Computers and Electronics in Agriculture 256 (2027) 112386

Contents lists available at ScienceDirect

Computers and Electronics in Agriculture

journal homepage: www.elsevier.com/locate/compag

Original papers

AgroInsect: A hybrid AI-driven edge-cloud architecture for precision pest detection and geostatistical interpolation in agriculture

Guilherme Pires Silva de Almeida [a,*], Leonardo Nazário Silva dos Santos [b], Ruy de Oliveira [a], Marconi Batista Teixeira [b], Mario De Oliveira [c], Pablo da Costa Gontijo [b], Adriano Soares de Oliveira Bailão [b], Heyde Francielle do Carmo França [b], Sergio Souza Novak [b], Jéssica Lauanda Stirle [b]

[a] *Instituto Federal de Educação, Ciência e Tecnologia de Mato Grosso, Campus Cuiabá, Cuiabá, 78005-200, Brazil*
[b] *Instituto Federal de Educação Ciência e Tecnologia Goiano, Campus Rio Verde, Rio Verde, 75901-970, Brazil*
[c] *Department of Engineering, Birmingham City University, Birmingham, B4 7XG, UK*

ARTICLE INFO

Keywords:
YOLOv11
TensorFlow lite
Deep learning
Pest detection
Ordinary kriging

ABSTRACT

This study presents AgroInsect, a hybrid edge-cloud framework for agricultural pest monitoring that integrates on-device deep learning inference with server-side geostatistical mapping into a unified, field-deployable smartphone workflow. Three lightweight YOLO variants (YOLOv5n, YOLOv8n, and YOLOv11n) were trained and benchmarked under three TensorFlow Lite quantization schemes (Float32, Float16, and INT8), evaluating detection accuracy (mAP@0.5, mAP@0.5:0.95), inference latency, energy consumption, and thermal behavior on a mid-range Android device. YOLOv11n-INT8 was selected for deployment, achieving mAP@0.5 of 79.3% with a model size of only 2.98 MB and an estimated battery drain of 8.0% h^{-1}. The system detects four pest species of agronomic relevance to soybean and maize crops in Brazil (*Diabrotica speciosa*, *Dalbulus maidis*, *Diceraeus* spp., and *Spodoptera frugiperda*) directly on the device, while geolocation is automatically extracted from image EXIF metadata and validated against farm boundaries using the Haversine formula. Detection records are synchronized to a cloud database, where Ordinary Kriging (PyKrige) generates continuous pest density surfaces that are returned to the mobile interface for visualization. Field evaluation achieved 95.1% overall accuracy and F1-scores of 0.94 or higher for all species. Kriging validation under dense synthetic sampling yielded R^2 up to 0.9656 (species-dependent), with performance degrading under sparse conditions (R^2 = 0.41–0.68 at 200 × 200 grids). Kriging was further validated against Inverse Distance Weighting (IDW) using leave-one-out cross-validation on real field detections ($n = 79$), which showed comparable point-prediction accuracy between the two methods, while Kriging preserved more spatial variance and captured a larger share of the observed infestation hotspot. Our approach integrates on-device pest detection, automated georeferenced validation, and server-side geostatistical mapping within a single operational pipeline, offering a practical tool for Integrated Pest Management in low-connectivity rural environments.

1. Introduction

According to the Food and Agriculture Organization of the United Nations (FAO), approximately 40% of global food crops are lost annually due to pests and diseases, resulting in estimated annual losses of approximately US$ 220 billion to the global economy (Savary et al., 2019). It is estimated that 70,000 pest species, including insects, pathogens, and weeds, can cause damage to agricultural crops worldwide, with the potential to generate yield losses ranging from 35% to 42% of production capacity (Pimentel, 1997). The significant investments in pest control via chemical pesticides, coupled with environmental contamination and human health concerns, underscore the urgent need for more effective and sustainable approaches to pest management (Pimentel, 2009; Tetila et al., 2024). Despite the vast diversity of existing agricultural pest species, this study focuses on four insect pests (de Almeida et al., 2024): *Diabrotica speciosa*, *Dalbulus maidis*, *Diceraeus* spp., and *Spodoptera frugiperda*. These species were selected due to their high agronomic and economic relevance to major crops such as maize and soybean in Brazil and Latin America, as well as their representativeness of distinct functional groups and damage

* Corresponding author.
E-mail address: guilherme.almeida@ifmt.edu.br (G.P.S. de Almeida).

https://doi.org/10.1016/j.compag.2026.112386
Received 21 February 2026; Received in revised form 9 August 2026; Accepted 31 August 2026
Available online 9 September 2026

mechanisms within Integrated Pest Management (IPM) programs. This targeted selection supports the development of practical and scalable on-device deep learning solutions for real field conditions. While the proposed approach is not limited to these species, it establishes a baseline that can be extended to additional pests as annotated data become available. Advances in artificial intelligence have catalyzed the integration of computational methods into pest management practices, as manual monitoring is inherently challenging, resource-intensive, and time-consuming (Zhou et al., 2024).

Providing farmers with accurate and timely information for the effective prevention and control of pests and diseases remains a significant challenge. The scarcity of granular data regarding pest typologies, damage intensity, and mitigation strategies limits the adoption of predictive and site-specific management practices (Karar et al., 2021). Moreover, restricted connectivity in rural areas, combined with the limited availability of trained evaluators, further constrains timely decision-making in the field (Han et al., 2024).

Edge Computing (EC) has emerged as a promising paradigm to address these constraints by enabling localized data processing near the data source, thereby reducing latency, lowering bandwidth requirements, and supporting operation under intermittent or absent network connectivity (Sapna et al., 2026). In precision agriculture, EC facilitates real-time decision-making directly on resource-constrained field devices, such as smartphones and embedded boards, without relying on permanent cloud access. A comprehensive review of EC applications in smart farming highlights a growing research interest in edge-based solutions; however, critical gaps persist, including the scarcity of balanced datasets, insufficient evaluation of energy consumption and inference latency on actual edge hardware, and limited real-world validation of developed systems (Sapna et al., 2026). These findings reinforce the need for integrated frameworks capable of performing on-device inference while maintaining practical deployability under field conditions.

