

RESEARCH ARTICLE

OPEN ACCESS

BENCHMARKING YOLO VARIANTS FOR DETECTING VARROA MITES IN BEEHIVE IMAGES

Youcef Naas¹, Slim rouabah², rabah abdelkader³, Meziane Kaci⁴, Salem Merabti⁵, Hassane Ezziane⁶

¹Department of Electronic and communication, Faculty of Sciences and Technology, University of Djilali Bounaama, Khemis Miliana, Ain Defla, Algeria.

²Djilali Bounaama University of Khemis Miliana, Algeria

³Department of Electronic and communication, Faculty of Sciences and Technology, University of Djilali Bounaama, Khemis Miliana, Ain Defla, Algeria.

⁴Department of Electrical Engineering, Faculty of Sciences and Technology, Djilali Bounaama University of Khemis Miliana, Ain Defla, Algeria.

⁵Department of Acoustics and Civil Engineering, Faculty of Sciences and Technology, Djilali Bounaama University of Khemis Miliana, Ain Defla, Algeria.

⁶Research Laboratory of Electrical Engineering and Automatics LREA, Yahia Fares University, Medea, Algeria.

¹<https://orcid.org/0009-0003-0653-728X>, ²<https://orcid.org/0009-0008-2295-9470>, ³<https://orcid.org/0000-0002-3738-3485>

⁴<https://orcid.org/0009-0008-9023-8173>, ⁵<https://orcid.org/0000-0002-4615-2179>, ⁶<https://orcid.org/0000-0002-4875-3226>

E-mail: y.naas@univ-dbkm.dz, s.rouabah@univ-dbkm.dz, r.abdelkader@univ-dbkm.dz, m.kaci@univ-dbkm.dz, s.merabti@univ-dbkm.dz, ezziane.hassan2051@gmail.com.

ARTICLE INFO

Article History

Received on: March 03, 2026

Reviewed on: April 02, 2026

Accepted on: May 01, 2026

Published on: July 31, 2026

Keywords:

V. destructor,
Beehive monitoring,
Artificial Intelligence,
Deep Learning,
YOLO,
Object detection.

ABSTRACT

The Varroa destructor mite drives major honey bee colony losses and economic damage in apiculture. Early detection supports timely intervention, yet manual inspection is slow and prone to error due to the mite's small size and dense hive conditions. Automated vision systems offer a scalable solution for continuous monitoring. This study benchmarks three YOLO variants, YOLOv7, YOLOv8, and YOLOv11, for automated Varroa detection. YOLOv11 achieves the highest performance, with mAP at 0.5 of 0.941 and recall of 87.9 percent, outperforming earlier versions. The shift from anchor-based detection in YOLOv7 to anchor free designs in YOLOv8 and YOLOv11, along with improved feature extraction through C2f modules and attention mechanisms, enhances small object detection critical for identifying mites. YOLOv8 provides the fastest training speed, operating twice as fast as YOLOv11 and 4.5 times faster than YOLOv7, which supports rapid deployment. All modern variants exceed 93 percent mAP at 0.5 and maintain real time inference at 30 frames per second or higher. Results position YOLOv11 as the preferred model for maximum accuracy and YOLOv8 as an efficient option for resource constrained beekeeping applications.



Copyright ©2026 by authors and Galileo Institute of Technology and Education of the Amazon (ITEGAM). This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

I. INTRODUCTION

Honeybees' contribution to the pollination of over 75% of flowering plants and 35% of the world's food crops demonstrates their critical role in global agriculture and ecosystems [1]. However, as any living being, honeybees face significant threats from various factors, such as pesticides, habitat loss, and pathogens. Among these, the Varroa destructor mite, also called V. destructor, is considered one of the most devastating pests, causing both widespread colony losses and worldwide economic damage [2]. V. destructor mites feed on the bodily fluids of bees, transmitting harmful viruses such as Deformed Wing Virus (DWV) and Acute Bee Paralysis Virus (ABPV), and weakening their immune systems, leading to colony collapse if left untreated [3]. Manual detection of Varroa destructor mites by beekeepers presents several critical challenges that fundamentally limit its effectiveness in modern apiculture. First, visual limitations pose a significant barrier, as these mites measure merely 1–2 mm in length and are frequently concealed between bee body segments, on the undersides of bees, or deep within brood cells, rendering visual identification exceptionally difficult even for experienced observers.

