ICPA
17th INTERNATIONAL CONFERENCE ON PRECISION AGRICULTURE
ConBAP 2026
CONGRESSO BRASILEIRO DE AGRICULTURA DE PRECISÃO E DIGITAL
13-16 July 2026 | Porto Alegre, Brazil
ISPA
AsBraAP

Automated Detection of Melons (*Cucumis Melo L.*) via multispectral UAV and Deep Learning in Honduras

Eduin Torres Avila[1], Ricardo Sobjak[2], Samuel Moro Lumertz[2], Erney Ricardo Cely Bonilla[2], Dayerling Del Carmen Perez Suarez[2], Tito Alex Medeiros[1,2], Kelyn Schenatto[2], Claudio Leones Bazzi[2], Marcio Furlan Maggi[1]

[1]State University of Western Paraná (UNIOESTE), PGEAGRI, Cascavel, PR, Brazil.

[2]Federal Technological University of Paraná (UTFPR), PPGTCA, Medianeira, PR, Brazil.

A paper from the Proceedings of the 17th International Conference on Precision Agriculture and 11th Brazilian Congress on Precision and Digital Agriculture

13-15 July 2026

Porto Alegre, Brazil

Abstract.

This study evaluates the performance of YOLO-based object detectors for melon detection in UAV RGB imagery acquired under dense field conditions. A dataset of aerial images was annotated with bounding boxes and organized through a reproducible preprocessing workflow, including annotation normalization, integrity checks, deterministic train/validation/test splitting, and export to the standardized training format. Multiple YOLO-based models were fine-tuned and compared under the same experimental protocol, with model selection based on validation performance and final reporting on a held-out test set. The best-performing model achieved a test mAP@0.5 of approximately 0.93 and a test mAP@0.5:0.95 of approximately 0.48, indicating strong detection performance at moderate IoU thresholds but reduced robustness under stricter localization criteria. These results show that modern one-stage detectors can effectively detect melons in dense UAV imagery, while also highlighting persistent challenges related to small-object localization, object overlap, and limited dataset diversity. Overall, this work provides a reproducible benchmark for UAV-based melon detection and a baseline for future extensions toward counting and yield estimation.

Keywords.

UAV; melon detection; object detection; YOLO; precision agriculture; deep learning.

The authors are solely responsible for the content of this paper, which is not a refereed publication. Citation of this work should state that it is from the Proceedings of the 17th International Conference on Precision Agriculture and 11th Brazilian Congress on Precision and Digital Agriculture. Torres Avila Eduin [2], Sobjak R[1], Moro Lumertz S[1], Cely Bonilla E[1], Perez D[1], Medeiros A.T[1], Schenatto K[1], Bazzi L. C[1], Furlan Maggi M[2] (2026). Title of paper. In Proceedings of the 17th International Conference on Precision Agriculture and 11th Brazilian Congress on Precision and Digital Agriculture (unpaginated, online). Monticello, IL: International Society of Precision Agriculture and Brazilian Congress on Precision and Digital Agriculture.

Introduction

Accurate and timely information on fruit presence and distribution is important for operational decision-making in precision agriculture, supporting activities such as field inspection, harvest planning, and crop monitoring (Tsouros et al., 2019; Wahab et al., 2018). Unmanned Aerial Vehicles (UAVs) have become an attractive sensing platform in this context because they can acquire high-resolution imagery on demand, enabling flexible, repeated monitoring at the field scale with comparatively low operational cost (Tsouros et al., 2019). In agricultural applications, UAV-derived imagery has been used both for indirect assessment through remote-sensing proxies and for direct visual analysis when the target objects can be resolved in the scene (Tsouros et al., 2019; Wahab et al., 2018).

For fruit crops, direct detection of visible fruits is a particularly relevant computer vision task because it can serve as the basis for downstream applications such as counting, spatial distribution analysis, and yield-oriented monitoring. However, automatic fruit detection in field imagery remains difficult due to the combination of small target size, object overlap, partial occlusion, illumination variation, and complex background patterns (Jiang et al., 2024; Khan et al., 2025). These difficulties become even more pronounced in UAV imagery, where the scene may contain many visually similar objects distributed over large areas, often with limited apparent size in the image plane.