Recent studies have demonstrated the potential of computer vision and deep learning for agricultural pest detection. Cloud-based solutions achieve high classification accuracy but depend on remote servers, making them vulnerable to connectivity failures in rural environments (Karar et al., 2021; Zhang et al., 2024). To mitigate this limitation, several works have explored lightweight detection models based on the YOLO family. Insect-YOLO was introduced to improve the detection of small or occluded insects through attention mechanisms, although performance remains sensitive to environmental variability (Wang et al., 2025). Other studies have proposed architectural optimizations to reduce computational overhead while maintaining accuracy, such as the incorporation of FasterNet and PConvGLU modules (Qin et al., 2024). However, some approaches focus solely on detecting insect presence without species-level discrimination, limiting their usefulness for pest-specific management decisions (Vilar-Andreu et al., 2024). Beyond smartphone-based solutions, robotic scouting platforms have also been explored: six YOLO variants were recently benchmarked on an autonomous robot equipped with RTK-GPS for Colorado potato beetle detection, with YOLOv11s achieving the most reliable precision–localization balance under real field conditions (Gill et al., 2026). Despite their accuracy, such systems depend on dedicated hardware infrastructure, limiting their scalability in smallholder and low-connectivity farming contexts.

Field-based studies using YOLO models have reported high detection accuracy for insects in soybean and maize crops (Verma et al., 2021), as well as for weed detection using UAV imagery (Tetila et al., 2024). Transfer learning strategies have also shown effectiveness in improving model performance under limited data availability (Souza et al., 2019). Despite these advances, most existing works remain centered on detection performance alone and do not address how detection outputs are spatially validated, integrated, and transformed into actionable information for decision support.

Although significant advances have been achieved in computer vision and deep learning for agricultural pest detection, most existing approaches remain limited to isolated tasks, primarily focusing on object detection performance. In practical agricultural scenarios, however, effective pest management requires more than accurate classification: it demands reliable on-device operation under field conditions, automatic validation of spatial information, and the transformation of point-based detections into interpretable spatial patterns that support decision-making. Current solutions often depend on cloud-based inference, external GIS platforms, or manual georeferencing, which restricts their applicability in areas with limited connectivity and increases operational complexity. As a result, there is still a lack of integrated, smartphone-embedded systems that combine on-device pest detection, spatial validation, and geostatistical modeling within a single operational workflow for precision pest monitoring.

To address this gap, this study proposes AgroInsect, a hybrid smartphone-server framework for agricultural pest monitoring in which pest detection, image annotation, and geospatial validation are performed entirely on-device, enabling field operation without network dependency, while geostatistical interpolation and density map generation are executed on a cloud-based server and the results returned to the mobile interface for visualization. This architecture reflects a deliberate design choice: computationally lightweight inference tasks are handled locally on standard Android smartphones, whereas Kriging-based spatial modeling, which requires higher computational resources, is offloaded to the server and accessed when connectivity is available.

This hybrid edge-cloud strategy aligns with a broader trend in precision agriculture, in which computationally lightweight tasks are executed locally on field devices while heavier analytical workloads are offloaded to remote infrastructure when connectivity allows (Sapna et al., 2026; Yang et al., 2026).

Unlike previous works that either rely on cloud-based inference (Karar et al., 2021) or evaluate detection in isolation without spatial integration (Vilar-Andreu et al., 2024), AgroInsect addresses the full monitoring pipeline, from on-device inference under quantization constraints to server-side geostatistical mapping, evaluated against standardized metrics including mAP, inference latency, energy consumption, and thermal behavior across multiple Android devices.

To the best of our knowledge, however, no existing solution integrates on-device pest detection, automatic georeferenced validation, and server-side geostatistical mapping within a unified field-deployable workflow for agricultural pest monitoring.

To address this, the specific objectives of this study are: (1) to train and benchmark multiple lightweight object detection models (YOLOv5n, YOLOv8n, and YOLOv11n) converted to TensorFlow Lite format under three quantization schemes (Float32, Float16, and INT8), evaluating detection accuracy, inference latency, energy consumption, and thermal behavior on Android devices, in order to identify the most suitable configuration for on-device pest monitoring; (2) to develop a mobile application for smartphones that captures images, executes the selected model locally (on-device), and logs detections associated with geospatial metadata; (3) to implement a mechanism to synchronize the data collected by the application with a cloud database, ensuring secure and scalable storage of images and detection records; and (4) to utilize the collected data to generate heatmaps and spatial interpolations, illustrating the distribution and incidence of pests in the fields over time (Zhou et al., 2024).

2. Materials and methods

2.1. Architecture and key technologies

The AgroInsect application was developed with a focus on on-device agricultural pest detection, image annotation, and geospatial visualization of detected occurrences. The application was built using the Flutter framework within the Android Studio environment and supported by

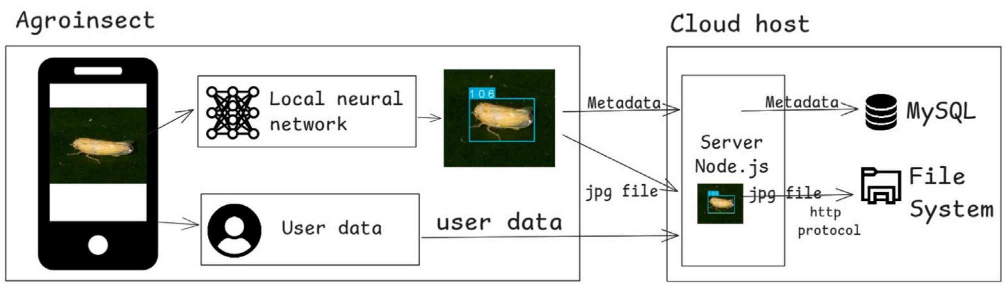


Fig. 1. Data flow between the AgroInsect mobile application and the cloud server: on-device inference and user input are transmitted as metadata and JPG files via HTTP to a Node.js server, which stores metadata in MySQL and images in the file system.

