



Article

AI-Driven Insect Detection, Real-Time Monitoring, and Population Forecasting in Greenhouses

Dimitrios Kapetas ¹, Panagiotis Christakakis ¹, Sofia Faliagka ², Nikolaos Katsoulas ^{2,*} and Eleftheria Maria Pechlivani ¹

¹ Centre for Research and Technology Hellas, Information Technologies Institute, 57001 Thessaloniki, Greece; dimikape@iti.gr (D.K.); christakakis@iti.gr (P.C.); riapechl@iti.gr (E.M.P.)

² Laboratory of Agricultural Constructions and Environmental Control, Department of Agriculture Crop Production and Rural Environment, University of Thessaly, 38446 Volos, Greece; sofaliag@uth.gr

* Correspondence: nkatsoul@uth.gr; Tel.: +30-24210-93249

Abstract: Insecticide use in agriculture has significantly increased over the past decades, reaching 774 thousand metric tons in 2022. This widespread reliance on chemical insecticides has substantial economic, environmental, and human health consequences, highlighting the urgent need for sustainable pest management strategies. Early detection, insect monitoring, and population forecasting through Artificial Intelligence (AI)-based methods, can enable swift responsiveness, allowing for reduced but more effective insecticide use, mitigating traditional labor-intensive and error prone solutions. The main challenge is creating AI models that perform with speed and accuracy, enabling immediate farmer action. This study highlights the innovating potential of such an approach, focusing on the detection and prediction of black aphids under state-of-the-art Deep Learning (DL) models. A dataset of 220 sticky paper images was captured. The detection system employs a YOLOv10 DL model that achieved an accuracy of 89.1% (mAP_{50}). For insect population prediction, random forests, gradient boosting, LSTM, and the ARIMA, ARIMAX, and SARIMAX models were evaluated. The ARIMAX model performed best with a Mean Square Error (MSE) of 75.61, corresponding to an average deviation of 8.61 insects per day between predicted and actual insect counts. For the visualization of the detection results, the DL model was embedded to a mobile application. This holistic approach supports early intervention strategies and sustainable pest management while offering a scalable solution for smart-agriculture environments.

Keywords: insect detection; deep learning; insect population prediction; machine learning; black aphids; mobile application; pests



Academic Editors: Sotirios K. Goudos and Lilong Chai

Received: 1 December 2024

Revised: 21 December 2024

Accepted: 20 January 2025

Published: 27 January 2025

Citation: Kapetas, D.; Christakakis, P.; Faliagka, S.; Katsoulas, N.;

Pechlivani, E.M. AI-Driven Insect Detection, Real-Time Monitoring, and Population Forecasting in Greenhouses. *AgriEngineering* 2025, 7, 29. <https://doi.org/10.3390/agriengineering7020029>

Copyright: © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Insect infestations are a critical issue in agriculture, with significant impacts on crop productivity, economic stability, and environmental sustainability. The Food and Agriculture Organization of the United Nations (FAO) reports that each year, pests are responsible for up to 40% of global crop losses, leading to economic damages exceeding USD 70 billion [1]. These challenges are further intensified by climate change, which contributes to increased outbreaks of diseases, pests, and viruses, as well as the development of pesticide resistance in insect populations [2–5]. To mitigate these losses, farmers have historically relied heavily on chemical pesticides. However, this approach has raised concerns due to its environmental repercussions, potential health risks, and challenges in maintaining long-term agricultural sustainability [6]. Therefore, early and precise detection of pests

using technological tools is essential for deploying timely management strategies that help reduce both economic and environmental costs [7].

Conventional pest monitoring methods, such as manual counting on traps and visual inspections, are still widely practiced but are often labor-intensive, prone to human error, and may delay crucial interventions [8]. These limitations can result in unchecked infestations, which increase the need for repeated pesticide applications, compounding costs, and ecological impact [9]. Recent advancements in Artificial Intelligence (AI) and the Internet of Things (IoT) offer promising solutions through automated pest detection systems that enhance efficiency and accuracy, providing scalable alternatives for agricultural monitoring [10,11].

In particular, Deep Learning (DL) models have shown considerable promise in identifying specific insect species from images captured by cameras [12]. In recent years, Convolutional Neural Networks (CNNs) in DL have driven significant advancements in computer vision, particularly in the area of general object detection [13]. Various DL architectures, including YOLO (You Only Look Once) [14], have proven effective in detecting small, challenging targets like insects.

Several studies have employed these methods for pest detection in agricultural settings. For example, in a recent study Kumar et al. (2023) [15] developed a YOLOv5-based insect detection system, enhanced with attention modules to improve recognition accuracy, achieving a mean Average Precision (*mAP*) of 93% on a custom pest dataset. Similarly, Verma et al. (2021) [16] applied YOLO v3, v4, and v5 algorithms for insect detection in soybean crops, finding YOLO v5 to be the most accurate, with a *mAP* of 99.5%. Liu et al. (2019) [17] proposed PestNet, a DL framework that achieved 75.46% *mAP* for multi-class pest detection on 80,000 labeled pest images, using a region-based end-to-end approach. Zhong et al. (2018) [18] developed a DL system using a multi-class classifier to identify and count six species of flying insects, utilizing a modified YOLO framework along with image augmentations to enhance the dataset.

In another study, Giakoumoglou et al. (2022) [19] evaluated YOLO-based models for identifying black aphids and whiteflies on adhesive traps, achieving a *mAP* of 75%, while in a more recent study [20] they introduced a synthetic data generation method, “Generate-Paste-Blend-Detect”, for agricultural object detection, achieving a *mAP* of 66% for whiteflies using YOLOv8, effectively reducing the need for extensive annotated datasets. Xie et al. (2015) [21] introduced a crop insect recognition method using sparse representations and multiple-kernel learning, achieving 97% accuracy across 24 insect classes. In another relevant study, Liu and Wang (2020) [22] created a dataset featuring tomato pests and diseases for detection in natural settings, comprising 15,000 images across 12 distinct classes. Among various models tested, an improved YOLOv3 with Darknet53 yielded the best performance, achieving a *mAP* of 92.39%. Lastly, Gutierrez et al. (2019) [23] assessed computer vision, ML, and DL techniques for pest detection in tomato farms, finding DL to be the most effective approach.

Forecasting insect population growth has also become an important aspect of pest management, as it allows for proactive intervention strategies. Machine Learning (ML) and time-series models, which incorporate environmental factors like temperature and humidity, have proven valuable in enhancing prediction accuracy [24]. Recent research emphasizes various approaches to pest population prediction, exploring both statistical and ML models in diverse agricultural contexts. Bahlai (2023) [25] highlighted the complexities in forecasting insect dynamics due to species diversity, environmental variability, and the limitations of the current modeling techniques. Integrated single-species monitoring and near-term iterative approaches show promise for improving accuracy while balancing generalization. Marković et al. (2021) [26] demonstrated that incorporating extended

weather data improved pest occurrence predictions, achieving an accuracy of 86.3% for insect detection over five-day periods. Skawsang et al. (2019) [27] applied Artificial Neural Networks (ANNs), random forests, and multiple linear regression to predict brown planthopper populations in rice fields using meteorological and satellite-derived crop phenology data. The ANN model proved most accurate, achieving a Root Mean Square Error (RMSE) of 1.686, outperforming both random forests and linear regression. Similarly, Rathod et al. (2021) [28] developed a climate-based prediction model for Asian rice gall midge populations, with the ANN model achieving the best results.