Recent advances in deep learning, especially modern object detection architectures, have substantially improved automated perception in agriculture, enabling practical systems for crop and fruit detection under real conditions (Khan et al., 2025). Among these methods, YOLO-style detectors are widely adopted because they offer a useful balance between detection accuracy and computational efficiency, making them especially attractive for applied agricultural scenarios (Khan et al., 2025; Redmon et al., 2016). In addition, standardized annotation formats and reproducible training protocols are essential for ensuring that model comparisons are fair and scientifically interpretable (Khan et al., 2025; Lin et al., 2014).

In this paper, we investigate the use of YOLO-based object detectors for melon detection in UAV RGB imagery acquired under dense field conditions. Rather than framing the task directly as yield estimation, we focus on the detection stage as the core computer vision problem that underlies future counting pipelines. Specifically, we (i) organize and normalize bounding-box annotations into a reproducible dataset preparation workflow, (ii) create deterministic train/validation/test splits with integrity checks, (iii) benchmark multiple YOLO variants under the same training protocol, and (iv) analyze the main limitations observed under strict localization metrics. The main contributions of this work are therefore a reproducible experimental pipeline for UAV melon detection, a controlled benchmark across multiple YOLO-based models, and an empirical analysis of error patterns in dense small-object field imagery. The remainder of this article presents the dataset and experimental protocol, reports the benchmark results, discusses the main limitations, and outlines directions for extending the approach toward robust counting in operational agricultural settings.

Materials and methods

Study Context and Data Acquisition

This study addresses melon detection in UAV RGB imagery under dense field conditions, where small object size, visual similarity between fruits, partial occlusion, and background clutter make automated localization challenging. RGB aerial images were collected over melon cultivation areas using a UAV platform. To preserve scientific reproducibility, the acquisition protocol should be described with the corresponding operational details, including the UAV and camera models, flight altitude, approximate ground sampling distance, number of flights, acquisition period, and field location. In the current workflow, flight planning prioritizes sufficient spatial detail to make individual fruits visible and annotatable while maintaining image quality and consistency across the surveyed area. Stable acquisition conditions and image overlap were adopted to reduce motion blur and facilitate downstream analysis.

Annotation Procedure

All images were annotated with axis-aligned bounding boxes around visible melons using Roboflow as an annotation interface. The labeling protocol followed a pragmatic rule: each clearly visible fruit received one bounding box; partially occluded fruits were annotated whenever a consistent box could be drawn; and ambiguous regions were left unannotated to reduce label noise. After annotation, the dataset was exported in COCO JSON format for standardization and downstream processing (Lin et al., 2014; Roboflow, 2025).

Dataset Normalization and Integrity Checks

To ensure reproducibility and consistent downstream conversions, the exported annotations were normalized into a canonical COCO representation (Lin et al., 2014). This normalization step enforces: (i) consistent category naming and indexing, (ii) standardized image metadata fields, (iii) stable and non-colliding identifiers for images and annotations, and (iv) bounding-box validity constraints. Integrity checks were executed to detect and handle common dataset issues, including bounding boxes exceeding image boundaries, missing image references, empty images, and invalid category references. When necessary, annotations were clipped to valid image boundaries or filtered to prevent invalid geometry from propagating into the training set.

Train/Validation/Test Split

The dataset split was performed locally rather than through the annotation platform, so that the annotation tool remained restricted to annotation and export. At the same time, all experimental partitions remained under explicit version control. A deterministic random split was created with a fixed seed and configured train/validation/test ratios, ensuring that no image filename appears in more than one split. The split procedure outputs per-split COCO JSON files and explicit file lists, enabling transparent auditing and repeatable reruns.

Although an image-level deterministic split was adopted for reproducibility, UAV imagery may contain highly similar and spatially adjacent frames. Therefore, the current protocol should be interpreted as an initial benchmark rather than a strict cross-location generalization test. Future experiments should consider block-wise, row-wise, or flight-wise splits to better control spatial leakage between training and evaluation subsets.