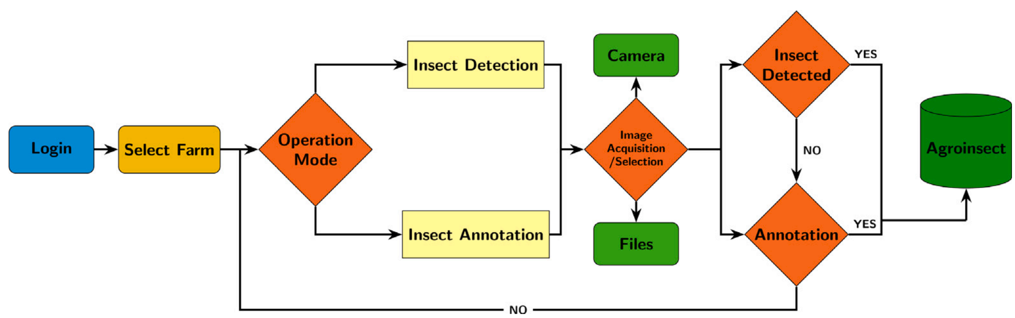


Fig. 2. Operational workflow of the AgroInsect application.

the TensorFlow Lite (TFLite) library, which enables efficient execution of the YOLOv11n model directly on Android devices (Xu et al., 2023; Yu et al., 2024).

To enable near real-time pest identification during field inspections, the AgroInsect app processes captured images locally using the YOLOv11 neural network. The Dart-based implementation, integrated with the TFLite library, performs object detection and renders bounding boxes directly on the images, providing immediate visual feedback to the user.

In this regard, the application architecture follows the MVVM (Model-View-ViewModel) pattern, promoting a clear separation between the user interface (Alcaide Zaragoza et al., 2020), business logic, and data handling layers, thereby improving maintainability, scalability, and long-term system evolution. Fig. 1 illustrates the architecture of the AgroInsect application.

Data generated from insect detections, including bounding box coordinates, user annotations, and geolocation metadata, are stored in a MySQL relational database (Zhou et al., 2024), enabling structured storage and efficient data retrieval.

The operational workflow of the AgroInsect application, illustrated in Fig. 2, integrates user interaction, image acquisition, and data processing into a unified pipeline. The process initiates with user authentication and the selection of the corresponding management unit, ensuring proper spatial attribution of collected records. Subsequently, users select the operational mode, either automatic insect detection or manual annotation, both of which function as complementary mechanisms within the system architecture and maintain consistency between field observations and the persistent database records.

Regardless of the selected mode, the workflow converges to the image acquisition or selection stage, where images can be captured directly via the mobile device camera or selected from local storage. In the automatic detection mode, the acquired images are processed by the embedded computer vision model. When insect detection occurs, the results are automatically recorded. In cases where no detection is made, or when the user identifies incorrect or incomplete detections, the workflow allows a transition to the manual annotation stage, enabling the recording or correction of information related to the observed insects.

For the spatial visualization of occurrences, the Google Maps API was integrated, allowing users to monitor the geographic distribution of detected pests. Additionally, Python scripts were developed to generate heatmaps and spatial interpolations using Ordinary Kriging, implemented through the PyKrige library, with GeoPandas supporting geospatial data handling and map generation. These tools enable the creation of detailed visual representations of infestation within agricultural areas, serving as a decision-making tool based on the distribution and intensity of pests across the monitored territory.

The AgroInsect application was field-tested at the Serra Azul Farm within a 100 × 100 m experimental area, subdivided into a 20 × 20 m sampling grid. Images were captured using a Motorola Edge 30 smartphone (8 GB RAM, Snapdragon 778G+ processor), selected to represent a mid-range commercial device. This configuration allowed for the evaluation of on-device detection performance and system responsiveness under actual field conditions. All detections were processed locally on the device (on-device inference) and georeferenced before being transmitted to the central MySQL database.

While pest detection and image annotation are executed directly on the mobile device to enable practical field operation without network dependency, geostatistical interpolation and density map generation are performed on a cloud-based server due to the higher computational complexity of Kriging-based methods.

2.2. Dataset

The dataset used in this study is the same as that employed in a previous performance analysis of YOLO and Detectron2 models for agricultural pest detection (de Almeida et al., 2024). It comprises images of four insect pest species that commonly affect maize and soybean crops: 591 images of *Diabrotica speciosa*, 177 images of *Dalbulus maidis*, 248 images of *Diceraeus* spp., and 334 images of *Spodoptera frugiperda*. This collection constitutes the AgroInsect dataset.

All images were manually annotated using the open-source software Label Studio (version 0.9.1) (Tkachenko et al., 2020). Annotations included Region of Interest (ROI) bounding boxes, insect class labels, and object positions formatted according to the YOLO annotation standard.

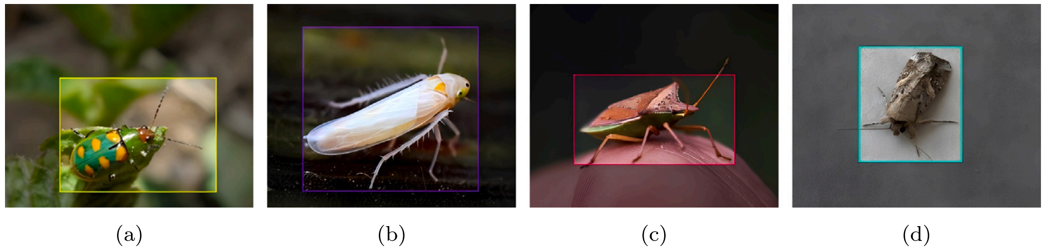


Fig. 3. Representative annotated instances for each insect class used in the training dataset: (a) *Diabrotica speciosa*, (b) *Dalbulus maidis*, (c) *Diceraeus* spp., and (d) *Spodoptera frugiperda*. Bounding boxes indicate manually validated annotations.

Fig. 3 presents representative examples of annotated instances for each insect class.

In total, 1510 images were annotated, considering that some images contained more than one insect of the same species. The annotation files were exported using the same filenames as their corresponding images and organized into training, validation, and test subsets, with dataset paths defined through YAML configuration files.