For greenhouse pest management, Chiu et al. (2019) [29] used ARIMA and ARIMAX models to forecast greenhouse whitefly populations, incorporating environmental data. The ARIMAX model proved the most effective, achieving an RMSE of approximately 1.30 for 7-day forecasts and providing valuable insights for pesticide scheduling. Kawakita and Takahashi (2022) [30] further demonstrated the potential of seasonal ARIMAX in predicting common cutworm population dynamics. By incorporating past temperature data, especially during key developmental stages, the ARIMAX model provided reliable forecasts, making it a powerful tool for proactive pest management.

Building on these developments, this study presents an integrated approach for monitoring and forecasting black aphid populations in cucumber cultivations, leveraging DL, ML, and time-series models. The methodology combines the YOLO object detection framework, optimized for real-time identification of black aphids on adhesive traps, with the ARIMAX model to predict population trends based on environmental data. By enabling both immediate detection and forecasting, this system facilitates timely and informed pest management interventions, aiming to reduce reliance on chemical pesticides and improve crop protection. Additionally, this study explores the potential for integrating these models into mobile applications, promoting accessible, scalable solutions within Agriculture 4.0 and advancing sustainable pest management practices.

The use case of this study focuses on detecting black aphids in cucumber cultivations, a pest that poses serious risks to crop health and yield [31]. Black aphid infestations can lead to significant yield reductions, causing both economic losses and increased vulnerability to secondary infections [32,33]. Although there are aphid-resistant cucumber varieties, biological controls, and chemical pesticides, these measures alone often fall short of achieving effective pest management [34,35]. Therefore, digital technologies that enable early detection and timely alerts for growers are essential to prevent crop damage and support sustainable agricultural practices.

2. Materials and Methods

The methodology for this study involved the systematic collection of images using mobile phone cameras from late October to early December 2023. Five pheromone sticky paper traps were consistently maintained in greenhouse facilities of the Laboratory of Agricultural Constructions and Environmental Control, at the University of Thessaly (UTH), in the area of Velestino (latitude 39°44', longitude 22°79', altitude 85 m), Greece. Throughout the study period, the number of sticky paper traps remained the same, with a replacement occurring due to a high number of captured insects to ensure continued data quality. In parallel, sensors inside the greenhouse continuously captured environmental conditions, including ambient temperature, relative humidity, and barometric pressure. The imagery data were annotated, and the environmental data were transmitted to the Green Deal Decision Support System (DSS) of the H2020 PestNu project [36] at the Centre for Research and Technology Hellas (CERTH), forming the basis for developing AI models for insect detection and population forecasting. The insect detection model was integrated into a mobile application [37] to provide real-time monitoring for end users. Figure 1 presents a

high-level overview of the approach, illustrating the process from data acquisition in the greenhouse to the deployment of AI models for insect detection in a mobile application.

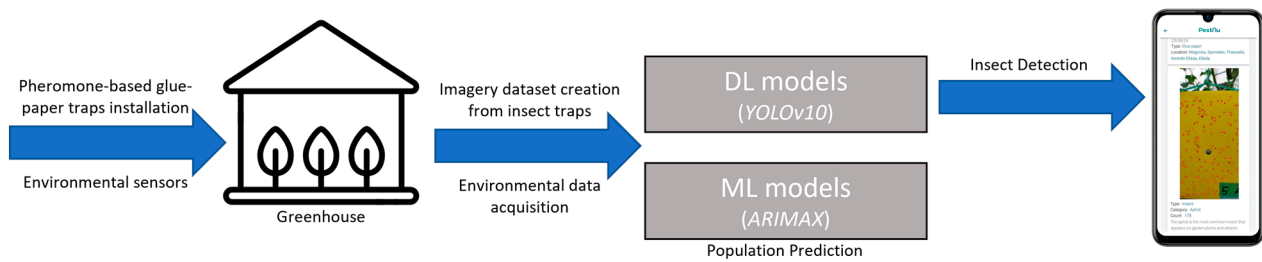


Figure 1. High-level approach architecture depicting the installation of pheromone-based sticky paper traps and environmental sensors in the greenhouse, data collection processes, and the development of AI models for insect detection (YOLOv10) and population prediction (ARIMAX). The insect detection results are integrated into a mobile application for real-time monitoring.

2.1. Image and Environmental Data Acquisition

During the data acquisition period, five pheromone-based sticky paper traps were consistently maintained within the greenhouse at the UTH facilities in Velestino, Greece. The number of traps remained constant throughout the study, with a single replacement made due to a high accumulation of insects on the sticky surface, ensuring consistent and reliable data collection. The traps were positioned systematically, and images were captured almost daily using mobile phone cameras from a distance of 30–40 cm. Image capturing occurred primarily on weekdays, with no data recorded during non-working days. This process resulted in a total of 220 images over the course of the 44-day study period (late October to early December) in a cucumber cultivation. Figure 2a illustrates the deployment of the sticky paper traps in the greenhouse, while Figure 2b shows an image captured with black aphids stuck on the pheromone-based sticky paper.

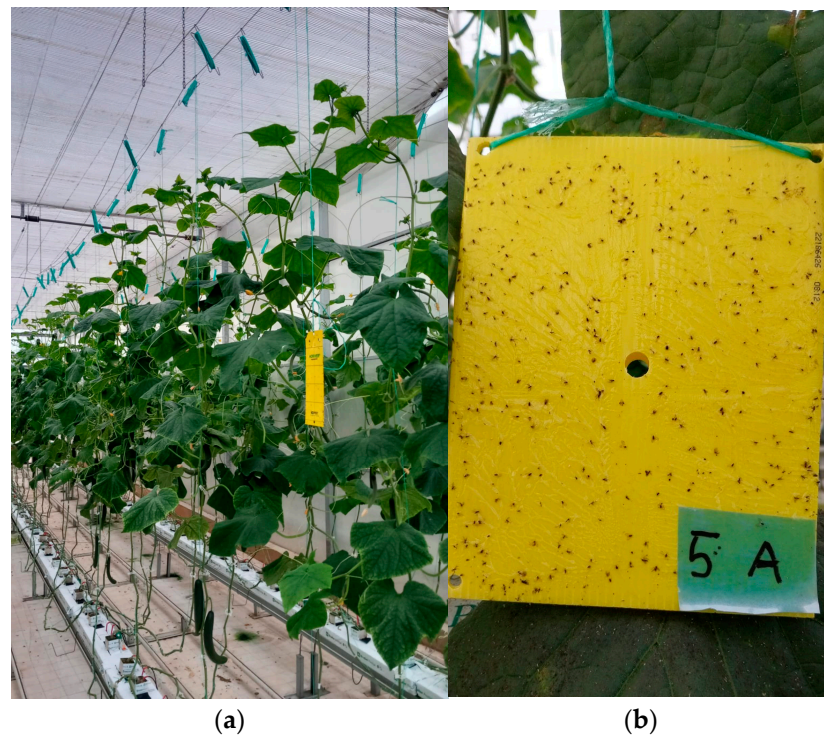


Figure 2. (a) Pheromone-based sticky paper traps deployed in the greenhouse facilities; (b) target insect black aphids stuck in the sticky paper trap.

The mobile phone cameras were positioned to capture the entire sticky paper surface, providing a comprehensive view of the captured insects for correct annotation. Alongside the imagery data, the environmental conditions inside the greenhouse were continuously monitored by deployed sensors. The environmental conditions in the greenhouse were automatically controlled and recorded every five minutes using a climate control computer (SERCOM, Automation SL, Lisse, The Netherlands). The sensors collected key data, including temperature, humidity, solar radiation, and atmospheric pressure. The environmental measurements were transmitted in real time to the DSS [36]. The integration of both imagery and environmental data into the DSS allowed for the subsequent development of AI models to detect black aphids and predict population dynamics. The dataset developed in the frame of this work is available for download at <https://zenodo.org/records/14097660> (accessed on 12 November 2024).

