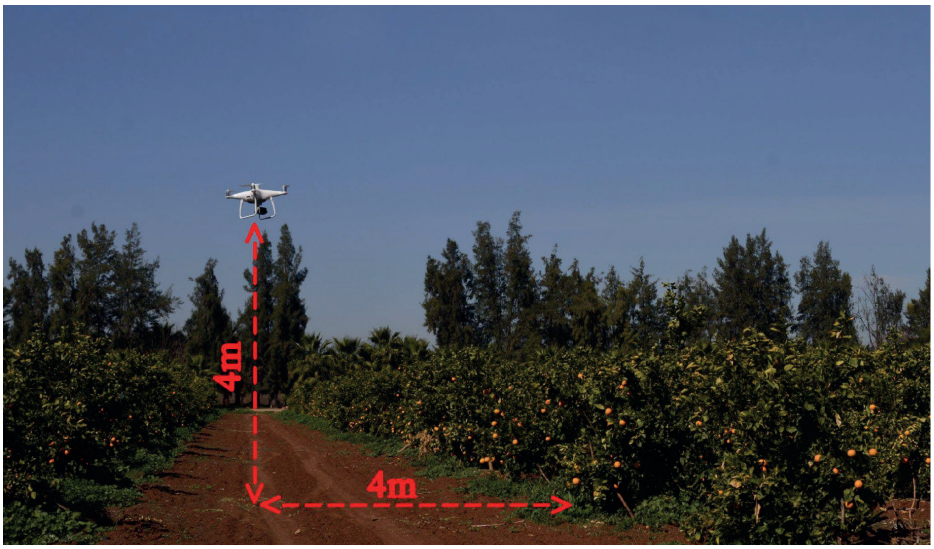

Figure 3 Phantom 4 (P4M) UAV imaging: distance from the tree is 4 m; altitude above ground level is 4 m

altitude was maintained at 4 m above ground level, as illustrated in Fig. 3. Consequently, the images obtained boasted a resolution of 0.21 cm/pixel.

We chose manual control instead of automated one to image the trees with a DJI Phantom 4 (P4M) UAV because we needed an accurate and focused data collection. Manual control allows for a high degree of operator engagement, enabling the UAV to maneuver around the selected trees effectively and flexibly. This approach made it possible to take pictures from both the left and right side at a constant height and distance. It provided a high control level, excellent data quality, and thorough coverage.

In the scope of this research, we had several reasons to use RGB sensor-captured images to develop a deep learning model focused on citrus fruit. Firstly, the RGB images were to be integrated with the deep learning model. As input data, RGB images were more effective in helping the model to identify and classify citrus fruits based on their color and other visual characteristics. Secondly, RGB images streamlined the computational and logistical complexities associated with the development of deep learning models, as opposed to multispectral images. Lastly, citrus fruits stood out clearly in RGB imagery, as illustrated in Fig. 4, thus facilitating the labeling and data verification, which, in turn, enhanced the overall reliability of the research outcomes.

Data preprocessing. In this study, we applied various preprocessing techniques to the original images. Initially, each original image underwent a cropping operation to create sub-images with dimensions of

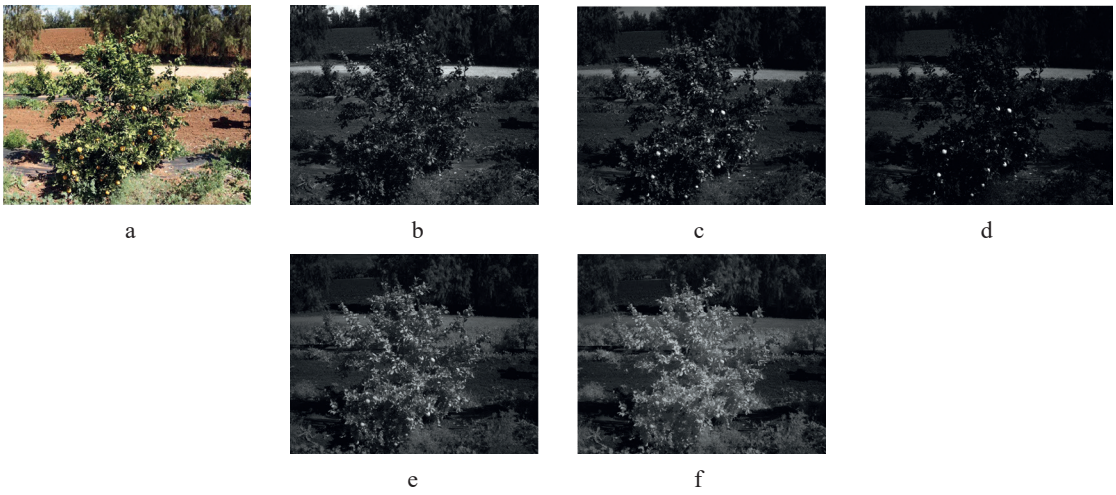

Figure 4 Exemplar citrus tree images captured by Phantom 4 Multispectral (P4M) UAV: (a) RGB (red, green, blue) composite; (b) blue channel; (c) green channel; (d) red channel; (e) red-edge channel; and (f) near-infrared image