2.3. YOLOv11n architecture

Among the various available configurations, YOLOv11n was selected for the development of the smartphone application due to its lightweight architecture, which provides an effective trade-off between detection accuracy and computational efficiency. In particular, the nano variant is characterized by a reduced number of parameters and floating-point operations, enabling low-latency inference and lower energy consumption on resource-constrained mobile devices. YOLOv11 belongs to the You Only Look Once family of object detectors, which has been widely adopted for real-time detection tasks due to its single-stage design and competitive performance across diverse application domains (Jocher and Qiu, 2024). Fig. 4 illustrates the main architectural components of the YOLOv11 network that support the object detection task.

Although YOLOv11 supports a broader range of computer vision tasks, including instance segmentation, pose estimation, oriented object detection (OBB), and object tracking, this study exclusively employs its object detection capability, as it best aligns with the requirements of real-time, on-device pest monitoring in agricultural environments.

YOLOv11 introduces several architectural innovations aimed at improving feature representation and robustness in complex visual scenes. These include the C3k2 block (Cross Stage Partial with kernel size 2), the SPPF (Spatial Pyramid Pooling–Fast) module, and the C2PSA (Convolutional block with Parallel Spatial Attention) module. The C3k2 block enhances the extraction of fine-grained details from small objects, which is particularly relevant for insect detection in agricultural images. The SPPF module enables efficient multi-scale feature aggregation, allowing the detection of targets at different spatial scales, including insects at various developmental stages. In addition, the C2PSA block improves spatial feature discrimination through parallel attention mechanisms, increasing robustness in visually cluttered scenarios such as crop fields with dense foliage and heterogeneous backgrounds (Khanam and Hussain, 2024).

In contrast to laboratory-based approaches, AgroInsect was evaluated under non-controlled field conditions, including variable natural illumination and complex crop backgrounds. These conditions highlight the robustness of the YOLOv11n architecture and the effectiveness of its attention mechanisms (C2PSA) for in-situ agricultural pest monitoring.

In this study, the YOLOv11n model was specifically trained for agricultural pest detection and subsequently converted to the TensorFlow Lite (TFLite) format (Evan et al., 2024), enabling its execution on Android devices. This conversion ensured fast inference, low resource consumption, and operational viability under field conditions, which are essential requirements for mobile and edge-based applications. The model processes RGB images and performs on-device inference in near real time, generating bounding boxes that indicate the spatial location of detected pests.

Model training was conducted on the Google Colab platform with GPU acceleration. Detection performance was evaluated using Average Precision (AP) and mean Average Precision (mAP) metrics. For validation, 15 images per species were randomly selected, while the testing phase employed 25 images per species. Even when deployed on hardware-constrained mobile devices, the YOLOv11n model maintained good detection accuracy and stable performance, reinforcing its suitability for embedded computer vision applications in precision agriculture (Oumaima et al., 2025).

2.4. Model training and selection

To identify the most suitable lightweight architecture for on-device agricultural pest detection, three nano-scale YOLO variants were systematically trained and evaluated under identical conditions: YOLOv5n, YOLOv8n, and YOLOv11n. This comparative approach ensures that the selection of YOLOv11n for deployment in the AgroInsect application is supported by empirical evidence rather than architectural assumptions alone.

All models were trained on the Google Colab platform using an NVIDIA T4 GPU (16 GB VRAM). Each model was trained for up to 300 epochs with early stopping (patience = 50), a batch size of 16, and an input image resolution of 640×640 px. Pre-trained weights (ImageNet) were used as initialization for all models. Data augmentation followed the default Ultralytics training pipeline, including random horizontal flipping, mosaic augmentation, HSV color-space jitter, and random scaling (Jocher and Qiu, 2024). Detection performance was evaluated using Average Precision (AP), mAP@0.5, and mAP@0.5:0.95 metrics. For validation, 15 images per species were randomly selected, while the testing phase employed 25 images per species.

Among the evaluated configurations, YOLOv11n was selected for integration into the AgroInsect application due to its superior balance between detection accuracy and computational efficiency, as well as its architectural innovations described in Section 2.3. The selected model was subsequently converted to TensorFlow Lite (TFLite) format under three quantization schemes (Float32, Float16, and INT8) using the Ultralytics export pipeline (Evan et al., 2024).

Tables 1 and 2 present the file sizes obtained after converting the YOLOv5n, YOLOv8n, and YOLOv11n models to TFLite format under three post-training quantization (PTQ) schemes. All models exhibited a consistent reduction in file size as quantization precision decreased: Float16 reduced the model size by approximately 50% relative to Float32, while INT8 achieved reductions of 71.7%, 72.4%, and 71.4% for YOLOv5n, YOLOv8n, and YOLOv11n, respectively. The YOLOv11n-INT8 variant achieved a file size of 2.98 MB, comparable to YOLOv5n-INT8 (2.85 MB) and substantially smaller than YOLOv8n-INT8 (3.33 MB). These compact sizes are well within the storage constraints of mid-range Android smartphones, confirming the suitability of all three architectures for on-device deployment.

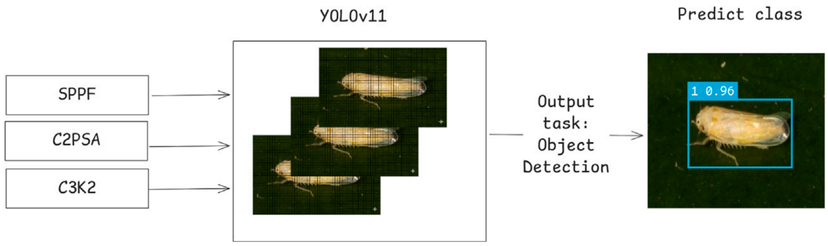


Fig. 4. System architecture of the AgroInsect application, showing on-device inference and cloud integration for metadata and image storage.

Table 1
File size (MB) of YOLOv5n, YOLOv8n, and YOLOv11n models after conversion to TensorFlow Lite format across three quantization schemes: 32-bit floating point (Float32), 16-bit floating point (Float16), and 8-bit integer (INT8).