2.2. Image Annotation and Augmentation Techniques

In any DL model, the accuracy and reliability of the results heavily depends on the quantity and quality of the data used for training. Given that the original dataset consisted of only 220 images of black aphids captured on pheromone-based sticky paper traps, it was essential to annotate and expand the dataset to provide sufficient data for model training. The annotation process was performed using Roboflow [38], resulting in 13,357 total annotations of black aphids, averaging 60.71 annotations per image.

To ensure the robustness of the DL model, it was crucial to augment the original dataset, which contained a relatively small number of images. Data augmentation is a vital step before training, particularly when dealing with limited datasets, as it helps prevent overfitting and enables the model to generalize better to unseen data. In this study, augmentations were designed to simulate real-world conditions observed within greenhouse environments. By artificially expanding the dataset, the diversity of the training data was enhanced without the need for additional image collection. Various augmentation techniques were applied, with a particular focus on changes that reflect typical variations in a controlled greenhouse setting. Adjustments in brightness, saturation, and exposure were implemented to account for fluctuations in lighting throughout the day. These augmentations help the model become more resilient to different light intensities and shadows that may affect the visual appearance of the insects.

The augmentations were applied specifically to the training set to artificially triple its size, ensuring that the model could handle a variety of environmental factors and conditions it might encounter in practice. Therefore, the original dataset was split into training and validation subsets randomly, following an 80–20% ratio, which resulted in 175 images for training and 45 images for validation. By applying the aforementioned augmentation techniques to the training set resulted in 534 images, bringing the total dataset size to 579 images after augmentation. Table 1 provides a detailed overview of the dataset split after augmentation

Table 1. Final dataset split after applying augmentation to the training set.

Set	Images	Annotated Insects
Training	534	10,873
Validation	45	2484
Total	579	13,357

Figure 3 presents a histogram of the object count per image, illustrating the distribution of black aphid annotations across the dataset and helping to understand the varying insect densities in the captured images. The combination of annotation, augmentation, and

preprocessing allowed for the creation of a robust dataset that is suitable for training DL models to detect black aphids effectively.

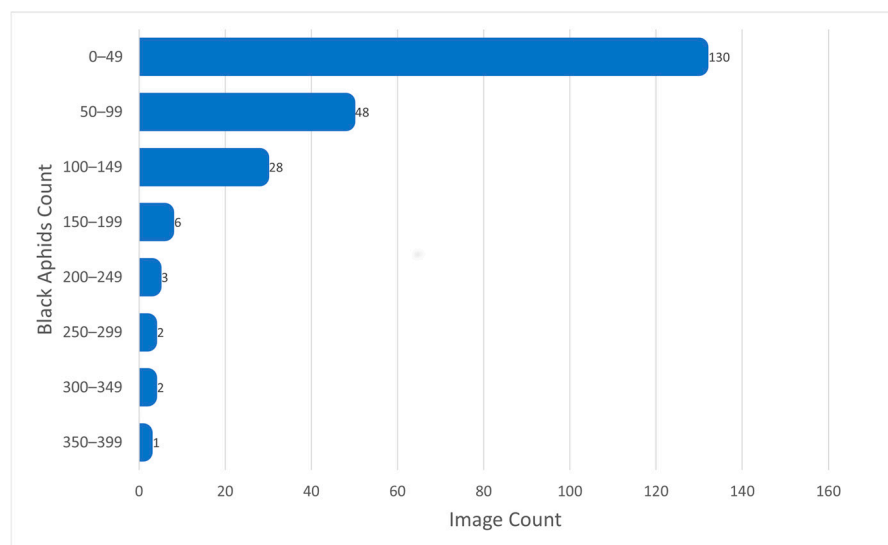


Figure 3. Histogram of black aphids counts.

2.3. Deep Learning-Based Insect Detection

In this study, the task of detecting black aphids was addressed using several versions of the YOLO object detection algorithm [14]. Specifically, three versions of the YOLO algorithm were utilized: YOLOv5 [39], YOLOv8 [40], and YOLOv10 [41]. For each of them, the largest variants ('large' and 'xlarge') were chosen for training in order to achieve the best possible detection precision and processing speed, making them suitable for different operational environments. The YOLO framework has become well-established in the field of object detection due to its ability to handle this task with a single pass through the network, significantly improving inference times when compared to two-stage models, which first generate regions of interest and then classify them. By treating detection as a unified task, YOLO has been widely adopted for scenarios that demand fast and efficient detection, including real-time pest monitoring in agriculture.

The YOLO algorithm has undergone continuous development through multiple versions, with each version enhancing its ability to detect smaller objects, such as insects, which pose unique challenges in computer vision. All YOLO models—YOLOv5, YOLOv8, and YOLOv10—are well-suited for small object detection, making them effective for insect detection tasks. Each version brings its own set of improvements and advantages: YOLOv5 is known for its efficiency and speed, making it ideal for resource-limited environments; YOLOv8 and YOLOv10 introduce architectural refinements that boost detection accuracy and performance, especially for complex backgrounds. This progressive evolution of the YOLO architecture makes it an excellent fit for this study, where the precise and efficient detection of small insect targets is essential.

To prepare the YOLO models for the specific task of identifying black aphids, transfer learning from models pre-trained on the widely used COCO dataset [42] was employed. Transfer learning allowed the model to benefit from a foundation of general object detection knowledge, requiring fewer data to adapt to the specific task of pest detection. The models underwent a training process of 150 epochs, with early stopping triggered if no further improvements were observed after 20 epochs, thus preventing the model from overfitting and wasting computational resources. The dataset was resized to 640×640 , 1024×1024 , and 1600×1600 pixels during training for testing different configurations. The different resolutions allowed for a more comprehensive evaluation of the DL model's performance

at varying levels of detail and computational complexity. Finally, a varying batch size of 2 to 8 was used to maintain a balance between efficiency and accuracy.

Optimization was handled using the Stochastic Gradient Descent (SGD) algorithm, with a learning rate set at 0.01. The initial experiments also evaluated alternative optimizers, such as Adam and AdamW, but their performance metrics, particularly in terms of mAP_{50} , were consistently lower compared to SGD. Consequently, SGD was selected as the most effective optimization algorithm for the final training process. The momentum was configured at 0.85 to facilitate smoother convergence, while a small weight decay factor of 0.0005 was introduced to help the models generalize better by reducing overfitting.

The models were trained and evaluated on a high-performance computing setup, powered by an Intel Core i9-14900F 2.00 GHz processor, an RTX 4090 with 24 GB of VRAM, and 128 GB of RAM (Techniki AE, Thessaloniki, Greece).

2.4. Machine Learning Insect Population Prediction

The task of predicting black aphid population growth was addressed using standardized ML models and time-series models trained on environmental data collected throughout the crop cycle using the sensors inside the greenhouse. These data included daily measurements of temperature, humidity, barometric pressure, and black aphid counts obtained from the detection phase. The aim was to build a predictive system capable of forecasting black aphid population growth over the course of seven days, enabling proactive pest management. To obtain reliable inputs for environmental conditions on future days (i.e., the seven days following the prediction start date), weather forecast data from the Open Meteo API [43] were incorporated, allowing the models to make accurate predictions. Since the goal was to predict future values, the dataset was divided sequentially, with the initial 80% used for training and the subsequent 20% reserved for testing.

A combination of ML techniques and time-series forecasting models was employed to capture the complex relationships between environmental variables and population growth. Random forest models were utilized for both classification and regression tasks, alongside gradient boosting and Long Short-Term Memory (LSTM) networks. These models were selected due to their well-established performance in handling structured data and their ability to model complex interactions among features. The Leave-One-Out Cross-Validation (LOOCV) technique was used for model evaluation, which ensured a robust and unbiased performance assessment by iteratively training the models on all data points except one.