Second, time and labor constraints place considerable burden on beekeeping operations, since traditional monitoring approaches—such as sticky boards or sugar shake tests—demand substantial time investments and require frequent hive inspections that inevitably disturb colony activity and elevate stress levels among the bees. Third, human error and subjectivity introduce considerable variability into the monitoring process; manual counting is susceptible to fatigue-induced inaccuracies, inconsistent interpretation between different observers, and difficulties in maintaining rigorous monitoring schedules, particularly within commercial enterprises managing hundreds of hives simultaneously. Finally, delayed detection represents perhaps the most concerning limitation, as visible symptoms often manifest only after mite infestation levels have already reached critical thresholds, by which point treatment interventions become significantly less effective. Collectively, these constraints generate an urgent and pressing need for accurate, automated, and scalable detection techniques capable of operating in real-time while providing robust early warning capabilities to protect honey bee colony health. Recent research has explored various technological approaches to address the Varroa detection challenge:

- Early automated approaches relied on classical image processing techniques. The Var-Gor device [4] achieved 100% detection accuracy in laboratory conditions using specialized imaging hardware, but required controlled lighting conditions and could not generalize to field environments. Morphological operators combined with handcrafted features showed promise [5], but struggled with the high variability in natural hive imagery, including varying lighting conditions, bee orientations, and background clutter.
- Traditional methods require extensive parameter tuning, are sensitive to environmental conditions, and cannot adapt to unseen scenarios without manual reconfiguration.
- Convolutional Neural Networks (CNNs) have been applied for mite detection with notable success. Sharma et al. applied ResNet-50 and Inception V3 for early V. destructor mite detection using data augmentation and CLAHE (Contrast Limited Adaptive Histogram Equalization), achieving 90.67% accuracy with ResNet-50. Scutaru et al. combined morphological operators with neural networks, training on 65,000 positive and 1.2 million negative samples, achieving errors below 1% when integrated into the BeeVS device [5].
- Classification-based approaches can determine the presence of mites but cannot localize them within images, limiting their utility for precise monitoring and treatment planning. Additionally, these methods typically process entire images rather than detecting individual mite instances.
- Object detection methods offer significant advantages by providing both classification and localization capabilities. Kriouile et al. proposed a modified training scheme for Mask R-CNN utilizing two separate datasets for bee and mite identification [6]. While Mask R-CNN achieves high accuracy, its two-stage architecture (region proposal followed by classification) results in slower inference speeds, making it less suitable for real-time applications.

Bilik et al. applied YOLOv5 to a specialized dataset of infected and healthy bees, achieving F1 scores of 0.874 and 0.727 for infected bees and mites respectively [7]. Narcia-Macias et al. developed the IntelliBeeHive system using YOLOv7-tiny, achieving 99.6% mite detection precision [8]. Many studies focus on a single YOLO version without systematic comparison across variants. Limited analysis of architectural differences and their impact on small object detection. Insufficient evaluation of the trade-off between accuracy and computational efficiency. Lack of comprehensive benchmarking against both traditional and modern detection methods. The You Only Look Once (YOLO) family of models was selected for this study based on several compelling advantages that align closely with the requirements of automated hive monitoring. First, real-time performance stands as a primary consideration; YOLO's single-stage architecture processes images in a single forward pass, enabling inference speeds ranging from 30 to 140 frames per second depending on the specific variant and hardware configuration—an essential capability for continuous, real-time hive surveillance. Second, localization accuracy distinguishes YOLO from mere classification approaches, as the model provides precise bounding box coordinates for each detected mite, thereby enabling quantitative infestation assessment and detailed spatial analysis within the colony.

Third, architectural evolution represents a significant motivation for this selection; the continuous development of YOLO from its initial version through to the current YOLOv11 has introduced substantial improvements in feature extraction mechanisms, anchor box design, and training strategies, making a systematic comparative analysis both timely and scientifically valuable. Fourth, edge deployment capability addresses practical deployment constraints, as modern YOLO variants offer lightweight versions such as nano and tiny models that can operate efficiently on resource-constrained devices, facilitating on-hive processing without dependency on cloud connectivity. Finally, generalization capabilities reduce barriers to implementation, since YOLO models pretrained on large-scale datasets such as COCO and ImageNet demonstrate robust transfer learning capabilities, thereby minimizing the requirement for massive domain-specific datasets in the apicultural context. This work aims to train and evaluate multiple versions of YOLO models on the same dataset [9], to develop a robust, real-time solution to assist beekeepers in monitoring hive health. The specific objectives are:

- Comparative Analysis: Systematically compare YOLOv7, YOLOv8, and YOLOv11 to detect V. destructor mites across different training epochs.
- Architectural Analysis: Explain the key architectural differences between YOLO variants and their implications for small object detection.
- Performance Benchmarking: Evaluate models using multiple metrics (precision, recall, mAP) to provide comprehensive performance characterization.
- State-of-the-Art Comparison: Compare YOLO variants against other deep learning approaches reported in literature.