Model	Float32 (MB)	Float16 (MB)	INT8 (MB)
YOLOv5n	10.09	5.12	2.85
YOLOv8n	12.05	6.10	3.33
YOLOv11n	10.42	5.30	2.98

Table 2
File size (MB) of YOLOv5n, YOLOv8n, and YOLOv11n after conversion to TensorFlow Lite format in Float32 and INT8 quantization schemes, and the corresponding size reduction achieved by INT8 post-training quantization.

Model	Float32 (MB)	INT8 (MB)	Reduction (%)
YOLOv5n	10.09	2.85	71.7
YOLOv8n	12.05	3.33	72.4
YOLOv11n	10.42	2.98	71.4

2.5. On-device model evaluation

To assess the practical deployment feasibility of the proposed models, a systematic on-device evaluation was conducted on a Motorola Edge 30 smartphone (8 GB RAM, Snapdragon 778G+ processor), selected to represent a mid-range commercial device widely available in smallholder and commercial farming contexts.

Four model configurations were evaluated: YOLOv5n, YOLOv8n, and YOLOv11n exported to TFLite format under Float32 quantization, together with YOLOv11n under INT8 quantization. These configurations were selected to represent the full performance spectrum, from the established YOLOv5n-Float32 baseline to the most compact deployable variant (YOLOv11n-INT8), while also including YOLOv8n-Float32 as an intermediate reference point.

Inference performance was measured programmatically using a dedicated Technical Evaluation module integrated into the AgroInsect application. This module recorded average inference latency (ms) and frames per second (FPS) over a standardized batch of 120 images, comprising 30 images per insect class, ensuring balanced class representation across the evaluation set. The evaluation set used for on-device benchmarking is distinct from the field images collected at Serra Azul Farm and used for geostatistical analysis (Section 2.6), as these two evaluations address complementary aspects of the system: detection model performance under controlled conditions and spatial pest distribution under real field sampling, respectively. The TFLite runtime was configured with the XNNPack delegate for CPU-based inference optimization, using 2 threads for Float32 models and 4 threads for the INT8 model. Detection accuracy was assessed using standard object detection metrics: mAP@0.5, mAP@0.5:0.95, Precision, Recall, and F1-Score, computed against manually annotated ground-truth labels following the YOLO annotation standard. The construction of the Precision-Recall curves and the calculation of AP/mAP followed the standard object detection evaluation protocol (Lin et al., 2014; Everingham et al., 2010), in which the metric is obtained by integrating the curve across all confidence levels of the detections produced by the model (conf = 0.001). However, as argued by Wenkel et al. (2021), mAP computed in this manner does not adequately penalize the occurrence of false positives and does not reflect model behavior in a real deployment scenario, in which a specific confidence threshold is always applied. For this reason, Precision, Recall, and F1-Score were additionally reported at a fixed confidence threshold of 0.5, representing the model's practical operating point.

To evaluate sustained operational viability under continuous field use, a 30-minute stress test was conducted for each model configuration. During this period, the application continuously processed the evaluation image set in repeated cycles, simulating realistic field acquisition conditions. The stress test module recorded the following metrics at the end of each session: total inference cycles completed, battery consumption (percentage consumed and estimated hourly drain rate), initial and final device temperature (°C), and thermal variation delta (ΔT, in degrees Celsius). These metrics collectively provide a comprehensive assessment of energy efficiency and thermal behavior under prolonged on-device inference, addressing deployment requirements in field conditions where continuous operation and battery autonomy are critical constraints (Sapna et al., 2026).

The results of this evaluation are presented in Section 3.6.

2.6. Spatial interpolation with Kriging

Kriging is a geostatistical interpolation method widely used to estimate values of a spatially continuous variable from a discrete set of sampled points (Seethalam et al., 2021). Unlike deterministic methods such as Inverse Distance Weighting (IDW), Kriging considers not only the distance between points but also the spatial dependence among samples, which is modeled through a variogram function. This approach allows for the estimation of unknown values based on a weighted linear combination of neighboring observations, optimizing the estimate in terms of minimum error and unbiasedness (Heusler et al., 2025).

The Kriging estimate of a variable $Z(x_0)$ at an unsampled point x_0 can be expressed by Eq. (1):

$$\hat{Z}(x_0) = \sum_{i=1}^{n} \lambda_i Z(x_i) \tag{1}$$

where $\hat{Z}(x_0)$ is the estimated value at location x_0; $Z(x_i)$ represents the observed values at n sampled points; and λ_i denotes the weights assigned to each sampled point, determined such that the sum of the weights equals unity ($\sum \lambda_i = 1$).

The weights λ_i are calculated based on the experimental variogram, which describes the spatial variability of the studied variable as a function of the distance between samples. The variogram is defined by Eq. (2):

$$\gamma(h) = \frac{1}{2N(h)} \sum_{i=1}^{N(h)} [Z(x_i) - Z(x_i + h)]^2 \tag{2}$$

where $\gamma(h)$ is the semivariance for a lag distance h; $N(h)$ is the number of pairs of points separated by the distance h; and $Z(x_i)$ and $Z(x_i + h)$ are the observed values at two locations separated by this distance.

The spherical variogram model was adopted in this study based on its widely recognized suitability for representing spatial dependence structures in ecological and entomological data, where spatial correlation decreases progressively with distance up to a defined range (Seethalam et al., 2021). Although exponential and Gaussian models were also considered, the spherical model provided the best fit to the empirical variogram computed from the field data, as assessed by cross-validation residuals.

Based on the fitted variogram, the Kriging model calculates the spatial covariance matrix and solves the linear system to provide the optimal weights λ_i for each estimated point, ensuring that the estimates exhibit minimum prediction error and incorporate the actual spatial structure of the data (Cho et al., 2022).

In the present study, Ordinary Kriging was implemented using the PyKrige library (Pereira et al., 2022) and integrated into a Flask-based web server to enable automated, on-demand density map generation. The model interpolates discrete insect counts collected in the field, generating a continuous density surface (heatmap) overlaid on the Google Maps interface. This integration transforms raw monitoring data into actionable spatial intelligence, allowing users to visualize the results interactively within the mobile application.