For the random forest models, both classifier and regressor variants were used. Each model was configured with 1000 estimators, a maximum tree depth of 7, and a minimum of 5 samples required for node splitting. These hyperparameters were selected to balance the model complexity and computational efficiency while capturing the underlying relationships in the dataset. Gradient boosting models were also employed using similar hyperparameters but with the addition of a learning rate set at 0.001, which allowed the models to improve their accuracy incrementally over successive iterations.

LSTM networks, designed to handle sequential data, were particularly suited to this task, as they capture long-term dependencies in time-series data. The LSTM architecture consisted of two LSTM layers, followed by batch normalization layers to stabilize training, and a final dense layer for output. This model was optimized using a learning rate scheduler that adjusted the learning rate when no improvement was observed, preventing overfitting during the training process. The LSTM was trained over 200 epochs with a batch size of 64 and an initial learning rate of 0.001, ensuring it had enough capacity to learn the temporal patterns in the dataset.

LOOCV was applied to evaluate the performance of all models. This cross-validation method is known for providing robust performance estimates, as it ensures that every data

point is used for both training and validation. By iterating through the entire dataset, this method avoids overfitting and provides a comprehensive evaluation of model accuracy and reliability.

In addition to the ML models, time-series forecasting techniques were also used, including ARIMA (AutoRegressive Integrated Moving Average), ARIMAX (ARIMA with exogenous variables), and SARIMAX (seasonal ARIMA with exogenous variables). These models are designed to handle time-dependent data and were chosen for their ability to capture trends and seasonality in insect population dynamics. ARIMA models rely solely on past values of the target variable to make predictions, while ARIMAX incorporates external factors, such as environmental data, to improve the accuracy of forecasts. SARIMAX adds a seasonal component, which is particularly useful in agricultural studies where population patterns often follow seasonal trends.

For the ARIMA and ARIMAX models, the key parameters include p (the autoregressive order), d (the degree of differencing), and q (the moving average order). These values were tuned using an exhaustive search across a predefined range to find the optimal combination for the dataset. The SARIMAX model required additional tuning of the seasonal order (s) to capture periodic fluctuations in the aphid populations. The same brute-force approach was used to fine-tune this parameter, ensuring that the model's seasonal adjustments were well-calibrated.

2.5. Evaluation Metrics

To assess the performance of the models employed in this study, a variety of evaluation metrics was utilized, each selected to provide insights into different aspects of model accuracy. For the DL models tasked with detecting black aphids, the key metrics used were precision, recall, and the mean Average Precision (mAP) at an Intersection over Union (IoU) threshold of 50%. Precision quantifies the proportion of correctly identified instances among all the positive predictions made by the model, essentially measuring how often the model's detections were accurate. Meanwhile, recall indicates the model's ability to find all relevant instances, reflecting its capacity to avoid missing detections. The mAP is an important aggregate metric that provides a comprehensive view of the model's overall performance. The mAP_{50} is a common benchmark for evaluating object detection tasks. Also, the detection speed (in seconds) was measured to assess the balance between model accuracy and inference speed.

For the task of predicting insect population growth, different metrics were necessary, as this problem involves regression rather than classification. The primary metric used for evaluating the population prediction models was the Mean Squared Error (MSE). MSE is widely used in regression tasks to measure the average of the squared differences between predicted and actual values. It reflects how closely the model's predictions align with the real-world data. A lower MSE value corresponds to higher prediction accuracy, indicating that the predicted values are closer to the observed values. This metric is particularly useful in understanding the variance between predicted and actual insect counts, making it a reliable choice for assessing the model's performance in forecasting tasks.

By applying these metrics, this study ensured that the evaluation was both comprehensive and targeted, providing a detailed picture of how well the models performed in both detection and forecasting tasks. These metrics allowed for a rigorous comparison of model effectiveness and were instrumental in determining their real-world applicability.

3. Results and Discussion

3.1. Model Performance Evaluation

3.1.1. Evaluation of Deep Learning Model for Insect Detection

Table 2 presents the performance results of the DL models in the aphid detection task. The evaluation was conducted using three different input image sizes— 640×640 , 1024×1024 , and 1600×1600 pixels—to determine the effect of image resolution on model performance.

Table 2. Performance evaluation (mAP_{50} , recall, precision, detection speed in seconds) of DL models for different input sizes.

Input Size	Model	Precision	Recall	mAP_{50}	Inference Time (s)
640×640	YOLOv5l	0.716	0.695	0.723	0.029
	YOLOv5x¹	0.720	0.708	0.726	0.045
	YOLOv8l	0.557	0.402	0.404	0.084
	YOLOv8x	0.660	0.530	0.566	0.099
	YOLOv10l	0.696	0.564	0.631	0.046
	YOLOv10x	0.635	0.596	0.621	0.057
1024×1024	YOLOv5l	0.814	0.793	0.817	0.029
	YOLOv5x¹	0.821	0.803	0.818	0.051
	YOLOv8l	0.649	0.637	0.601	0.083
	YOLOv8x	0.657	0.647	0.612	0.098
	YOLOv10l	0.755	0.734	0.777	0.069
	YOLOv10x	0.715	0.724	0.748	0.088
1600×1600	YOLOv5l	0.818	0.813	0.826	0.082
	YOLOv5x	0.830	0.824	0.832	0.091
	YOLOv8l	0.769	0.784	0.796	0.225
	YOLOv8x	0.787	0.809	0.814	0.237
	<u>YOLOv10l^{1,2}</u>	<u>0.848</u>	<u>0.877</u>	<u>0.891</u>	<u>0.134</u>
	YOLOv10x	0.803	0.851	0.866	0.151

¹ Bold row indicates the best performing model for each input size in terms of mAP_{50} . ² Underlined row indicates the best performing model across all trained models in terms of mAP_{50} .

By comparing the different input sizes, it became clear that increasing the image resolution generally improved the detection metrics across all YOLO models, though at the cost of increased computational resources. For instance, with the input size set as 640×640 , the YOLOv5l model achieved a precision of 71.6%, a recall of 69.5%, and an mAP_{50} of 72.3%. The YOLOv5x variant slightly improved on these metrics with a precision of 72%, a recall of 70.8%, and an mAP_{50} of 72.6%. These results suggest that even at smaller input sizes, more complex models, like YOLOv5x, can provide superior performance, making them suitable for environments where computational efficiency is crucial.

As the input size increased to 1024×1024 pixels, both the precision and recall improved noticeably. At this resolution, YOLOv5 models continued to lead in performance, achieving the best results among the variants. More specifically, YOLOv5l achieved a precision of 81.4%, a recall of 79.3%, and an mAP_{50} of 81.7%, while YOLOv5x further outperformed this with a precision of 82.1%, a recall of 80.3%, and an mAP_{50} of 81.8%. These findings highlight the advantages of utilizing higher resolution images, as they offer improved detection accuracy.

When using the largest input size, 1600×1600 pixels, all YOLO models demonstrated the highest performance metrics, with the YOLOv10 variants outperforming both YOLOv5 and YOLOv8. As showcased in Table 2, the best metrics were achieved by YOLOv10l with a precision of 84.8%, a recall of 87.7%, and an mAP_{50} of 89.1%. This pattern did not extend to YOLOv8x, which, although it outperformed other YOLOv5 and YOLOv8 models,

showed only moderate improvements across all metrics. This suggests that increasing model complexity does not necessarily result in the highest detection accuracy.