The paper is structured as follows: Section 2 details the materials and methods including YOLO architecture description, Section 3 presents the experimental results and analysis including comparisons with other techniques, and Section 4 provides conclusions and future directions.

II. MATERIALS AND METHODS

The You Only Look Once (YOLO) architecture revolutionized object detection by framing it as a single regression problem, directly predicting bounding boxes and class probabilities from full images in one evaluation.

This section describes the general YOLO architecture and the specific innovations introduced in each version evaluated. The YOLO architecture comprises three fundamental components that work in concert to achieve efficient object detection. The backbone serves as the feature extraction engine, processing input images through successive convolutional layers to extract hierarchical features; early layers capture low-level visual information such as edges and textures, while deeper layers progressively encode high-level semantic information including object parts and shapes, with modern YOLO variants employing diverse backbone designs such as Darknet, CSPDarknet, and C2f modules to enhance representational capacity. The neck component functions as a feature aggregation mechanism, combining features extracted from different backbone layers to create rich multi-scale feature representations—an essential capability for detecting objects of varying sizes within the same image—with common implementations including Feature Pyramid Networks and Path Aggregation Networks that facilitate effective information flow across scales. Finally, the detection head processes these aggregated features to generate predictions, outputting bounding box coordinates in the form of center positions, width, and height, alongside objectness scores and class probabilities, with the architecture utilizing anchor boxes as predefined reference templates to guide the prediction of object shapes and dimensions. YOLOv7, released in 2022 by Wang and colleagues [10], introduced several notable architectural innovations that advanced the state of single-stage object detection. Central to these improvements is the Extended Efficient Layer Aggregation Network, which enhances gradient flow throughout the network by employing expand, shuffle, and merge cardinality techniques, thereby enabling the model to learn more diverse feature representations without compromising the integrity of the original gradient propagation path.

The architecture also addresses fundamental challenges in model scaling through the introduction of compound scaling methods specifically designed for concatenation-based models; these methods preserve optimal model properties during both upscaling and downscaling operations, overcoming the limitation that conventional scaling approaches typically degrade performance in architectures reliant on feature concatenation. Furthermore, YOLOv7 incorporates a suite of sophisticated training strategies collectively termed trainable bag-of-freebies, including planned re-parameterized convolution, coarse-to-fine lead head guided label assignment, and dynamic label assignment mechanisms for auxiliary heads, all of which enhance training efficiency and final model performance without increasing inference costs. The key specifications of YOLOv7 include an E-ELAN backbone built upon CSPDarknet, a PANet neck enhanced with SPPCSPC for multi-scale feature processing, an anchor-based detection head utilizing dynamic label assignment, and a standard input resolution of 640 by 640 pixels. YOLOv8, introduced by Ultralytics in 2023 [11], represents a significant architectural departure from its predecessors, incorporating several fundamental innovations that enhance both efficiency and detection performance. The architecture features the C2f module, which replaces the C3 module employed in YOLOv5 with a more efficient design that effectively combines high-level features with contextual information, thereby improving gradient flow throughout the network while simultaneously reducing computational overhead. Perhaps the most consequential change is the adoption of anchor-free detection; YOLOv8 abandons the traditional anchor box paradigm in favor of directly predicting object center points, an approach that simplifies the overall model architecture, reduces the number of hyperparameters requiring tuning, and notably improves generalization capabilities for objects exhibiting unusual aspect ratios.