Training Format Export

For model training, the split dataset was converted to the Ultralytics YOLO layout, with separate images/ and labels/ folders for each split.yaml file defining paths and class names (Ultralytics, 2025a, 2025b). Each label file stores one object per line in YOLO format, i.e., a class ID and normalized bounding-box coordinates (xc, yc, w, h) relative to the image width and height. The export step includes a validation pass to confirm that each image has a matching label file when applicable, that label values are within valid numeric ranges, and that the dataset.yaml parsing is correct in the Ultralytics training pipeline.

Models and Training Protocol

We benchmarked multiple YOLO-family detectors under the same protocol, using pretrained weights followed by fine-tuning on the melon dataset. YOLO-style detectors are widely used for real-time object detection because they combine single-stage prediction with efficient end-to-end training [6]. In our benchmark, the evaluated models were YOLO26m, YOLO26l, YOLO11l, and YOLOv8l, all trained on the same dataset split under comparable hyperparameter settings [10]. Training used a fixed input resolution and a consistent optimization setup defined in the project configuration files, so that observed performance differences are primarily attributable to architecture capacity and scaling rather than major protocol changes.

To strengthen reproducibility, the final version of the manuscript should explicitly report the main training hyperparameters, including input image size, batch size, number of epochs, optimizer, learning rate, augmentation configuration, patience or early-stopping settings, random seed, and inference thresholds used during evaluation.

Data Augmentation and Preprocessing

To improve robustness to photometric variation and small appearance changes, we applied training-time augmentations via Albumentations (Buslaev et al., 2020). The policy includes mild blurring, contrast-limited adaptive histogram equalization (CLAHE), and occasional grayscale conversion, reflecting plausible variations in UAV imagery, such as lighting shifts, haze, compression artifacts, and contrast variation. Augmentations were applied only to the training split; validation and test splits remained unaugmented to preserve unbiased evaluation.

Runtime Environment

Experiments were executed in a controlled Python environment with fixed dependencies. Training was run on the CPU or an NVIDIA GPU, depending on the execution platform, and device selection was explicitly configured at runtime to avoid unintended fallbacks. For the final manuscript version, the exact hardware configuration should also be reported, including processor, GPU model, memory, CUDA environment when applicable, and software versions relevant to training and inference.

Experiment Tracking and Reproducibility

Each run was logged with its configuration, dataset identifiers, outputs, and metrics to support reproducibility and fair comparison across models. For each experiment, the workflow stored training arguments, the best and last weights, and training artifacts such as plots and prediction visualizations in a structured directory hierarchy. MLflow was used to consolidate run information and exported results, enabling reruns from recorded configurations and systematic comparison of candidate models (MLflow Contributors, 2026).

Evaluation Metrics

Model performance was evaluated using standard object detection metrics: precision, recall, mAP@0.5, and mAP@0.5:0.95. During model development, validation mAP@0.5:0.95 was adopted as the primary ranking metric because it better reflects localization quality under stricter IoU thresholds than mAP@0.5 alone. Model selection was therefore based on validation performance, and final results were reported once on the held-out test set. Since the broader application of interest is counting, the present study should be interpreted specifically as a detection benchmark; no direct counting-error metric was used in the current evaluation.

Results

Dataset summary

This section summarizes the final dataset used in the experiments after normalization, integrity checks, and deterministic splitting. Table 1 reports the number of images and annotated instances per split. These values indicate a relatively small dataset in terms of image count but a high number of annotated objects per image, consistent with a dense detection scenario.

Table 1. Dataset summary and split counts after local deterministic splitting.

Split	Images	Instances (Boxes)
Train	39	5455
Validation	9	1574
Test	8	1122
Total	56	8151

Dataset characteristics. All images were acquired at a fixed spatial resolution of 786×1313 pixels and contain a high density of small objects. The mean number of instances per image was 145.55 (median: 148.5), reflecting a challenging scenario with frequent object proximity and overlap. The typical bounding-box area ratio relative to the image was 1.30×10^{-4} (range: 1.27×10^{-5}–3.67×10^{-4}), indicating that melons often occupy only a very small portion of the frame (Fig. 1).