To evaluate the performance of the implemented Kriging model, a synthetic controlled simulation was conducted. A total of 8000 simulated occurrence points were generated, distributed among the four target species, each exhibiting a distinct spatial pattern characterized by a high-density area (hotspot) and a dispersed distribution throughout the remainder of the study region. The experimental area was subdivided into grids of 20 × 20, 50 × 50, 100 × 100, and 200 × 200 cells, representing different sampling density levels. Interpolated maps were subsequently compared with the simulated gradient maps, which served as the reference surface (ground truth), allowing for the quantification of prediction error and the analysis of the method sensitivity to sampling data scarcity.

In addition to the synthetic controlled simulation, Kriging was validated against real field data collected at the Serra Azul Farm. For each of the four target species and for the aggregated pest count, leave-one-out cross-validation (LOO-CV) was performed over the $n = 79$ georeferenced detection points: each point was sequentially withheld, the remaining $n - 1$ points were used to predict its value, and the predicted value was compared against the observed count. Ordinary Kriging was compared against Inverse Distance Weighting (IDW) with power parameters $p = 1$ and $p = 2$, using RMSE, MAE, R^2, and mean bias as evaluation metrics. Beyond point-prediction accuracy, the practical utility of each method's infestation map was assessed by comparing the spatial variance of the predicted surface to the spatial variance of the observed data (variance preservation ratio) and by quantifying the fraction of the observed hotspot magnitude recovered at its location (hotspot capture ratio). This real-field validation complements the synthetic controlled simulation and directly addresses the reliability of Kriging interpolation under sparse, real sampling conditions; results are reported in Section 3.2.1.

3. Results

3.1. Main functionalities of AgroInsect

AgroInsect is a smartphone application designed to support decision-making for farmers in pest management. Fig. 5 illustrates the application's interface. Upon logging into the system, the user selects an associated farm. The system architecture includes modules for image-based pest detection and annotation, as well as management features for users, farms, roles, pest classes, and map visualization.

The image detection and annotation modules operate in an integrated manner. Within the detection module, the user can either capture an image directly using the smartphone camera or upload an image from the device gallery. The selected image is then processed on-device by the trained YOLOv11n model, deployed in the TensorFlow Lite (TFLite) format, for object detection and counting. After inference, the user may save the detection results, manually correct the generated annotations, or add new annotations for objects not automatically identified by the model.

To generate the maps, image geolocation coordinates are required. The system is capable of extracting this information from the EXIF metadata of the images, both for those captured directly by the device camera and those uploaded from the smartphone gallery. To ensure that images are acquired exclusively within the boundaries of the agricultural property associated with the authenticated user, a georeferenced validation based on the farm's operational radius was implemented. Each property has a central point defined by geographic coordinates (`farm_latitude`, `farm_longitude`) and a maximum operational radius (`radius`), expressed in meters.

The distance between the image capture point (extracted from EXIF metadata) and the farm center is calculated using the Haversine formula, as shown in Eq. (3):

$$d = 2R\arcsin\left(\sqrt{\sin^2\left(\frac{\phi_2-\phi_1}{2}\right)+\cos(\phi_1)\cos(\phi_2)\sin^2\left(\frac{\lambda_2-\lambda_1}{2}\right)}\right), \quad (3)$$

In Eq. (3), R denotes the Earth's mean radius, assumed as 6371 km, while ϕ_1 and ϕ_2 represent the latitudes and λ_1 and λ_2 the longitudes of the image capture point and the farm center, respectively, all expressed in radians.

An image is considered valid if the computed distance satisfies the condition:

$$d \le r_{\max}, \quad (4)$$

where $r_{\max}$ represents the maximum operational radius of the farm, expressed in kilometers. This validation ensures the spatial integrity of the dataset by preventing the inclusion of samples acquired outside the monitored agricultural area.

Fig. 6 illustrates the relational database design (MySQL) of the AgroInsect system. The multitenant architecture employs the `user_farm` table to ensure complete data isolation among farms. The schema was designed to provide logical isolation between different agricultural properties, ensuring that each property maintains its own operational data and associated user information.

1. The `user` table stores global authentication and registration information for all users.
2. The `farm` table contains specific data for each agricultural property registered in the system.
3. The `user_farm` junction table establishes the link between users and farms, enabling a many-to-many relationship where a single user can be associated with one or more farms.

Data isolation is ensured by the way operational information, such as images and annotations, is linked to the `user_farm` relationship. Thus, the system guarantees that each user can only save, query, or view images and data belonging to the farms with which they have an active association (a valid record in the `user_farm` table). Users without a link lack access to data from other farms, thereby maintaining information integrity and confidentiality.

3.2. Validation of the Kriging model under a simulated scenario

The validation stage aimed to assess the precision and spatial consistency of the abundance estimates generated by the Kriging model. Widely adopted geostatistical performance metrics were employed to quantify prediction errors and model accuracy, including the Root Mean Square Error (RMSE), Mean Absolute Error (MAE), Mean Bias, and the Coefficient of Determination (R^2). Let Z_i denote the observed (simulated) value at location i, $\hat{Z}_i$ the corresponding predicted value,

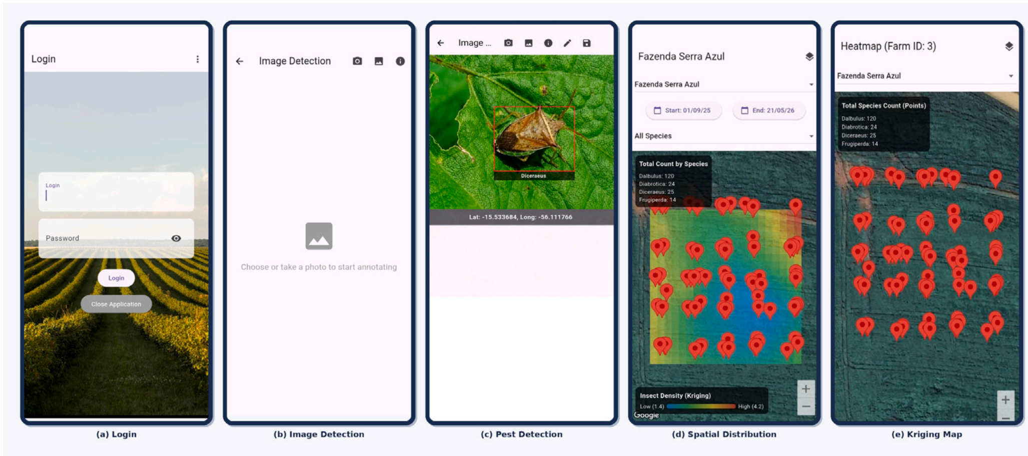


Fig. 5. AgroInsect application interface, showcasing object detection, density heatmap, and Kriging interpolation map.