The results indicate that higher input resolutions generally enhanced the detection performance, allowing the models to capture finer details and making it easier to detect minute insects like black aphids, which might otherwise not be detected. Additionally, for both YOLOv5 and YOLOv8, more complex models tended to improve the detection accuracy, though this advantage requires greater computational resources. Among the model variants, the 'large' versions consistently offered a favorable balance of accuracy and efficiency, making them better suited for real-time applications where resource constraints are a priority compared to the 'xlarge' versions. These findings highlight the importance of selecting the appropriate model and input size to match the specific accuracy requirements and computational limitations of the deployment environment.

The best performing model across all the trained models was YOLOv10l, achieving an mAP_{50} of 89.1% with an inference speed of 0.134 s, making it the optimal choice in terms of both accuracy and speed. Although YOLOv5x at 1600×1600 achieved a mAP_{50} of 83.2% with a slightly faster inference speed of 0.091 s, the minimal difference in speed was outweighed by YOLOv10l's nearly 6% increase in detection accuracy. The detection performance of YOLOv10l is visualized in Figure 4, which illustrates the model's output under both low- and high-density aphid conditions. These visual examples further validate the quantitative metrics, emphasizing the model's reliability in diverse detection scenarios.

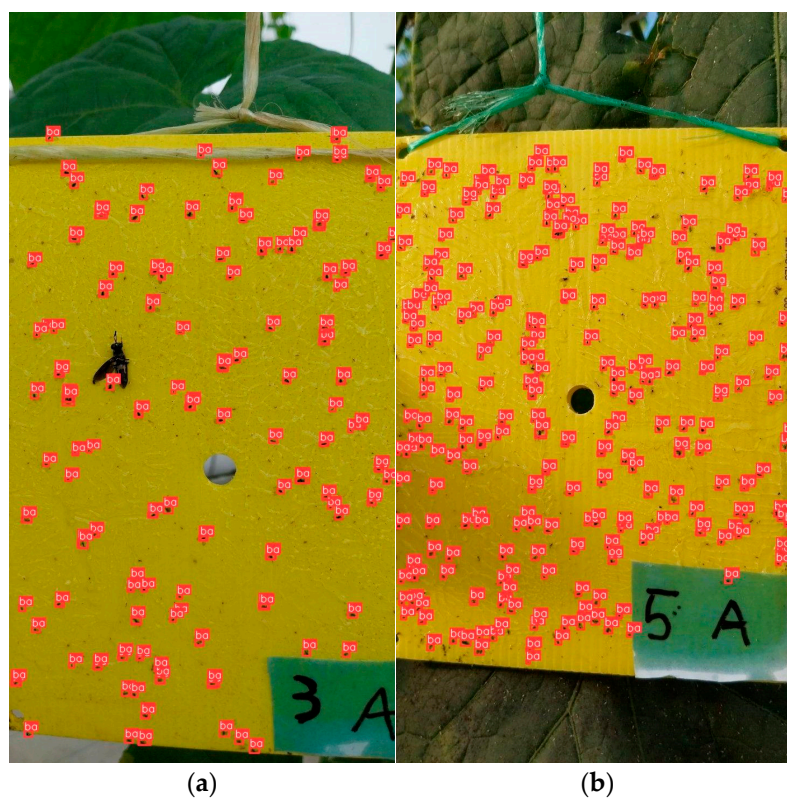


Figure 4. Detection results of the YOLOv10x model. Images (a) and (b) both contain red bounding boxes that indicate the detected black aphid insects stuck on the pheromone-based sticky paper trap.

Overall, the models trained and tested in this study demonstrated strong performance in detecting black aphids, with YOLOv10 and YOLOv5 emerging as the top-performing variants, particularly at larger input sizes. These models maintained high detection accuracy across varying insect densities and image complexities, proving their effectiveness in practical pest management applications. In contrast, the YOLOv8 variants (YOLOv8l and

YOLOv8x) delivered relatively disappointing results, even at the highest input resolution of 1600×1600 pixels. Despite being more recent versions, the YOLOv8 models consistently underperformed compared to YOLOv5 across all metrics and inference speed. This outcome suggests that, for this specific task, the added complexity of YOLOv8 did not translate into improved detection performance, highlighting the superior efficiency and accuracy of YOLOv5 and YOLOv10 models for black aphid detection.

3.1.2. Evaluation of Insect Population Prediction Model

Following the assessment of the DL model's efficacy in detecting black aphids, the focus shifted to leveraging the detection data alongside the environmental variables to evaluate the performance of various models in predicting insect population growth. The dataset generated for this study comprised daily records of environmental factors such as temperature, humidity, and barometric pressure, along with the black aphid population counts derived from the detection model. It was enriched by recording not only the number of insects captured on the sticky paper traps but also the day-to-day changes in insect count. This focus on population variation allowed for more meaningful predictions, as the aim was to model the trend in insect population growth rather than absolute daily counts. The data, accessible through the DSS, provided the foundation for training several ML models aimed at forecasting changes in the aphid population over time.

Various ML models were applied to this refined dataset, including time-series models like ARIMA and ARIMAX, as well as more advanced techniques such as gradient boosting and random forest. These models were evaluated based on their Mean Squared Error (MSE) in predicting daily changes in the insect population. Among the models tested, the ARIMAX model emerged as the most effective, achieving an MSE of 75.61, which corresponds to an average deviation of 8.6 insects per day. This demonstrates that incorporating external variables, such as weather conditions, significantly improved the model's predictive power. Consequently, subsequent analyses focused primarily on fine-tuning and optimizing the ARIMAX model for this task.

The ARIMAX model was configured with the following hyperparameters: the p , q , and d parameters were set to 3, 4, and 1 respectively. These values were determined through an iterative tuning process, optimizing the model for the specific characteristics of the dataset. By taking into account both past aphid population counts and environmental data, the model was able to effectively anticipate population trends and provide valuable predictions for pest management.

The results of this analysis underscore the importance of accounting for external variables and operational factors, such as sticky paper replacements, when building predictive models for insect populations. The integration of these additional data points allowed for a more accurate and reliable forecasting system, which is essential for guiding early intervention strategies in pest control. The performance of the ARIMAX model, alongside other models, is presented in Figure 5.

Furthermore, a detailed comparison of the predicted versus actual aphid population changes can be seen in Figure 6. These findings highlight the potential of the ML and time-series techniques in agricultural settings, particularly in optimizing pest management efforts and minimizing the need for reactive pesticide applications.

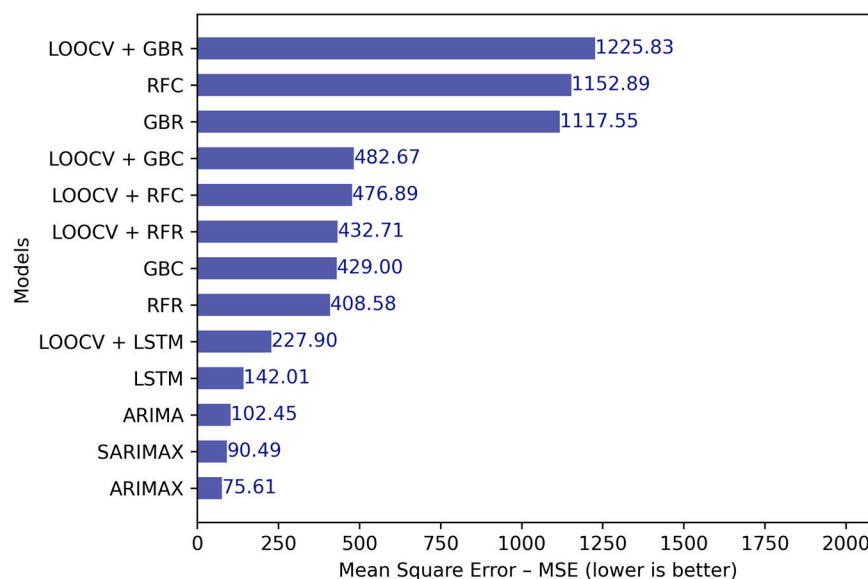


Figure 5. Mean Squared Error (MSE) for different models forecasting insect population count. A lower MSE indicates better performance. Abbreviations: GBC—GradientBoostingClassifier, GBR—GradientBoostingRegressor, RFC—RandomForestClassifier, RFR—RandomForestRegressor.