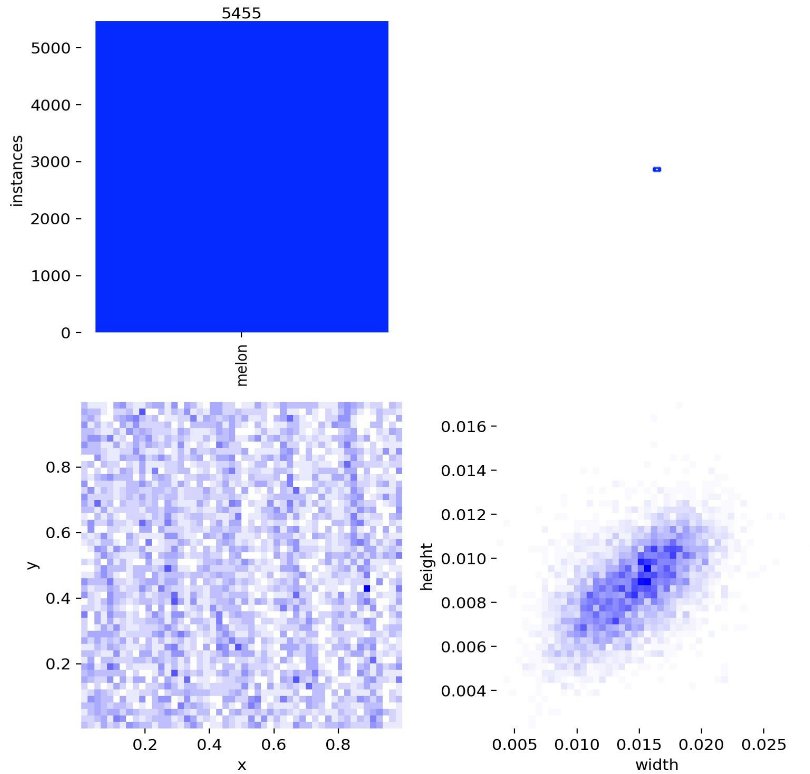


Fig. 1. Training-set label distribution and annotation statistics exported by the training pipeline.

Benchmark Results

We compared multiple YOLO-based detectors under the same training and evaluation protocol. Model selection was based on validation mAP@0.5:0.95, and final performance was then reported on the held-out test set. Table 2 summarizes the test results for all evaluated models.

Table 2. Benchmark comparison across YOLO variants on the held-out test set.

Model	Precision	Recall	mAP@0.5	mAP@0.5:0.95
YOLO26	0.877	0.864	0.929	0.477
YOLO26m	0.864	0.864	0.934	0.472
YOLOv8l	0.868	0.875	0.927	0.451
YOLO11l	0.855	0.867	0.915	0.449

Overall performance. All evaluated models achieved high detection quality at IoU=0.5, with test mAP@0.5 values ranging from 0.915 to 0.934. In contrast, test mAP@0.5:0.95 values ranged from 0.449 to 0.477, indicating that stricter localization accuracy is the main difficulty in this task. The best-performing model, YOLO26l, achieved a test mAP@0.5:0.95 of 0.4769, together with a precision of 0.8770 and a recall of 0.8640. Although the ranking between models was consistent under the primary metric, the absolute differences were modest, suggesting that all evaluated architectures behave similarly under the current dataset conditions.

Qualitative Dataset Examples

Figs. 2 and 3 present representative annotated examples from the evaluation data. These images illustrate the visual difficulty of the task, including dense fruit arrangements, small apparent object size, strong proximity between adjacent melons, and partial occlusion. Even before considering model predictions, the annotation examples already indicate why precise localization is more difficult than coarse detection in this scenario.

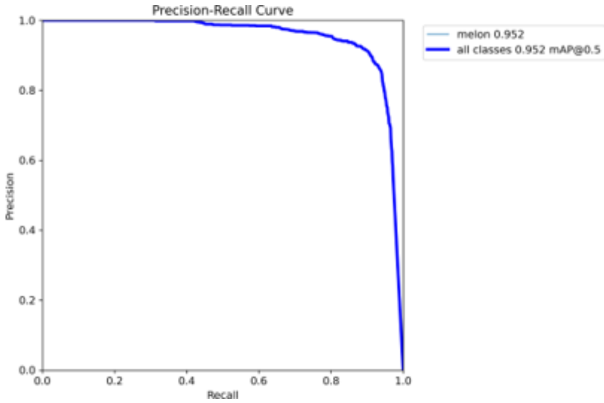
Fig. 2. Annotated example from the held-out test data.