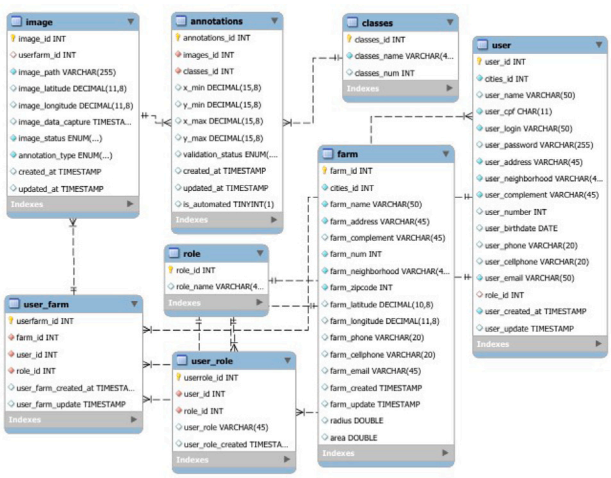


Fig. 6. Relational database structure (MySQL) of the AgroInsect system. The multi-tenant architecture uses the user_farm table to ensure complete data isolation between farms.

$\bar{Z}$ the mean of the observed (simulated) values, and n the total number of observations.

The prediction error was quantified using the Root Mean Square Error (RMSE), defined as:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(Z_i - \hat{Z}_i)^2} \tag{5}$$

Predictive accuracy was further evaluated using the Coefficient of Determination (R^2), expressed as:

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(Z_i - \hat{Z}_i)^2}{\sum_{i=1}^{n}(Z_i - \bar{Z})^2} \tag{6}$$

In addition, the Mean Absolute Error (MAE) and Mean Bias were computed to assess, respectively, the average magnitude of prediction errors and the presence of systematic overestimation or underestimation:

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|Z_i - \hat{Z}_i|, \quad Bias = \frac{1}{n}\sum_{i=1}^{n}(Z_i - \hat{Z}_i) \tag{7}$$

Table 3 summarizes the results obtained for each simulated species across different grid resolutions.

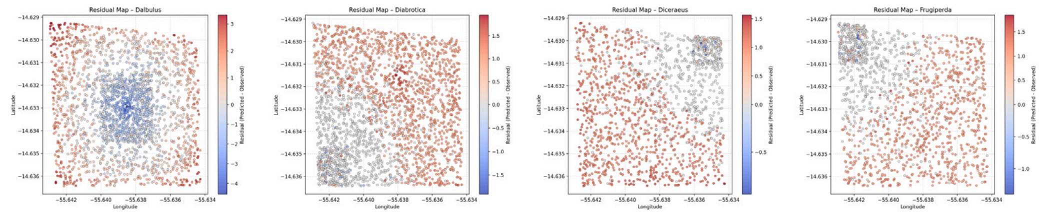
The results demonstrate an overall decrease in interpolation accuracy as sampling density decreases (i.e., with increasing grid size), although minor species-specific variations were observed. All species exhibited their highest interpolation performance under the two densest sampling grids (20 × 20 and 50 × 50), with R^2 values above 0.90 and the lowest prediction errors. Spodoptera frugiperda achieved the highest interpolation accuracy under these dense sampling conditions (R^2 = 0.9656 and 0.9597), whereas Diabrotica speciosa exhibited the most consistent performance across sampling densities, maintaining R^2 values above 0.90 up to the 100 × 100 grid. Diceraeus spp. also performed well under dense sampling, with R^2 values above 0.93 for the 20 × 20 and 50 × 50 grids. As sampling density decreased, interpolation performance generally deteriorated across species, as evidenced by increasing RMSE and MAE values accompanied by lower R^2 values.

In addition to the quantitative validation metrics, the spatial structure of the prediction errors was analyzed to identify potential biases or regional deviations in the estimates obtained by Kriging. The residuals were computed as the difference between the values estimated by the Kriging model and the corresponding reference values from the simulated universe. Let $\hat{Z}_i$ denote the value estimated by Kriging at location i and Z_i the corresponding observed (simulated) value. The residual at location i is therefore defined as:

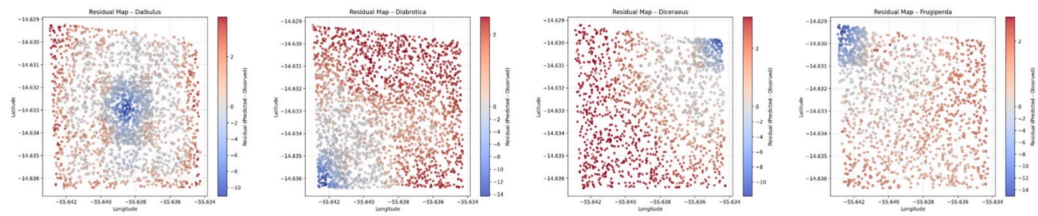
$$e_i = \hat{Z}_i - Z_i \tag{8}$$

Table 3
Performance evaluation of kriging-based spatial interpolation under varying grid resolutions for simulated insect abundance. Reported metrics include training sample size (n_{train}), validation sample size (n_{valid}), root mean square error (RMSE), mean absolute error (MAE), and coefficient of determination (R^2). Lower RMSE and MAE values and higher R^2 values indicate improved interpolation performance.