Additionally, the implementation of a decoupled detection head represents a significant structural refinement, separating classification and regression tasks into independent branches that can optimize separately during training, ultimately yielding improved overall accuracy. The key specifications of YOLOv8 include a CSPDarknet backbone enhanced with C2f modules, a PANet neck featuring improved feature fusion mechanisms, a decoupled anchor-free detection head, a standard input resolution of 640 by 640 pixels, and a COCO mean average precision of 53.9% for the YOLOv8x variant. YOLO11, released in 2024 [12], focuses primarily on efficiency and performance optimization through several targeted architectural refinements. The model features an optimized backbone architecture that reduces computational overhead while preserving robust feature extraction capabilities, achieved through improved layer design and more efficient channel configurations. The neck structure has been similarly enhanced, refining feature aggregation mechanisms to operate with fewer parameters while simultaneously achieving superior multi-scale fusion efficiency. Additionally, YOLO11 incorporates attention modules that enable the model to focus selectively on relevant image regions, a capability particularly beneficial for detecting small objects that might otherwise be overlooked. The key specifications include an optimized CSPDarknet backbone, a refined PANet neck augmented with attention mechanisms, an improved decoupled detection head, a standard input resolution of 640 by 640 pixels, and a COCO mean average precision of 54.7% for the YOLOv11x variant. Table 1 shows the key differences between the three YOLO versions.

Table 1: Comparison of YOLO Architecture Features.

Feature	YOLOv7	YOLOv8	YOLO11
Backbone	E-ELAN + CSPDarknet	CSPDarknet + C2f	Optimized CSPDarknet
Detection	Anchor-based	Anchor-free	Anchor-free
Head Design	Coupled	Decoupled	Enhanced Decoupled
Key Innovation	E-ELAN, BoF	C2f, Anchor-free	Attention, Efficiency
Training Speed	Moderate	Fast	Fast

Source: Authors, (2026).

II.1 METHODOLOGICAL CONTRIBUTION

This study makes several distinct methodological contributions to the field of automated Varroa mite detection. First, we present a systematic YOLO evolution analysis that, unlike previous studies confined to single-version evaluations, provides the first comprehensive comparison of three consecutive YOLO generations—versions 7, 8, and 11—specifically tailored for Varroa detection; this approach enables quantification of architectural improvements across versions, identification of which specific innovations most benefit small object detection performance, and practical guidance for beekeeping practitioners on optimal version selection.

Second, we offer detailed training dynamics characterization by evaluating models at multiple training checkpoints spanning epochs 1, 50, and 100, thereby illuminating early training behavior and convergence patterns, assessing the stability of different architectural approaches, and determining optimal training durations for each variant. Third, we provide critical insights into small object detection, as Varroa mites represent a particularly challenging scenario for tiny target identification; our analysis reveals how specific architectural choices influence small object recall, characterizes the inherent trade-offs between precision and recall for diminutive targets, and establishes optimal confidence thresholds for practical field deployment. Finally, we deliver practical deployment guidance that extends beyond conventional accuracy metrics to encompass training time requirements—which prove critical for iterative model development—inference speed considerations for real-time applications, and model size alongside memory footprint constraints relevant to edge computing scenarios.

II.2 DATASET

The experiments utilize the "bees infected with V. destructor destructor" dataset sourced from Roboflow [9], which comprises a training set of 1,569 images and a validation set of 448 images containing 273 mite instances, with annotations provided in standard YOLO format bounding box notation. The image characteristics reflect realistic apicultural conditions, encompassing varied lighting conditions, diverse bee densities, and mites positioned across different locations within the frame. This dataset presents several inherent challenges that mirror authentic field scenarios, including occlusion where mites are frequently concealed by bee body parts, significant scale variation arising from mites positioned at different distances from the camera, substantial background clutter introduced by hive debris and densely packed multiple bees, and considerable lighting variations characteristic of natural hive environments.

II.3 EXPERIMENTAL SETUP

All models were trained under rigorously controlled identical conditions to ensure equitable comparison across architectural variants. The computational environment utilized Kaggle cloud infrastructure equipped with an NVIDIA Tesla P100 GPU featuring 16 GB of video memory and 29 GB of system RAM. The implementation framework comprised PyTorch with the official Ultralytics YOLO library, employing the Adam optimizer alongside categorical cross-entropy loss functions. Training proceeded for a total duration of 100 epochs, with intermediate checkpoints preserved at epochs 1, 50, and 100 to facilitate longitudinal analysis of training dynamics. Additional fixed hyperparameters included a batch size of 16 images, a standardized input resolution of 640 by 640 pixels, and the application of standard Ultralytics data augmentation techniques encompassing mosaic composition, mixup blending, and random horizontal flipping. Model performance is evaluated using established object detection metrics that capture different aspects of detection quality. Precision, in equation (1), measures the proportion of correctly detected objects among all detections generated by the model, thereby quantifying the effectiveness of false positive reduction, calculated as the ratio of true positives to the sum of true positives and false positives.