Fig. 3. Annotated example from the validation data.

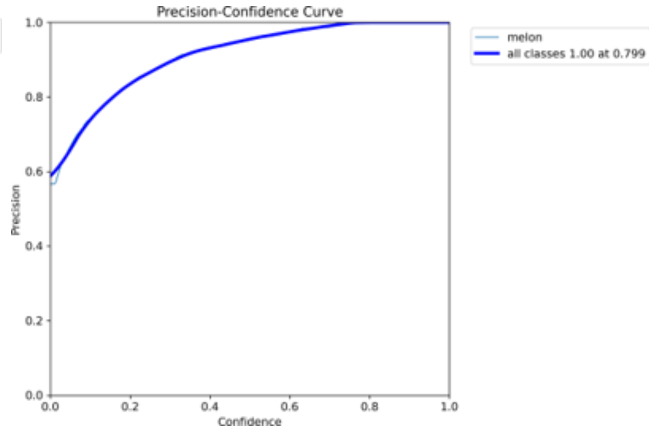
Curves and Confusion Matrix

To further interpret the behavior of the best-performing detector, we report precision–recall curves, confidence-dependent precision-recall curves, the F1 curve (Fig. 4), and the confusion matrix (Fig. 5) generated during evaluation. These artifacts provide complementary information beyond aggregate mAP values, showing how detector behavior changes across confidence thresholds (Fig. 4) and how predictions are distributed between the target class and background (Fig. 5). In a single-class problem such as this, the confusion matrix primarily reflects the balance between correct detections, missed detections, and background false positives.

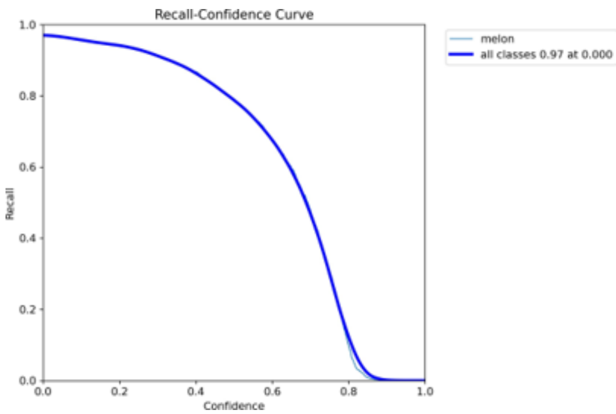
Error patterns. The combination of high mAP@0.5 and noticeably lower mAP@0.5:0.95 suggests that many detections are correct at a coarse level but become less reliable under stricter overlap requirements. This pattern is compatible with scenes where bounding-box placement is sensitive to object adjacency, partial occlusion, and ambiguous fruit boundaries. Therefore, the principal limitation revealed by the current benchmark is not the inability to detect melon presence, but rather the difficulty of tight localization in dense small-object imagery.



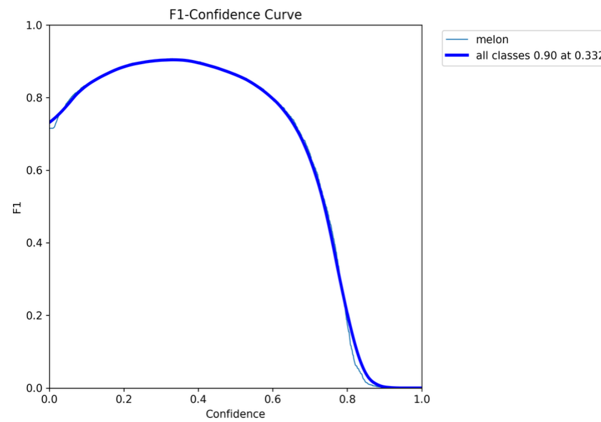
(a) PR curve



(b) Precision curve



(c) Recall curve



(d) F1 curve

Fig. 4. Evaluation curves for the best-performing model.