Figure 5 Instances of sub-images in the dataset: back lighting (a), front lighting (b), occlusion (c), and overlap (d)

400×433 pixels. This action reduced the background noise and directed the focus towards the areas of particular interest. Consequently, we conducted an elimination process to filter out images devoid of citrus fruits, a measure taken to enhance the precision of the model by eliminating irrelevant data.

The sub-images in question manifested four distinct categories of interference: overlap, occlusion, front lighting, and back lighting. Figure 5 illustrates these types of interference. Overlap interference became evident when multiple citrus fruits partially obscure each other in an image. Occlusion interference arose when segments of a citrus fruit were concealed or shrouded by branches and leaves. Front lighting interference occurred when the illumination on citrus fruits grew intense from the frontal direction. Backlighting interference took place when the light source (the sun) was behind the citrus fruits, causing the fruits to appear as dark silhouettes.

Subsequently, we performed a manual annotation of a total of 1804 sub-images, using the LabelImg software (Fig. 6) to delineate the bounding boxes encompassing citrus fruits within each sub-image. After that, the program generated .txt format files with these annotations.

The dataset was further partitioned into three distinct subsets: a training set, a test set, and a validation set, with a distribution ratio of 70:20:10 (Table 2). The training set was comprised of 1263 sub-images with a total of 8185 citrus fruits. Meanwhile, the test set encompassed 361 sub-images, featuring 1253 citrus fruits, and the validation set comprised 130 sub-images with a combined total of 747 citrus fruits.

Data augmentation. We used a range of data augmentation strategies to enhance the diversity and size of the dataset. These strategies encompassed morphological operations, including angle rotation, saturation adjustment, image flipping (both vertically and horizontally), and translation. The mosaic data enhancement method involved the amalgamation of four defect images with random scaling, random clipping, and random layout adjustments: it bolstered the classification performance of the model. We also appealed to the mix-up data enhancement method to create mixed samples by proportionally interpolating two images. Additionally, we explored the color space conversion, modifications in picture hue, saturation, and exposure. The primary objective behind the incorporation of these data augmentation techniques was to curb the overfitting tendencies and bolster the model's capacity for generalization.

Object detection framework. YOLOv7 is a computer vision model within the YOLO (You Only Look Once) family of object detection models, renowned for its rapid detection, high precision, and user-friendly nature in both training and deployment. The YOLOv7 model architecture comprises five primary compo-

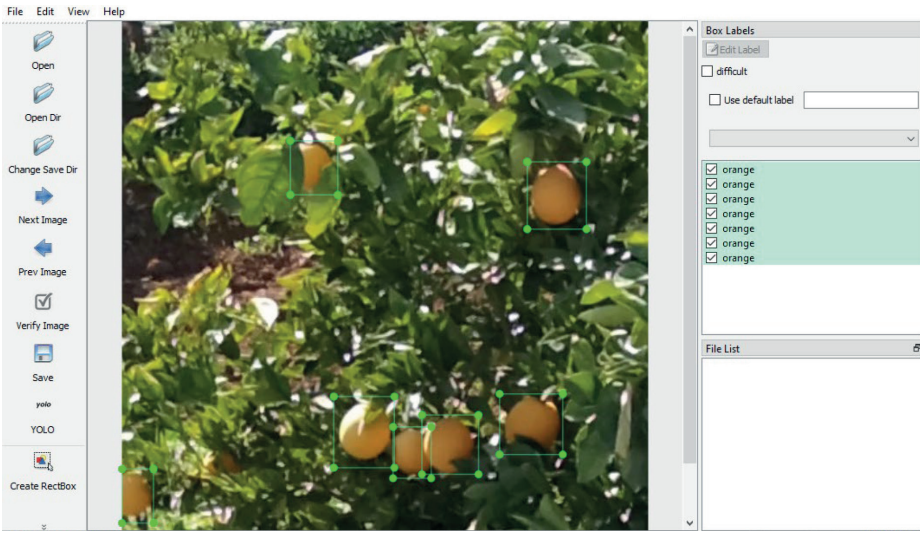

Figure 6 Manual annotation with green rectangles

Table 2 Dataset structure

Data set	Ratio, %	Number of sub-images	Number of fruits
Training se	70	1419	8185
Test set	20	361	1253
Validation set	10	180	747
Total	100	1804	10 185