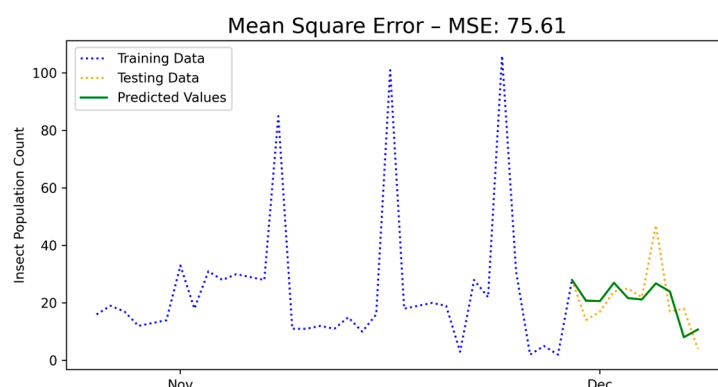


Figure 6. Performance of the ARIMAX model for daily insect population count predictions, with an MSE of 75.61. The dataset was split sequentially, with the initial 80% used as training data (blue) and the final 20% as testing data (orange), while the model’s predictions are shown in green.

3.2. Integration of Deep Learning Detection Model in Mobile Application

Building on the evaluation of the best-performing DL model for black aphid detection presented in Section 3.1.1, this model was seamlessly integrated into an existing mobile application created for pest and plant disease identification [37]. This mobile application, which was previously field-tested in a different pest scenario, is designed to support scalability and flexibility, allowing new models to be deployed without significant architectural changes. With cross-platform compatibility built in, the mobile application supports both iOS and Android devices and leverages the Ionic Framework along with Capacitor for streamlined development. The application operates by utilizing the smartphone’s camera to capture images of infected plants or pests, which are then processed using AI models hosted on a back-end system. The core of the application is a DSS, which manages the incoming data, executes the relevant DL model, and returns detection results to the user.

The integration of the black aphid YOLOv10l detection model into this system highlights the flexibility of the architecture. The robust design of the mobile application enables it to accommodate various DL models for different pests, thus enhancing its functionality. By utilizing a Python-based REST API [44], the application efficiently communicates with the server-side DSS, which is also designed for performance and scalability. Security, perfor-

mance, and scalability are critical priorities for the mobile app's architecture. Keycloak was implemented for identity and access management, ensuring secure interactions between the user and the server. Moreover, RabbitMQ was used for asynchronous communication between the mobile app and the back-end services, which allows for efficient task handling and data processing. These architectural choices guarantee that the system can scale to support a growing number of users and different AI models, ensuring long-term viability and adaptability.

The mobile application features a user-friendly interface that allows users to either take a photo using their smartphone camera or select an image from the gallery (as shown in Figure 7a). Once the user captures an image, they can select the type of image photographed (either a sticky paper trap image or a leaf image) and transmitted to the DSS (Figure 7b). After submission, the app sends the image to the DSS, where the DL model processes the data and returns the detection results. The app then displays the results, such as bounding boxes around the detected pests along with the insect count (Figure 7c). This real-time detection and visualization process enables users to quickly assess the level of infestation and take necessary action in the field.

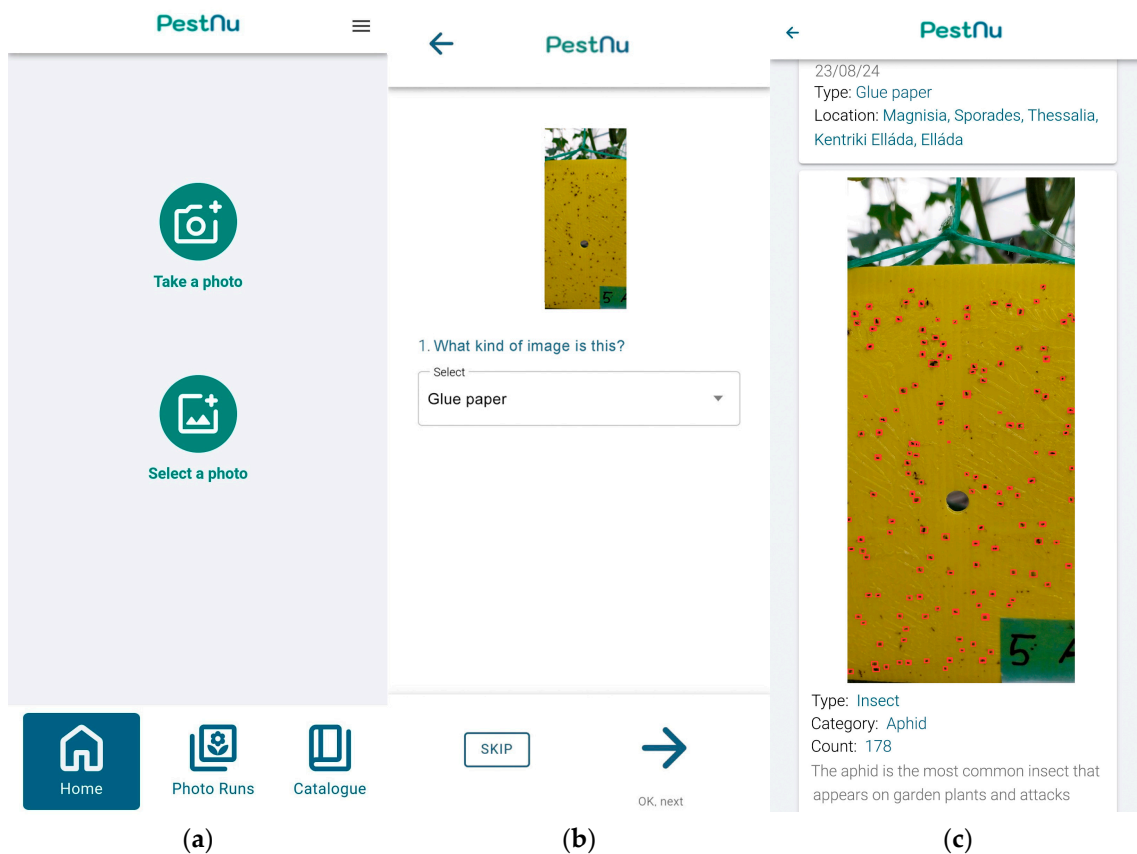


Figure 7. Mobile application screens: (a) Home screen of the app, where the user can capture a new photo or select an existing one from their smartphone; (b) screen where the user selects the type of image to be uploaded and transmitted to the DSS; (c) Detection screen displaying the AI model's results and recommendations. The example shows a pheromone-based sticky paper trap where the model identifies black aphids, highlighting them with red bounding boxes and counting them.

The mobile app's design emphasizes scalability, allowing new DL models for different agricultural pests and diseases to be easily added as new modules. This ensures that the platform can continue to evolve, supporting a wider range of pest detection and management solutions. Its integration with cloud-based services ensures that the system

remains highly responsive while maintaining the robustness and security required for widespread agricultural use.