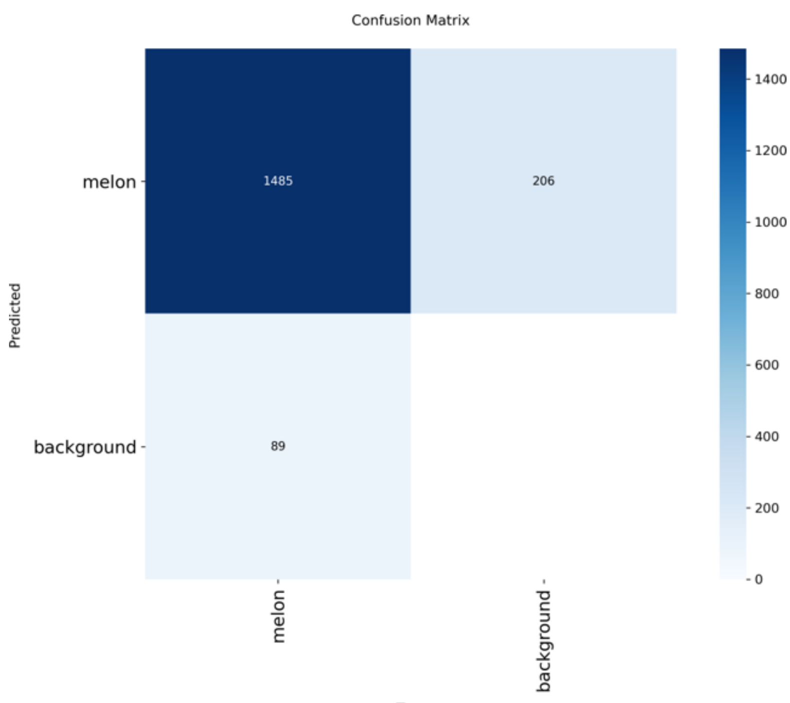


Fig. 5. Confusion matrix for the best-performing model.

Discussions

The experimental results indicate that YOLO-based detectors are effective for melon detection in dense UAV imagery, with all evaluated models achieving high test mAP@0.5 values. This suggests that most fruits can be detected under moderate overlap criteria, even in scenes containing a large number of small and closely spaced objects. However, the clear drop from mAP@0.5 to mAP@0.5:0.95 shows that precise localization remains substantially more difficult than coarse detection. In practical terms, the models often identify the presence of melons correctly, but bounding-box alignment becomes less reliable under stricter IoU requirements (Lin et al., 2014).

A plausible explanation for this behavior lies in the visual characteristics of the dataset. The images contain a very high density of small objects per frame, which increases the probability of neighboring fruits having partially overlapping or visually ambiguous boundaries. Similar challenges have been reported in UAV-based fruit detection studies, where small target size, variable illumination, and object clustering make localization less robust under field conditions (Khan et al., 2025; Zhao et al., 2024). In this scenario, even relatively small localization deviations can strongly affect mAP@0.5:0.95 without necessarily producing a large drop in mAP@0.5.

Another important point is that the benchmarked models achieved relatively close scores, especially under the primary ranking metric. This suggests that, under the current dataset conditions, performance is not determined only by the nominal model family but also by broader factors such as annotation consistency, image diversity, target scale, and split design. Therefore, although YOLO26l obtained the highest test mAP@0.5:0.95 in the current setup, the result should be interpreted as the best performance observed in this specific benchmark rather than as a definitive statement that one architecture is universally superior.

This study also has important limitations. First, the dataset is relatively small in terms of image count and appears to represent a narrow acquisition context, which restricts conclusions about generalization across fields, seasons, cultivars, and imaging conditions. Second, the current split was performed at image level, which may not fully eliminate spatial correlation between training and test samples in UAV surveys. Third, although the broader application motivating this work is counting, the present evaluation focuses exclusively on detection metrics; therefore, the results should be interpreted as a detection benchmark rather than a direct validation of counting accuracy or yield estimation.