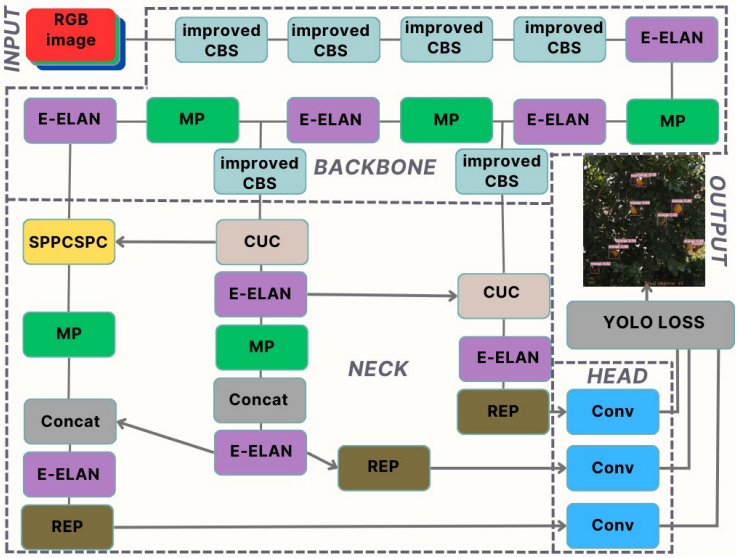

Figure 7 YOLOv7 improved architecture

nents: inputs, backbone network, neck, head, and loss function (Fig. 7). In this study, we applied various modifications to the original YOLOv7 architecture and training parameters to enhance its accuracy in detecting citrus fruits.

The input layer incorporated three techniques to enhance the quality of the data used for citrus fruit detection, i.e., mosaic data augmentation, adaptive anchor box calculation, and adaptive image scaling.

In this research, the backbone network played a crucial role in the feature extraction process. It comprised several modules, including BConv convolution layers, E-ELAN convolution layers, and MPConv convolution layers [23]. The BConv module, or CBS layer, consisted of a convolution layer, batch normalization (BN) layer, and SiLU activation function. It was specifically designed to extract image features at various scales (Fig. 8).

We conducted a series of experiments to explore different modifications in the backbone network, with a focus on the convolution (Conv) layer. The objective was to enhance the model's performance in citrus fruit detection. The incorporation of a double CBS (Conv-

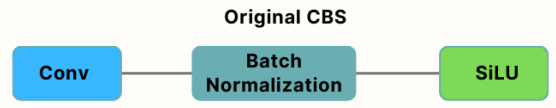

Figure 8 CBS Layer

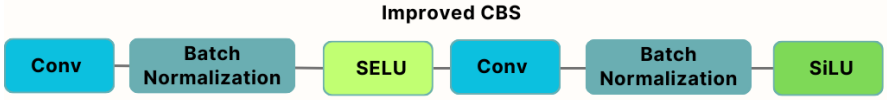

Figure 9 Improved CBS layer

BN-SELU-Conv-BN-SiLU) layer instead of a single CBS layer was the most successful change of all modifications tested. Additionally, we replaced the SiLU (sigmoid linear unit) activation function in the first layer with SELU, i.e., scaled exponential linear unit (Fig. 9).

The neck module in the YOLOv7 model architecture played a crucial role in feature fusion and feature pyramids. It served as a bridge between the backbone network and the head module, facilitating the integration of features from different network layers. The neck module consisted of two components: the feature pyramid network module and the path aggregation network module. These modules were responsible for merging and harmonizing the features extracted from multiple layers of the backbone network.

The head module in the YOLOv7 model architecture generated the final detections and predicted the locations and classes of objects within the input image. It was the last component in the YOLOv7 network before the output. Within the head module, the features that had been combined and mixed in the neck module passed through a series of layers that performed the necessary computations for object detection. These layers analyzed the feature representations, as well as made predictions about the bounding boxes and associated object classes. Additionally, the convolutional architecture was updated with the improved CBS to align the head architecture with the backbone architecture and enable the prediction of bounding boxes for small objects.

As for the loss function, YOLOv7 utilized a loss calculation method that consisted of three main components: object confidence loss, classification loss, and coordinate loss. These loss functions were important for training the model and optimizing its performance. The object confidence loss and classification loss in YOLOv7 were computed using the binary cross-entropy loss function. The binary cross-entropy loss measured the dissimilarity between the predicted probabilities and the ground truth labels for both object presence and class predictions. The coordinate loss in YOLOv7 employed the CIoU (complete intersection over union) loss function [33]. The CIoU loss took into account various factors, including the overlapping area, center distance, and aspect ratio, to measure the localization accuracy of the predicted bounding boxes.