$$\text{Precision} = \text{TP} / (\text{TP} + \text{FP}) \quad (1)$$

Recall, in equation (2), represents the proportion of correctly detected objects relative to all ground truth instances present in the dataset, indicating the model's capability for false negative reduction, computed as true positives divided by the sum of true positives and false negatives.

$$\text{Recall} = \text{TP} / (\text{TP} + \text{FN}) \quad (2)$$

The mean average precision at an intersection over union threshold of 0.5 provides an aggregate measure of detection accuracy under moderate localization strictness, while the more stringent mAP averaged across IoU thresholds from 0.5 to 0.95 in increments of 0.05 demands precise localization for high scores. Intersection over union, in equation (3), serves as the fundamental overlap metric, quantifying the spatial correspondence between predicted and ground truth bounding boxes through the ratio of intersection area to union area.

$$\text{IoU} = \text{Area_Intersection} / \text{Area_Union} \quad (3)$$

III. RESULTS AND DISCUSSION

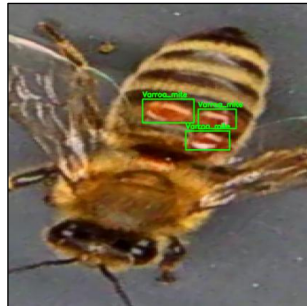
Table 2 summarizes the detection performance of YOLOv7, YOLOv8, and YOLO11 across training epochs 1, 50, and 100.

Table 2: Detection Performance of YOLOv7, YOLOv8, and YOLO11 Across Training Epochs.

Model	Epochs	Precision	Recall	mAP@0.5	mAP@0.5:0.95	Train Time (s)
YOLOv7	1	0	0	0	0	174.15
	50	0.932	0.249	0.244	0.107	4576.21
	100	0.901	0.630	0.624	0.282	8965.99
YOLOv8	1	0.884	0.139	0.515	0.206	28.8
	50	0.904	0.868	0.940	0.493	1011.6
	100	0.925	0.858	0.935	0.492	1990.8
YOLO11	1	0.847	0.465	0.674	0.256	34.28
	50	0.910	0.884	0.938	0.501	1169.21
	100	0.921	0.879	0.941	0.503	2350.04

Source: Authors, (2026).

At epoch 1, the analysis reveals substantial differences in initialization behavior across the three architectural variants. YOLOv7 exhibits no meaningful detection capability, with all performance metrics registering at zero, which indicates insufficient feature learning during the initial training phase and suggests that the E-ELAN architecture requires considerably more training iterations to establish effective feature representations. In contrast, YOLOv8 demonstrates notably strong precision at 0.884 but correspondingly weak recall at 0.139, indicating that the model generates highly confident predictions yet confines these to a limited subset of easily detectable mites; this pattern suggests that the anchor-free design combined with the decoupled head architecture enables faster initial learning of discriminative features. YOLO11 achieves the most favorable balance at this early stage, attaining precision of 0.847 alongside substantially higher recall of 0.465, which indicates that the optimized backbone architecture and integrated attention mechanisms facilitate rapid feature learning from the outset of training; the mAP@0.5 of 0.674 observed at epoch 1 is particularly impressive and underscores the efficiency of the architectural refinements introduced in this generation. By epoch 50, all three models demonstrate substantial performance improvements, though distinct behavioral patterns persist across architectures. YOLOv7 achieves high precision at 0.932 yet remains notably constrained in recall at 0.249, indicating that while the model rarely generates false positive detections, it fails to identify approximately three-quarters of actual mite instances present in the validation set; this represents a critical limitation for practical apicultural applications, where undetected infestations can rapidly escalate and lead to severe colony loss. YOLOv8 exhibits excellent overall performance, attaining recall of 0.868 and mAP@0.5 of 0.940, which demonstrates that the anchor-free design combined with C2f modules enables effective multi-scale feature learning capable of capturing diverse mite appearances and scales. YOLO11 achieves the highest recall among all variants at 0.884 alongside competitive mAP@0.5 performance of 0.938, thereby demonstrating superior capability in detecting challenging mite instances that prove elusive for other architectures. At epoch 100, YOLO11 achieves the highest overall performance among all evaluated variants, with precision of 0.921, recall of 0.879, mAP@0.5 of 0.941, and mAP@0.5:0.95 of 0.503. YOLO8 demonstrates strong competitive performance, attaining precision of 0.925, recall of 0.858, mAP@0.5 of 0.935, and mAP@0.5:0.95 of 0.492. YOLOv7, while achieving respectable precision of 0.901, lags considerably in recall at 0.630 and shows substantially lower detection accuracy with mAP@0.5 of 0.624 and mAP@0.5:0.95 of 0.282. The 24.9 percentage point recall advantage of YOLO11 over YOLOv7 is particularly significant for practical deployment scenarios, as failing to detect mites can precipitate severe consequences for overall hive health and colony survival.