From an applied perspective, the current results already provide a useful baseline for detection-driven agricultural analysis in still UAV imagery. However, future work should focus on improving localization quality and cross-condition robustness. Promising directions include expanding the dataset with more diverse flights and fields, adopting stricter spatial split strategies, increasing effective input resolution through tiling or multi-scale processing, and enforcing tighter annotation consistency checks. Once the detection stage becomes more robust, the next logical extension is to evaluate counting explicitly through per-image or per-plot error metrics and, eventually, to integrate temporal reasoning or tracking when repeated UAV passes or video data are available (Wang et al., 2025). Overall, the main contribution of the present study is to establish a reproducible and transparent benchmark that clarifies both the potential and the current limitations of UAV-based melon detection in dense field imagery.

Conclusions

This paper presented a reproducible benchmark for UAV-based melon detection using YOLO-based object detectors in dense field imagery. The study organized the dataset preparation workflow, standardized train/validation/test generation, and compared multiple detector variants under the same protocol. The best-performing model achieved strong test performance at IoU=0.5, while the lower mAP@0.5:0.95 values revealed that precise localization remains a major challenge in scenes with small, dense, and partially occluded objects.

The main contribution of this work is not only the benchmark itself, but also the establishment of a reproducible experimental baseline for future research on UAV-based melon analysis. Although the present evaluation is limited to detection, the results provide a useful foundation for subsequent studies on counting accuracy, spatial aggregation, and field-scale estimation. Future work should expand dataset diversity, adopt stricter spatial split strategies, and directly evaluate counting error in order to support more robust deployment in precision agriculture.

Acknowledgments

The authors thank the UNIOESTE, UTFPR, and the Multi-user Agricultural Technology Laboratory (AGRILAB-MD).

References

Buslaev, A., Iglovikov, V. I., Khvedchenya, E., Parinov, A., Druzhinin, M., & Kalinin, A. A. (2020). Albumentations: Fast and flexible image augmentations. *Information*, 11(2), 125. https://doi.org/10.3390/info11020125

Jiang, L., Jiang, H., Jing, X., Dang, H., Li, R., Chen, J., Majeed, Y., Sahni, R., & Fu, L. (2024). UAV-based field watermelon detection and counting using YOLOv8s. *Artificial Intelligence in Agriculture*, 13, 117–127. https://doi.org/10.1016/j.aiia.2024.07.001

Khan, Z., Shen, Y., & Liu, H. (2025). Object detection in agriculture: A comprehensive review of methods, applications, challenges, and future directions. *Agriculture*, 15(13), 1351.

Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., & Zitnick, C. L. (2014). *Microsoft COCO*: Common objects in context. arXiv preprint arXiv:1405.0312.

MLflow Contributors. (2026). *MLflow documentation*: Tracking and experiment management. https://mlflow.org/docs/latest/index.html. Accessed 8 Mar 2026.

Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (CVPR) (pp. 779–788).

Roboflow. (2025). *Roboflow documentation*: Dataset export (COCO JSON). https://docs.roboflow.com/. Accessed 8 Mar 2026.

Tsouros, D. C., Bibi, S., & Sarigiannidis, P. G. (2019). A review on UAV-based applications for precision agriculture. *Information*, 10(11), 349. https://doi.org/10.3390/info10110349

Ultralytics. (2025a). *Ultralytics YOLO dataset YAML format*. https://docs.ultralytics.com/datasets/detect/. Accessed 8 Mar 2026.

Ultralytics. (2025b). *Ultralytics YOLO repository*. https://github.com/ultralytics/ultralytics. Accessed 8 Mar 2026.

Wahab, I., Hall, O., & Jirström, M. (2018). Remote sensing of yields: Application of UAV imagery-derived NDVI for estimating maize vigor and yields in complex farming systems in sub-Saharan Africa. *Drones*, 2(3), 28. https://doi.org/10.3390/drones2030028

Wang, K., Kooistra, L., Wang, W., & Valente, J. (2025). UAV-assisted multi-object tracking and segmentation of apples under occlusions in orchard settings. The International Archives of the Photogrammetry, *Remote Sensing and Spatial Information Sciences*, XLVIII-2/W11-2025, 293–298.

Zhao, Y., Li, Y., & Xu, X. (2024). Object detection in high-resolution UAV aerial remote sensing images of blueberry canopy fruits. *Agriculture*, 14(10), 1842.