Evaluation metrics. In this work, we used several metrics to assess the YOLOv7 performance, i.e., precision (*P*), recall, and *F*1-score (*F*1):

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

$$\text{Recall}=\frac{\text{TP}}{(\text{TP}+\text{FN})}$$

$$F1=\frac{2(\text{Precision}+\text{Recall})}{(\text{Precision}+\text{Recall})}$$

where the true positive (TP) was the number of images that the developed model correctly identified as containing citrus fruits; the false positive (FP) was the number of images that the model incorrectly identified as containing citrus fruits when they did not; the false negative (FN) was the number of images that the model incorrectly identifies as not containing citrus fruits when they did.

We used another formula to calculate the percentage of accurate citrus fruit count provided by the YOLOv7 model compared to the actual number of citrus fruits in the dataset:

$$\text{Rate of precision in yield estimation} =$$

$$=\frac{\text{Number of citrus fruits by YOLOv7}}{\text{Actual fruit number}}\times 100$$

where the number of citrus fruits counted by YOLOv7 was the count of citrus fruits detected by the YOLOv7 model; the actual fruit number was the real, or ground truth, count of citrus fruits in the dataset.

Experimental details. The network model was trained and evaluated on a dedicated laboratory workstation. It included the following hardware components: an Intel i9 13th Gen 13900K processor, an Nvidia RTX 4090 graphics card, 128 GB of 3200 MHz RAM, and a 2 TB Gen 4 SSD for storage. The operating system in use was a 64-bit professional edition of Win-

dows 10. For deep learning tasks, we used a PyTorch 2 with CUDA 11 as a framework; Python 3.8 served as a programming language. Throughout the training process, the input images were maintained at a resolution of 640×640 pixels.

We used the YOLO Evolve hyperparameter optimization method to determine the optimal hyperparameters for the YOLOv7 model. This approach involved 10 trials, each comprising 30 epochs, to assess various hyperparameter combinations and identify the most effective configuration. The relevant hyperparameter values were defined as follows: the model's initial learning rate was set to 0.129, the learning rate momentum was 0.892, the Adam algorithm served as optimizer, and the weight decay value was 0.00052. The training batch size was 32 while the total number of training epochs was 500. Additionally, we applied transfer learning by utilizing the pre-trained weights from *'yolov7_training.pt*,' a standard YOLOv7 model previously trained on the MS COCO dataset.

RESULTS AND DISCUSSION

Training results. Figures 10–14 provide an overview of various training metrics, including box loss, objective loss, precision, recall, and mAP0.5 values tracked after each training epoch. The box loss assessed the model's accuracy in locating the center of a citrus fruit within an image and drawing a bounding box around it. The objectness gauged the likelihood that a given image region contained the object of interest during detection. Over the training epochs, both box loss and objectness exhibited fluctuations and an overall