(a)



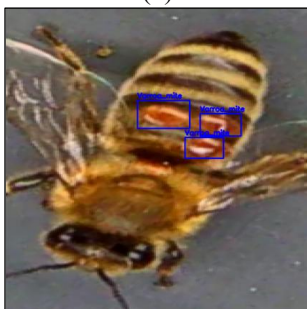
(b)



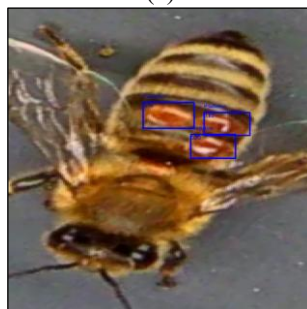
(c)



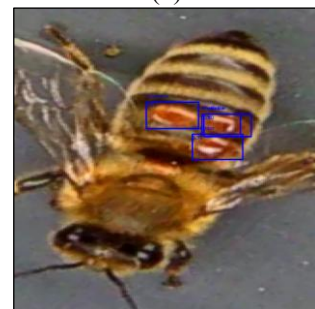
(d)



(e)



(f)



(g)

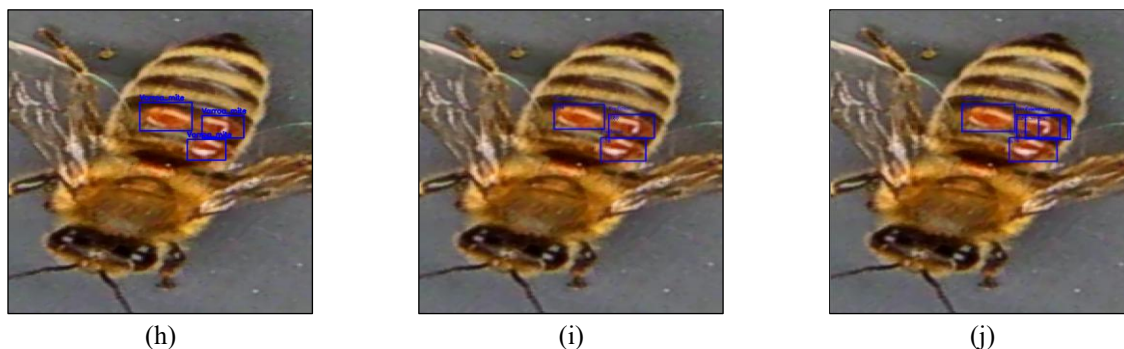


Figure 1: Varroa mite detection performance across YOLO models and training iterations. (a) Ground truth with manually annotated Varroa mite. Row 1: Model detections after 1 iteration. Row 2: Detections after 50 iterations. Row 3: Detections after 100 iterations. Columns correspond to YOLOv7, YOLOv8, and YOLOv11, respectively. Source: Authors, (2026).

Figures 1a through 1j collectively illustrate the detection performance trajectories across all three model architectures and training stages. Figure 1a presents the ground truth annotation, displaying expert-annotated bounding boxes in red rectangles surrounding Varroa mites on honey bee bodies, thereby establishing the reference standard against which all model predictions are evaluated. Figures 1b, 1e, and 1h depict YOLOv7's detection progression: at epoch 1 (Figure 1b), the model completely fails to detect any mites; by epoch 50 (Figure 1e), sporadic detections emerge though numerous instances remain undetected; and at epoch 100 (Figure 1h), performance improves modestly yet the model continues to miss small or occluded mites, visually corroborating the lower recall metrics observed quantitatively. Figures 1c, 1f, and 1i demonstrate YOLOv8's detection evolution: at epoch 1 (Figure 1c), the model generates confident detections on visually prominent mites; by epoch 50 (Figure 1f), the majority of mites are successfully detected with accurate localization; and at epoch 100 (Figure 1i), performance stabilizes at a high precision level. Figures 1d, 1g, and 1j showcase YOLOv11's superior and most consistent improvement trajectory: at epoch 1 (Figure 1d), initial detections are already reasonably accurate; by epoch 50 (Figure 1g), performance approaches full convergence; and at epoch 100 (Figure 1j),