4. Conclusions

This study explored the detection and prediction of black aphid populations within a greenhouse environment with cucumber cultivation using DL and ML models integrated with real-time monitoring capabilities. By employing YOLO-based object detection models, the research demonstrated effective insect detection with varying model complexities and input sizes. Specifically, three different input image sizes— 640×640 , 1024×1024 , and 1600×1600 pixels—were used, from which YOLOv10l emerged as the best-performing model, achieving an mAP_{50} of 89.1% with an inference speed of 0.134 s at an input size of 1600×1600 . This model was particularly advantageous for accurate insect detection, balancing high detection accuracy and computational efficiency. Notably, YOLOv8, despite its newer architecture, underperformed compared to YOLOv5, highlighting that increased complexity does not always yield better results.

The environmental dataset used for population prediction spanned 44 days and included environmental variables, such as temperature, humidity, and barometric pressure, alongside daily insect counts. This limited timeframe presents a challenge; a longer data collection period could improve model robustness and accuracy. Nevertheless, the time-series ARIMAX model effectively captured population trends, outperforming more generic ML models, and demonstrated the importance of incorporating environmental data for predictive accuracy. The ARIMAX model achieved an MSE of 75.61, corresponding to an average deviation of 8.61 insects per day. Expanding the dataset could further refine this model for improved pest management.

By integrating the detection model into a mobile application, real-time monitoring of pest populations is made accessible to users in agricultural settings. However, future work should focus on incorporating the population prediction models directly into the mobile application. This integration would enable end users to receive both immediate detection data and short-term population forecasts, facilitating timely and informed pest control decisions.

Future work could involve developing models tailored to specific environmental and pest conditions. Moreover, extended deployments in open field and greenhouse deployments could periodically update models, progressively enhancing their accuracy for early intervention in pest management. Regular and consistent cleaning or replacement of the pheromone sticky paper traps could also enhance the reliability of predictions by reducing noise from excessive insect accumulation. This study establishes a foundation for scalable, AI-driven pest monitoring solutions that support precision agriculture and sustainable pest management practices.

Author Contributions: Conceptualization, D.K., P.C. and S.F.; methodology, D.K. and P.C.; software, P.C. and D.K.; validation, D.K.; formal analysis, P.C.; investigation, D.K.; resources, E.M.P. and N.K.; data curation, S.F. and P.C.; writing—original draft preparation, P.C., D.K., S.F. and N.K.; writing—review and editing, E.M.P., S.F. and N.K.; visualization, P.C. and D.K.; supervision, E.M.P. and N.K.; project administration, E.M.P. and N.K.; funding acquisition, E.M.P. and N.K. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The dataset developed in the frame of this work is available for download at <https://zenodo.org/records/14097660> (accessed on 12 November 2024) [45].

Acknowledgments: This work was partially carried out under the PestNu project that has received funding from the European Union's Horizon 2020 research and innovation program under the Green

Deal grant agreement No. 101037128—PestNu and partially under the E-SPFdigit project, funded by European Union’s Horizon Europe research and innovation programme under the grant agreement No. 101157922.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. IPPC Secretariat. *Scientific Review of the Impact of Climate Change on Plant Pests*; FAO on behalf of the IPPC Secretariat: Rome, Italy, 2021; Available online: <https://openknowledge.fao.org/handle/20.500.14283/cb4769en> (accessed on 12 November 2024).
2. Trebicki, P. Climate change and plant virus epidemiology. *Virus Res.* **2020**, *286*, 198059. [[CrossRef](#)] [[PubMed](#)]
3. Zayan, S.A. Impact of Climate Change on Plant Diseases and IPM Strategies. In *Plant Diseases—Current Threats and Management Trends*; IntechOpen: London, UK, 2019. [[CrossRef](#)]
4. Pu, J.; Wang, Z.; Chung, H. Climate change and the genetics of insecticide resistance. *Pest Manag. Sci.* **2020**, *76*, 846–852. [[CrossRef](#)] [[PubMed](#)]
5. Canto, T.; Aranda, M.A.; Fereres, A. Climate change effects on physiology and population processes of hosts and vectors that influence the spread of hemipteran-borne plant viruses. *Glob. Change Biol.* **2009**, *15*, 1884–1894. [[CrossRef](#)]
6. Sattler, C.; Kächele, H.; Verch, G. Assessing the intensity of pesticide use in agriculture. *Agric. Ecosyst. Environ.* **2007**, *119*, 299–304. [[CrossRef](#)]
7. Magarey, R.D.; Klammer, S.S.; Chappell, T.M.; Trexler, C.M.; Pallipparambil, G.R.; Hain, E.F. Eco-efficiency as a strategy for optimizing the sustainability of pest management. *Pest Manag. Sci.* **2019**, *75*, 3129–3134. [[CrossRef](#)]
8. Bock, C.H.; Poole, G.H.; Parker, P.E.; Gottwald, T.R. Plant Disease Severity Estimated Visually, by Digital Photography and Image Analysis, and by Hyperspectral Imaging. *Crit. Rev. Plant Sci.* **2010**, *29*, 59–107. [[CrossRef](#)]
9. Andréa, M.M.; Peres, T.B.; Luchini, L.C.; Pettinelli, A., Jr. Impact of long-term pesticide applications on some soil biological parameters. *J. Environ. Sci. Health Part B* **2000**, *35*, 297–307. [[CrossRef](#)]
10. Qazi, S.; Khawaja, B.A.; Farooq, Q.U. IoT-Equipped and AI-Enabled Next Generation Smart Agriculture: A Critical Review, Current Challenges and Future Trends. *IEEE Access* **2022**, *10*, 21219–21235. [[CrossRef](#)]
11. Misra, N.N.; Dixit, Y.; Al-Mallahi, A.; Bhullar, M.S.; Upadhyay, R.; Martynenko, A. IoT, Big Data, and Artificial Intelligence in Agriculture and Food Industry. *IEEE Internet Things J.* **2022**, *9*, 6305–6324. [[CrossRef](#)]
12. Høye, T.T.; Årje, J.; Bjerger, K.; Hansen, O.L.P.; Iosifidis, A.; Leese, F.; Mann, H.M.R.; Meissner, K.; Melvad, C.; Raitoharju, J. Deep learning and computer vision will transform entomology. *Proc. Natl. Acad. Sci. USA* **2021**, *118*, e2002545117. [[CrossRef](#)]
13. Simonyan, K.; Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognition. *arXiv* **2014**, arXiv:1409.1556.
14. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You Only Look Once: Unified, Real-Time Object Detection. *arXiv* **2016**, arXiv:1506.02640.
15. Kumar, N.; Nagarathna; Flammini, F. YOLO-Based Light-Weight Deep Learning Models for Insect Detection System with Field Adaption. *Agriculture* **2023**, *13*, 741. [[CrossRef](#)]
16. Verma, S.; Tripathi, S.; Singh, A.; Ojha, M.; Saxena, R.R. Insect Detection and Identification using YOLO Algorithms on Soybean Crop. In Proceedings of the TENCON 2021—2021 IEEE Region 10 Conference (TENCON), Auckland, New Zealand, 7–10 December 2021; pp. 272–277. [[CrossRef](#)]
17. Liu, L.; Wang, R.; Xie, C.; Yang, P.; Wang, F.; Sudirman, S.; Liu, W. PestNet: An End-to-End Deep Learning Approach for Large-Scale Multi-Class Pest Detection and Classification. *IEEE Access* **2019**, *7*, 45301–45312. [[CrossRef](#)]
18. Zhong, Y.; Gao, J.; Lei, Q.; Zhou, Y. A Vision-Based Counting and Recognition System for Flying Insects in Intelligent Agriculture. *Sensors* **2018**, *18*, 1489. [[CrossRef](#)]
19. Giakoumoglou, N.; Pechlivani, E.M.; Katsoulas, N.; Tzovaras, D. White Flies and Black Aphids Detection in Field Vegetable Crops using Deep Learning. In Proceedings of the 2022 IEEE 5th International Conference on Image Processing Applications and Systems (IPAS), Genova, Italy, 5–7 December 2022; pp. 1–6. [[CrossRef](#)]
20. Giakoumoglou, N.; Pechlivani, E.M.; Tzovaras, D. Generate-Paste-Blend-Detect: Synthetic dataset for object detection in the agriculture domain. *Smart Agric. Technol.* **2023**, *5*, 100258. [[CrossRef](#)]
21. Xie, C.; Zhang, J.; Li, R.; Li, J.; Hong, P.; Xia, J.; Chen, P. Automatic classification for field crop insects via multiple-task sparse representation and multiple-kernel learning. *Comput. Electron. Agric.* **2015**, *119*, 123–132. [[CrossRef](#)]
22. Liu, J.; Wang, X. Tomato Diseases and Pests Detection Based on Improved Yolo V3 Convolutional Neural Network. *Front. Plant Sci.* **2020**, *11*, 898. [[CrossRef](#)]
23. Gutierrez, A.; Ansuategi, A.; Susperregi, L.; Tubío, C.; Rankić, I.; Lenža, L. A Benchmarking of Learning Strategies for Pest Detection and Identification on Tomato Plants for Autonomous Scouting Robots Using Internal Databases. *J. Sens.* **2019**, *2019*, 5219471. [[CrossRef](#)]