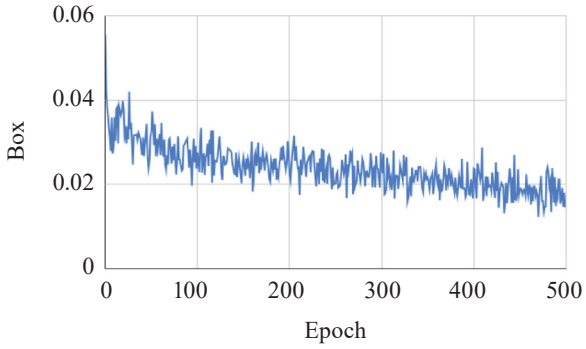

Figure 10 Plot of box loss for the training set

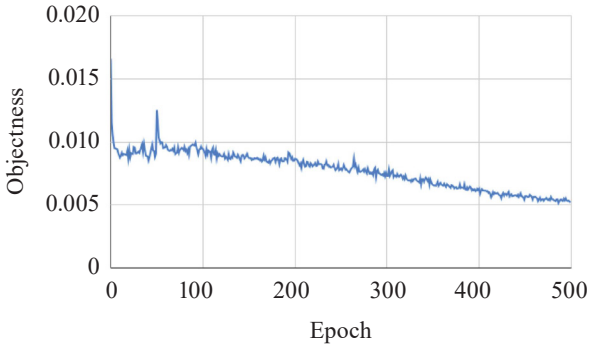

Figure 11 Plot of objectness loss for the training set

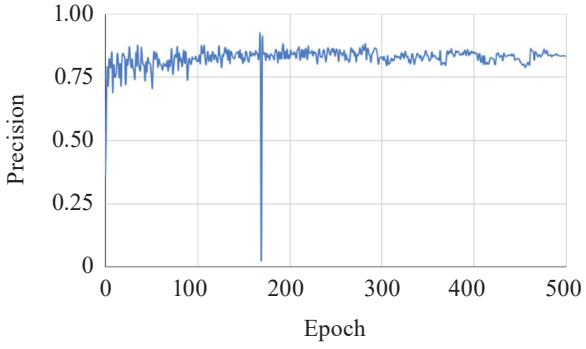

Figure 12 Plot of precision for the training set

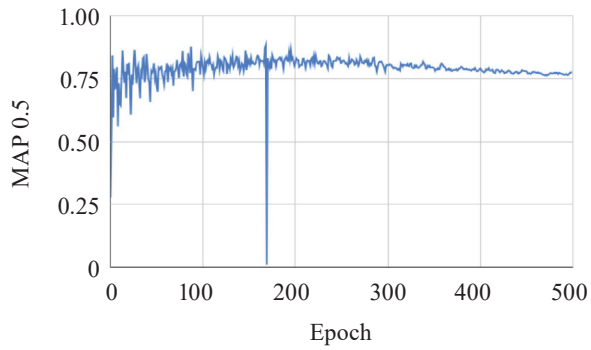

Figure 13 Plot of mean average precision (MAP) for the training set

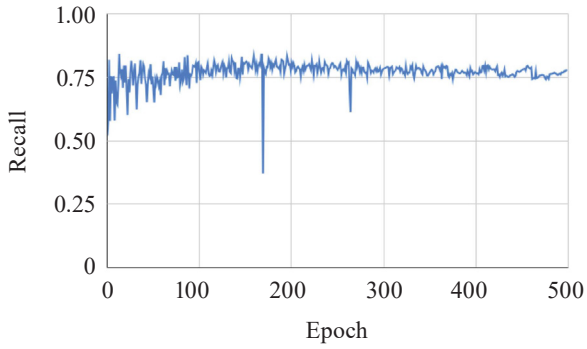

Figure 14 Plot of recall for the training set

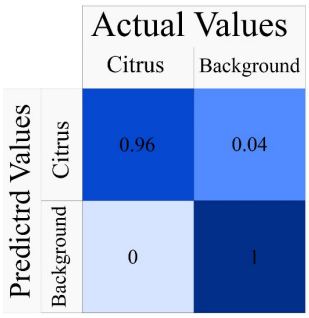

Figure 15 Confusion matrix of the test results

Figure 16 Visualizing citrus fruit detection performance in test dataset images

consistent decrease, indicating the progresses of the improved model. In the initial epochs (approximately, epochs 0–10), a rapid decrease signified quick learning. Subsequently, stability with fluctuations might have appeared due to varied data augmentation presenting both complex and simple instances. Towards the end of training (e.g., after epoch 400), stabilization signified that the model reached its learning capacity from the given data.

The metrics, including precision, recall, and mAP0.5 values, demonstrated fluctuations across epochs with an overall upward trend, reflecting improved model performance over training. In the early epochs (e.g., epochs 0–10), these metrics were relatively low but displayed significant improvement as the model learned data patterns. Around epochs 10–50, the rate of improvement slowed down as the model approached a better data representation. Throughout training, occasional fluctuations might be attributed to data augmentation, offering challenging and straightforward examples. A period of relative stability in precision from epochs 50–150 suggested a performance plateau given the architecture and data. Towards the end (epochs 400–500), a slower but continued improvement highlighted the model's refinement of learned features.

Test results. Figure 15 presents the confusion matrix of the test results, i.e., a critical visual representation of the deep learning model's performance in detecting citrus fruits amidst background objects. The model excelled in accurate identification, achieving a 96% true positive rate, but still exhibited a minor shortcoming with a 4% false positive rate. On the other hand, it effectively identified the background as not containing citrus fruits with a true negative rate of 100%. Figure 16 displays a visualization of selected output from YOLOv7 on several test images from the dataset.

Utilizing YOLOv7 for fruit detection yielded good outcomes, underscoring the model's effectiveness. The achieved results included the precision of 96%, a recall of 100%, and an $F1$-score of 97.95%. Such a high level of detection accuracy could be attributed to a range of strategic approaches, i.e., various modifications to the original YOLOv7 architecture, data augmentation, the careful selection of hyperparameters through the YOLO Evolve method, and the application of transfer learning. Expanding the dataset with additional images of citrus fruits collected from various farms and at different time intervals could prove advantageous to further enhance the performance.

However, despite these optimizations, the model may not attain perfect accuracy due to potential interferences, e.g., leaves and branches obstructing the view. The algorithm relied on a combination of parameters, including color, texture, and various other features. Additionally, the overall detection performance relied on the image quality, which, in its turn, depended on such factors as UAV camera specifications, the time of image capture, lighting conditions, the altitude of the UAV flight, and the horizontal distance between the UAV and the tree.