YOLOv11 achieves the most comprehensive detection coverage with precisely localized bounding boxes across diverse mite instances. Training time represents an important practical consideration for model deployment and iterative development. YOLOv8 exhibits the fastest training duration, completing 100 epochs in 1,990.8 seconds, a characteristic particularly advantageous for rapid prototyping and extensive hyperparameter tuning cycles. YOLOv11 demonstrates moderate training time at 2,350.04 seconds, thereby offering the most favorable accuracy-time trade-off among the three variants. YOLOv7 proves substantially slower, requiring 8,965.99 seconds to complete equivalent training—approximately 4.5 times longer than YOLOv8—which significantly constrains iterative development workflows. The accelerated convergence observed in both YOLOv8 and YOLOv11 can be attributed to several architectural factors: the implementation of more efficient C2f modules in contrast to the more computationally demanding E-ELAN architecture employed in YOLOv7, the anchor-free design which simplifies the learning objective by eliminating anchor box optimization, and improved optimization dynamics stemming from decoupled detection heads that allow independent optimization of classification and regression tasks.

IV. CONCLUSION

This study presents a comprehensive benchmarking analysis of YOLO variants for automated Varroa destructor mite detection, yielding several key findings with significant implications for both research and practical apiculture. YOLOv11 demonstrates clear superiority among evaluated variants, achieving the highest detection performance with mAP@0.5 of 0.941 and recall of 87.9%, representing a substantial advancement over both YOLOv7 and YOLOv8. The architectural impact of evolving from anchor-based detection in YOLOv7 to anchor-free approaches in YOLOv8 and YOLOv11, combined with improved feature extraction through C2f modules and integrated attention mechanisms, directly translates to enhanced small object detection capability critical for identifying diminutive mites. Regarding training efficiency, YOLOv8 offers the optimal training speed, operating twice as fast as YOLOv11 and 4.5 times faster than YOLOv7, thereby making it particularly suitable for rapid deployment scenarios where maximum accuracy is not the paramount concern. All modern YOLO variants achieve mAP@0.5 exceeding 93%, performance levels sufficient for practical beekeeping applications, while maintaining real-time inference capability at 30 frames per second or higher to enable continuous hive monitoring.

This work provides the first systematic comparison of YOLO evolution specifically for agricultural pest detection, demonstrating conclusively that recent architectural innovations significantly benefit small object detection tasks. For practitioners, we recommend deploying YOLOv11 for maximum detection accuracy in critical applications, utilizing YOLOv8 for rapid deployment and resource-constrained environments, and avoiding YOLOv7 for new deployments due to its inferior recall characteristics and protracted training requirements. Future research directions include integration of the detection framework with automated treatment systems, validation across diverse hive environments and seasonal conditions, development of lightweight models for edge deployment on Raspberry Pi or comparable devices, and exploration of multi-modal detection combining visual and thermal imaging modalities. These artificial intelligence-powered tools hold substantial promise for supporting honeybee health monitoring, enhancing hive management practices, and ultimately benefiting pollination-dependent agriculture and broader ecological sustainability.

V. AUTHOR'S CONTRIBUTION

Conceptualization: Youcef Naas, Slim rouabah, rabah abdelkader, Meziane Kaci, Salem Merabti, Hassane Ezziane.

Methodology: Youcef Naas, Slim rouabah, rabah abdelkader, Meziane Kaci, Salem Merabti, Hassane Ezziane.

Investigation: Youcef Naas, Slim rouabah, rabah abdelkader, Meziane Kaci, Salem Merabti, Hassane Ezziane.

Discussion of results: Youcef Naas, Slim rouabah, rabah abdelkader, Meziane Kaci, Salem Merabti, Hassane Ezziane.

Writing – Original Draft: Youcef Naas, Slim rouabah, rabah abdelkader, Meziane Kaci, Salem Merabti, Hassane Ezziane.