24. Ibrahim, E.A.; Salifu, D.; Mwalili, S.; Dubois, T.; Collins, R.; Tonnang, H.E.Z. An expert system for insect pest population dynamics prediction. *Comput. Electron. Agric.* **2022**, *198*, 107124. [[CrossRef](#)]
25. Bahlai, C.A. Forecasting insect dynamics in a changing world. *Curr. Opin. Insect Sci.* **2023**, *60*, 101133. [[CrossRef](#)] [[PubMed](#)]
26. Marković, D.; Vujičić, D.; Tanasković, S.; Đorđević, B.; Randić, S.; Stamenković, Z. Prediction of Pest Insect Appearance Using Sensors and Machine Learning. *Sensors* **2021**, *21*, 4846. [[CrossRef](#)] [[PubMed](#)]
27. Skawsang, S.; Nagai, M.K.; Tripathi, N.; Soni, P. Predicting Rice Pest Population Occurrence with Satellite-Derived Crop Phenology, Ground Meteorological Observation, and Machine Learning: A Case Study for the Central Plain of Thailand. *Appl. Sci.* **2019**, *9*, 4846. [[CrossRef](#)]
28. Rathod, S.; Yerram, S.; Arya, P.; Katti, G.; Rani, J.; Padmakumari, A.P.; Somasekhar, N.; Padmavathi, C.; Ondrasek, G.; Amudan, S.; et al. Climate-Based Modeling and Prediction of Rice Gall Midge Populations Using Count Time Series and Machine Learning Approaches. *Agronomy* **2022**, *12*, 22. [[CrossRef](#)]
29. Chiu, L.-Y.; Arcega Rustia, D.J.; Lu, C.-Y.; Lin, T.-T. Modelling and Forecasting of Greenhouse Whitefly Incidence Using Time-Series and ARIMAX Analysis. *IFAC-PapersOnLine* **2019**, *52*, 196–201. [[CrossRef](#)]
30. Kawakita, S.; Takahashi, H. Time-series analysis of population dynamics of the common cutworm, *Spodoptera litura* (Lepidoptera: Noctuidae), using an ARIMAX model. *Pest Manag. Sci.* **2022**, *78*, 2423–2433. [[CrossRef](#)]
31. Dedryver, C.-A.; Ralec, A.L.; Fabre, F. The conflicting relationships between aphids and men: A review of aphid damage and control strategies. *Comptes Rendus Biol.* **2010**, *333*, 539–553. [[CrossRef](#)]
32. Sharma, S.; Sood, A.K.; Ghongade, D.S. Assessment of losses inflicted by the aphid, *Myzus persicae* (Sulzer) to sweet pepper under protected environment in north western Indian Himalayan region. *Phytoparasitica* **2022**, *50*, 51–62. [[CrossRef](#)]
33. Budnik, K.; Laing, M.D.; da Graça, J.V. Reduction of yield losses in pepper crops caused by potato virus Y in KwaZulu-Natal, South Africa, using plastic mulch and Yellow sticky traps. *Phytoparasitica* **1996**, *24*, 119–124. [[CrossRef](#)]
34. Ning, S.; Xia, L.; Fang, Y.; Zhou, Z.; Wang, Y.; Chen, J. Characterization and Genetic Mapping of Resistance to Cotton–Melon Aphid (*Aphis gossypii*) in Cucumber. *Plant Breed.* **2024**, *144*, 11–21. [[CrossRef](#)]
35. Zhang, L.; Chen, C.; Li, Y.; Suo, C.; Zhou, W.; Liu, X.; Deng, Y.; Sohail, H.; Li, Z.; Liu, F.; et al. Enhancing Aphid Resistance in Horticultural Crops: A Breeding Prospective. *Hortic. Res.* **2024**, *11*, uhae275. [[CrossRef](#)]
36. Pechlivani, E.M.; Gkogkos, G.; Giakoumoglou, N.; Hadjigeorgiou, I.; Tzovaras, D. Towards Sustainable Farming: A Robust Decision Support System’s Architecture for Agriculture 4.0. In Proceedings of the 2023 24th International Conference on Digital Signal Processing (DSP), Rhodes (Rodos), Greece, 11–13 June 2023; pp. 1–5. [[CrossRef](#)]
37. Christakakis, P.; Papadopoulou, G.; Mikos, G.; Kalogiannidis, N.; Ioannidis, D.; Tzovaras, D.; Pechlivani, E.M. Smartphone-Based Citizen Science Tool for Plant Disease and Insect Pest Detection Using Artificial Intelligence. *Technologies* **2024**, *12*, 101. [[CrossRef](#)]
38. Roboflow: Computer Vision Tools for Developers and Enterprises. Available online: <https://roboflow.com/> (accessed on 5 October 2024).
39. Ultralytics, YOLOv5. 2020. Available online: <https://docs.ultralytics.com/models/yolov5> (accessed on 12 November 2024).
40. Ultralytics, YOLOv8. 2023. Available online: <https://github.com/ultralytics/ultralytics> (accessed on 12 November 2024).
41. Wang, A.; Chen, H.; Liu, L.; Chen, K.; Lin, Z.; Han, J.; Ding, G. YOLOv10: Real-Time End-to-End Object Detection. *arXiv* **2024**, arXiv:2405.14458.
42. Lin, T.-Y.; Maire, M.; Belongie, S.; Bourdev, L.; Girshick, R.; Hays, J.; Perona, P.; Ramanan, D.; Zitnick, C.L.; Dollár, P. Microsoft COCO: Common Objects in Context. *arXiv* **2015**, arXiv:1405.0312.
43. Free Open-Source Weather API | Open-Meteo.com. Available online: <https://open-meteo.com/> (accessed on 26 July 2024).
44. Fielding, R.T. Architectural Styles and the Design of Network-Based Software Architectures. Ph.D. Thesis, University of California, Irvine, CA, USA, 2000.
45. Christakakis, P.; Kapetas, D.; Pechlivani, E.-M. Black Aphids Glue Paper Traps Dataset [Data Set]. Zenodo. 2024. Available online: <https://zenodo.org/records/14097660> (accessed on 12 November 2024).

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.