Yield estimation results. Table 3 illustrates the results of citrus detection achieved by the enhanced YOLOv7 model, conducted across a sample of 20 distinct trees. For each tree, we determined the accuracy rate in estimating the yield. The highest accuracy observed was 94% while the lowest was 74%; the overall average across all trees stood at 87.68%. The improved YOLOv7 model recorded a lower count of citrus fruits compared to the actual count. This disparity between the manual counting and the proposed algorithm could be attributed to several factors, including complete occlusion, shadows, and viewing angles. Manual counting involved capturing fruit numbers from multiple angles whereas the algorithm relied on images taken from two sides of the tree. We performed a regression analysis to assess the correlation between the manual count and the count generated by YOLOv7 for 20 trees (Fig. 17). The resulting regression equation was $y = 1.10 \cdot x + 3.27$.

Table 3 Citrus fruit count results for 20 trees

Tree number	Actual fruit number	Fruit count by YOLOv7	Precision rate in yield estimation
1	52	46	88.46
2	125	113	90.40
3	74	69	93.24
4	131	115	87.70
5	88	74	84.09
6	81	60	74.07
7	140	121	86.42
8	129	109	84.49
9	52	49	94.23
10	61	51	,83.60
11	83	74	89.15
12	129	114	88.37
13	141	127	90.07
14	86	79	91.86
15	104	93	89.42
16	72	57	79.16
17	66	59	89.39
18	133	120	90.22
19	89	81	91.01
20	68	60	88.23

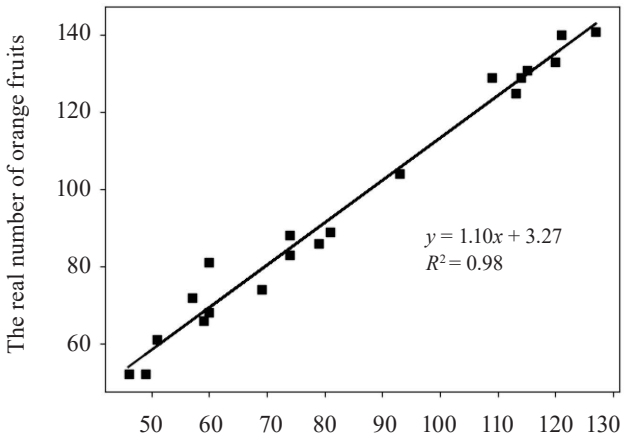

Figure 17 Regression analysis: actual number of citrus fruits vs. citrus fruits detected by the YOLOv7 model

It was accompanied by a high correlation coefficient (R^2 = 0.98), which signified a robust correlation within the dataset.

Evaluation. Table 4 compares the proposed approach with previous studies in terms of the algorithm, precision, recall, and *F*1-score.

In this comparative analysis, we assessed the performance of our approach against the methodologies employed in previous studies that utilized various YOLO (You Only Look Once) variants for citrus fruit detection. Xu *et al.* used HPL-YOLOv4 to achieve the precision, recall, and *F*1 score metrics of 93.45, 94.30, and 94.00%, respectively [27]. Yang *et al.*, who used BCo-YOLOv5, reported a higher recall at 97.11%, albeit with a slightly lower precision and an *F*1-score of 89.15 and 92.96% [28]. Chen *et al.* introduced Citrus-YOLOv7: they showcased a well-balanced precision of 94.25%, a recall of 93.37%, and an *F*1-score of 93.81% [30]. Our resalts proved remarkable with the precision of 96%, a recall of 100%, and an *F*1-score of 97.95%. These findings suggest that the method described in this paper may represent a significant advancement in enhancing the accuracy of citrus fruit detection compared to earlier methodologies in this field.

Table 4 Comparative analysis of object detection model performances: precision, recall, and *F*1-score

Reference	Model	Precision, %	Recall, %	*F*1-score, %
[27]	HPL-YOLOv4	93.45	94.30	94.00
[28]	BCo-YOLOv5	89.15	97.11	92.96
[30]	Citrus-YOLOv7	94.25	93.37	93.81
This work	YOLOv7	96.00	100.00	97.95

CONCLUSION

In this study, we used unmanned aerial vehicle (UAV) RGB (red, green, blue) remote-sensing imagery and the YOLOv7 object detection model to estimate citrus fruit yield. The innovative modifications to the YOLOv7 model included the introduction of a double CBS (Conv-BN-SELU-Conv-BN-SiLU) layer and the adoption of the SELU activation function. They made it possible to achieve commendable results in citrus fruit detection. The UAV RGB remote-sensing technology enhanced the capabilities of the deep learning model by providing high-resolution, real-time aerial imagery, and, eventually, a more comprehensive assessment of citrus orchards. Hyperparameter optimization with the YOLO Evolve method further improved the performance, resulting in high precision, recall, and *F*1-score values.

Our findings demonstrated the potential of deep learning object detection models in addressing the challenges associated with traditional fruit counting methods. Cutting-edge technologies, e.g., UAVs, may reduce the labor-intensive and error-prone nature of manual fruit counting, thus providing accurate and efficient estimates for citrus fruit yield.