Writing – Review and Editing: Youcef Naas, Slim rouabah, rabah abdelkader, Meziane Kaci, Salem Merabti, Hassane Ezziane.

Resources: Youcef Naas, Slim rouabah, rabah abdelkader, Meziane Kaci, Salem Merabti, Hassane Ezziane.

Supervision: Youcef Naas, Slim rouabah, rabah abdelkader, Meziane Kaci, Salem Merabti, Hassane Ezziane.

Approval of the final text: Youcef Naas, Slim rouabah, rabah abdelkader, Meziane Kaci, Salem Merabti, Hassane Ezziane.

VI. REFERENCES

- [1] A. Klein, B. Vaissière, J. Cane, I. Steffan-Dewenter, S. Cunningham, C. Kremen, and T. Tscharntke, "Importance of pollinators in changing landscapes for world crops," *Proc. R. Soc. B: Biol. Sci.*, vol. 274, pp. 303–313, 2006, doi: 10.1098/rspb.2006.3721.
- [2] P. Rosenkranz, P. Aumeier, and B. Ziegelmann, "Biology and control of *Varroa destructor*," *Journal of Invertebrate Pathology*, vol. 103, suppl., pp. S96–S119, 2010, doi: 10.1016/j.jip.2009.07.016.
- [3] F. Nazzi and Y. Le Conte, "Ecology of *Varroa destructor*, the major ectoparasite of the western honey bee, *Apis mellifera*," *Annu. Rev. Entomol.*, vol. 61, pp. 417–432, 2016, doi: 10.1146/annurev-ento-010715-023731.
- [4] V. Dietemann, F. Nazzi, S. J. Martin, D. L. Anderson, B. Locke, K. S. Delaplane, *et al.*, "Standard methods for varroa research," *J. Apic. Res.*, vol. 52, no. 1, pp. 1–54, 2013, doi: 10.3896/IBRA.1.52.1.09.
- [5] D. Scutaru, S. Bergonzoli, C. Costa, S. Violino, C. Costa, S. Albertazzi, V. Capano, M. M. Kostić, and A. Scarfone, "An AI-based digital scanner for *Varroa destructor* detection in beekeeping," *Insects*, vol. 16, no. 1, p. 75, 2025, doi: 10.3390/insects16010075.
- [6] Y. Kriouile, C. Ancourt, K. Wegrzyn-Wolska, *et al.*, "Nested object detection using mask R-CNN: Application to bee and varroa detection," *Neural Comput. Appl.*, vol. 36, pp. 22587–22609, 2024, doi: 10.1007/s00521-024-10393-x.
- [7] S. Bilik, L. Kratochvila, A. Ligocki, O. Bostik, T. Zemcik, M. Hybl, K. Horak, and L. Zalud, "Visual diagnosis of the *Varroa destructor* parasitic mite in honeybees using object detector techniques," *Sensors*, vol. 21, no. 8, p. 2764, 2021, doi: 10.3390/s21082764.
- [8] C. I. Narcia-Macias *et al.*, "IntelliBeeHive: An Automated Honey Bee, Pollen, and Varroa Destructor Monitoring System," *2024 International Conference on Machine Learning and Applications (ICMLA)*, Miami, FL, USA, 2024, pp. 845–850, doi: 10.1109/ICMLA61862.2024.00122.
- [9] MetinBarlas. 2023. VarroaDetection dataset. Roboflow. <https://universe.roboflow.com/metinbarlas/varroadetection-hc3zl>
- [10] C. -Y. Wang, A. Bochkovskiy and H. -Y. M. Liao, "YOLOv7: Trainable Bag-of-Freebies Sets New State-of-the-Art for Real-Time Object Detectors," *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Vancouver, BC, Canada, 2023, pp. 7464–7475, doi: 10.1109/CVPR52729.2023.00721.
- [11] V. Afifah and S. Erniwati, "YOLOv8 for object detection: A comprehensive review of advances, techniques, and applications," *Int. J. Adv. Comput. Inform.*, vol. 2, pp. 53–61, 2025, doi: 10.71129/ijaci.v2i1.pp53-61.
- [12] R. Khanam and M. Hussain, "YOLOv11: An overview of the key architectural enhancements," *arXiv preprint arXiv:2410.17725*, 2024. [Online]. Available: <https://arxiv.org/abs/2410.17725>