Our algorithm proved effective in identifying and quantifying citrus fruits, as evidenced by the strong positive correlation between the recognized fruit numbers and the actual fruit numbers from a sample of 20 trees. Our algorithm, combined with UAV RGB remote-sensing, can assist farmers in making informed decisions about crop management.

While the results are promising, we have to acknowledge certain limitations, such as occlusion, that may affect detection accuracy. Further research could expand the dataset to encompass diverse conditions and varieties of citrus fruits, potentially enhancing the model's robustness.

CONTRIBUTION

All the authors were equally involved in the research analysis and manuscript writing.

CONFLICT OF INTEREST

The authors declared no conflict of interests regarding the publication of this article.

REFERENCES

1. Marani R, Milella A, Petitti A, Reina G. Deep neural networks for grape bunch segmentation in natural images from a consumer-grade camera. Precision Agriculture. 2021;22:387–413. https://doi.org/10.1007/s11119-020-09736-0
2. Gongal A, Amatya S, Karkee M, Zhang Q, Lewis K. Sensors and systems for fruit detection and localization: A review. Computers and Electronics in Agriculture. 2015;116:8–19. https://doi.org/10.1016/j.compag.2015.05.021
3. Sengupta S, Lee WS. Identification and determination of the number of immature green citrus fruit in a canopy under different ambient light conditions. Biosystems Engineering. 2014;117:51–61. https://doi.org/10.1016/j.biosystemseng.2013.07.007
4. Maldonado Jr W, Barbosa JC. Automatic green fruit counting in orange trees using digital images. Computers and Electronics in Agriculture. 2016;127:572–581. https://doi.org/10.1016/j.compag.2016.07.023
5. Zhao C, Lee WS, He D. Immature green citrus detection based on colour feature and sum of absolute transformed difference (SATD) using colour images in the citrus grove. Computers and Electronics in Agriculture. 2016;124:243–253. https://doi.org/10.1016/j.compag.2016.04.009
6. Dorj U-O, Lee M, Yun S. An yield estimation in citrus orchards via fruit detection and counting using image processing. Computers and Electronics in Agriculture. 2017;140:103–112. https://doi.org/10.1016/j.compag.2017.05.019
7. Liu T-H, Ehsani R, Toudeshki A, Zou X-J, Wang H-J. Detection of citrus fruit and tree trunks in natural environments using a multi-elliptical boundary model. Computers in Industry. 2018;99:9–16. https://doi.org/10.1016/j.compind.2018.03.007
8. Liu S, Yang C, Hu Y, Huang L, Xiong L. A method for segmentation and recognition of mature citrus and branches-leaves based on regional features. In: Wang Y, Jiang Z, Peng Y, editors. Image and graphics technologies and applications. Singapore: Springer; 2018. pp. 292–301. https://doi.org/10.1007/978-981-13-1702-6_29
9. Xu L, Zhu S, Chen X, Wang Y, Kang Z, Huang P, *et al.* Citrus recognition in real scenarios based on machine vision. DYNA. Ingenier´ıa e Industria. 2020;95(1):87–93. https://doi.org/10.6036/9363
10. Zhang X, Toudeshki A, Ehsani R, Li H, Zhang W, Ma R. Yield estimation of citrus fruit using rapid image processing in natural background. Smart Agricultural Technology. 2022;2:100027. https://doi.org/10.1016/j.atech.2021.100027
11. Maheswari P, Raja P, Apolo-Apolo OE, Perez-Ruiz M. Intelligent fruit yield estimation for orchards using deep learning based semantic segmentation techniques – A review. Frontiers in Plant Science. 2021;12:684328. https://doi.org/10.3389/fpls.2021.684328
12. Yamamoto K, Guo W, Yoshioka Y, Ninomiya S. On plant detection of intact tomato fruits using image analysis and machine learning methods. Sensors. 2014;14(7):12191–12206. https://doi.org/10.3390/s140712191
13. Loddo A, Loddo M, Di Ruberto C. A novel deep learning based approach for seed image classification and retrieval. Computers and Electronics in Agriculture. 2021;187:106269. https://doi.org/10.1016/j.compag.2021.106269
14. Han B-G, Lee J-G, Lim K-T, Choi D-H. Design of a scalable and fast YOLO for edge-computing devices. Sensors. 2020;20(23):6779. https://doi.org/10.3390/s20236779
15. Sivakumar ANV, Li J, Scott S, Psota E, Jhala AJ, Luck JD, *et al.* Comparison of object detection and patch-based classification deep learning models on mid-to late-season weed detection in UAV imagery. Remote Sensing. 2020;12(13):2136. https://doi.org/10.3390/rs12132136
16. Liu W, Anguelov D, Erhan D, Szegedy C, Reed S, Fu C-Y, *et al.* SSD: Single shot multibox detector. In: Leibe B, Matas J, Sebe N, Welling M, editors. Computer Vision – ECCV 2016. Cham: Springer; 2016. pp. 21–37. https://doi.org/10.1007/978-3-319-46448-0_2
17. Redmon J, Divvala S, Girshick R, Farhadi A. You only look once: Unified, real-time object detection. Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition; 2016; Vegas. IEEE; 2016. pp. 779– 788. https://doi.org/10.1109/CVPR.2016.91
18. Redmon J, Farhadi A. YOLO9000: better, faster, stronger. Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition; 2017; Honolulu. IEEE; 2017. pp. 7263–7271. https://doi.org/10.1109/CVPR.2017.690
19. Redmon J, Farhadi A. YOLOv3: An incremental improvement. 2018. https://doi.org/10.48550/arXiv.1804.02767
20. Bochkovskiy A, Wang C-Y, Liao H-YM. YOLOv4: Optimal speed and accuracy of object detection. 2020. https://doi.org/10.48550/arXiv.2004.10934

21. Jocher G, Chaurasia A, Stoken A, Borovec J, Kwon Y, Fang J, *et al.* ultralytics/yolov5: v6.1 – TensorRT, TensorFlow edge TPU and OpenVINO export and inference. Zenodo. 2022. https://doi.org/10.5281/zenodo.6222936
22. Ge Z, Liu S, Wang F, Li Z, Sun J. YOLOX: Exceeding YOLO series in 2021. 2021. https://doi.org/10.48550/arXiv.2107.08430
23. Wang C-Y, Bochkovskiy A, Liao H-YM. YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. Proceedings of the 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition; 2023; Vancouver. IEEE; 2023. pp. 7464–7475. https://doi.org/10.1109/CVPR52729.2023.00721
24. Girshick R. Fast r-cnn. Proceedings of the 2015 IEEE International Conference on Computer Vision; 2015; Santiago. IEEE; 2015. pp. 1440–1448. https://doi.org/10.1109/ICCV.2015.169
25. Ren S, He K, Girshick R, Sun J. Faster R-CNN: Towards real-time object detection with region proposal networks. In: Cortes C, Lawrence N, Lee D, Sugiyama M, Garnett R, editors. Advances in neural information processing systems. Purchase Printed Proceeding; 2015.
26. Lucena F, Breunig FM, Kux H. The combined use of UAV-based RGB and DEM images for the detection and delineation of orange tree crowns with mask R-CNN: An approach of labeling and unified framework. Future Internet. 2022;14(10):275. https://doi.org/10.3390/fi14100275
27. Xu L, Wang Y, Shi X, Tang Z, Chen X, Wang Y, *et al.* Real-time and accurate detection of citrus in complex scenes based on HPL-YOLOv4. Computers and Electronics in Agriculture. 2023;205:107590. https://doi.org/10.1016/j.compag.2022.107590
28. Yang R, Hu Y, Yao Y, Gao M, Liu R. Fruit target detection based on BCo-YOLOv5 model. Mobile Information Systems. 2022;2022:8457173. https://doi.org/10.1155/2022/8457173
29. Lai Y, Ma R, Chen Y, Wan T, Jiao R, He H. A pineapple target detection method in a field environment based on improved YOLOv7. Applied Sciences. 2023;13(4):2691. https://doi.org/10.3390/app13042691
30. Chen J, Liu H, Zhang Y, Zhang D, Ouyang H, Chen X. A multiscale lightweight and efficient model based on YOLOv7: Applied to citrus orchard. Plants. 2022;11(23):3260. https://doi.org/10.3390/plants11233260
31. Yang H, Liu Y, Wang S, Qu H, Li N, Wu J, *et al.* Improved apple fruit target recognition method based on YOLOv7 model. Agriculture. 2023;13(7):1278. https://doi.org/10.3390/agriculture13071278
32. Ministry of Agriculture. https://www.agriculture.gov.ma
33. Zheng Z, Wang P, Liu W, Li J, Ye R, Ren D. Distance-IoU loss: Faster and better learning for bounding box regression. AAAI-20 Technical Tracks 7. 2020;34(7):12993–13000. https://doi.org/10.1609/aaai.v34i07.6999

ORCID IDs
Mohamed Jibril Daiaeddine https://orcid.org/0009-0006-5525-7956
Sara Badrouss https://orcid.org/0009-0000-2675-4810
Abderrazak El Harti https://orcid.org/0000-0003-3976-4588
El Mostafa Bachaoui https://orcid.org/0000-0003-4163-6307
Mohamed Biniz https://orcid.org/0000-0002-9448-6165
Hicham Mouncif https://orcid.org/0000-0003-3312-8230