Species	Grid	n_{train}	n_{valid}	RMSE	MAE	R^2
Diabrotica speciosa	20 × 20	950	2501	0.8070	0.6189	0.9511
	50 × 50	321	2501	0.8830	0.7402	0.9414
	100 × 100	74	2501	1.0341	0.8588	0.9197
	200 × 200	16	2501	2.6374	1.9396	0.4774
Dalbulus maidis	20 × 20	1963	2455	1.2596	1.0152	0.9113
	50 × 50	761	2455	1.2971	0.9913	0.9060
	100 × 100	198	2455	2.6761	1.9419	0.5997
	200 × 200	46	2455	2.3932	1.6535	0.6799
Diceraeus spp.	20 × 20	558	1515	0.8073	0.6323	0.9478
	50 × 50	272	1515	0.8748	0.7288	0.9388
	100 × 100	76	1515	1.1860	1.0217	0.8874
	200 × 200	13	1515	2.7187	2.2817	0.4085
Spodoptera frugiperda	20 × 20	601	1529	0.7664	0.5860	0.9656
	50 × 50	287	1529	0.8300	0.6919	0.9597
	100 × 100	70	1529	1.4723	1.1575	0.8731
	200 × 200	19	1529	2.9628	1.9192	0.4859



(a) *Dalbulus maidis* (20×20) (b) *Diabrotica speciosa* (20×20) (c) *Diceraeus* spp. (20×20) (d) *Spodoptera frugiperda* (20×20)



(e) *Dalbulus maidis* (200×200) (f) *Diabrotica speciosa* (200×200) (g) *Diceraeus* spp. (200×200) (h) *Spodoptera frugiperda* (200×200)

Fig. 7. Spatial distribution of Kriging residuals for the four simulated insect species under two sampling densities (20 × 20 and 200 × 200 grids). Positive values indicate overestimation, while negative values denote underestimation of abundance.

Fig. 7 illustrates the spatial distribution of these residuals for both sampling configurations (20 × 20 and 200 × 200).

The residual maps show that prediction errors exhibit spatial structures consistent with the simulated spatial patterns. In the synthetic dataset, each species exhibited a high-density region (hotspot) and a gradual dispersion across the remaining area. Consequently, a trend of underestimation was observed in high-concentration zones, while overestimation occurred in peripheral regions. This smoothing effect is especially evident for *Dalbulus maidis* and *Spodoptera frugiperda*, reflecting the inherent behavior of Kriging, which produces smoothed local estimates around concentrated data points.

Conversely, *Diabrotica speciosa* showed a more homogeneous residual distribution, consistent with its simulated spatial pattern. This observation is consistent with the quantitative results presented in Table 3, where *Diabrotica speciosa* exhibited the most consistent interpolation performance across sampling densities, maintaining R^2 values above 0.90 up to the 100 × 100 grid. Together, these observations indicate that the Kriging model adequately reconstructed the simulated spatial patterns under dense and intermediate sampling densities.

Overall, the spatial analysis of residuals supports the consistency of the implemented Kriging procedure under the simulated scenarios evaluated in this study. Even with simplified spatial patterns, the interpolation reproduced the main spatial gradients without evidence of systematic directional artifacts in the residual distribution. Although prediction accuracy generally decreased under lower sampling densities, the interpolation procedure remained stable across all simulated sampling configurations (20 × 20 to 200 × 200), supporting the feasibility of the proposed implementation prior to broader validation under real field conditions. Future work should incorporate larger georeferenced field datasets together with auxiliary covariates, such as vegetation indices, soil attributes, and climatic variables, to further evaluate model generalizability and enhance predictive performance under low-sampling-density conditions.

3.2.1. Validation of Kriging vs. IDW using real field data

The synthetic validation reported in Section 2.6 quantifies point-prediction accuracy, but the operational purpose of AgroInsect's geostatistical module is not point prediction in isolation: it is the generation of continuous infestation maps that support site-specific decision-making, such as localized pesticide application (Section 2.6, Fig. 5). Evaluating Kriging solely against this point-prediction criterion is therefore an incomplete test of its suitability for the application. To assess both dimensions under real field conditions, Ordinary Kriging was compared against Inverse Distance Weighting (IDW, $p = 1$) using leave-one-out cross-validation (LOO-CV) on the $n = 79$ georeferenced detections collected at the Serra Azul Farm. Table 4 reports RMSE, MAE, R^2, and mean bias for each species and for the aggregated pest count, together with two complementary indicators of practical map utility: the percentage of observed spatial variance preserved by each method, and the percentage of the observed hotspot magnitude captured at its location.

Under this real-field, sparse sampling regime, Kriging was not superior to IDW in point-prediction accuracy. Point-prediction accuracy was broadly comparable between the two interpolation methods: IDW ($p = 1$) yielded slightly lower RMSE values for *Dalbulus maidis*, *Diceraeus* spp., and the aggregated pest count, whereas Kriging marginally outperformed IDW for *Diabrotica speciosa*. Both methods produced identical RMSE for *Spodoptera frugiperda*. Negative R^2 values were obtained for both methods across all categories, indicating that, under $n = 79$ sparse sampling, neither interpolator improved on a simple mean-based prediction; this likely reflects the combined effects of sparse sampling and heterogeneous spatial distribution rather than a deficiency specific to either method, and is consistent with the accuracy degradation already observed for the sparsest synthetic grids in Table 3.

Despite this comparable point-prediction accuracy, Kriging preserved more spatial variance than IDW for *Dalbulus maidis* (7.5% vs. 1.8%) and for the aggregated count (8.3% vs. 2.1%), and captured a larger share of the observed hotspot magnitude for these two categories (13.9% vs. 9.2%, and 23.1% vs. 18.6%, respectively). This advantage was not uniform across all species: for *Diabrotica speciosa* and *Diceraeus* spp., IDW preserved comparable or slightly higher variance than Kriging. Fig. 8 illustrates this pattern for the aggregated pest count. Fig. 8a shows observed versus predicted values under LOO-CV, where both methods exhibit similar deviation from the 1:1 line and similarly underestimate the observed hotspot (observed count of 10, both methods predicting approximately 2). Fig. 8b shows the percentage of spatial variance preserved by each method for every species, highlighting *Dalbulus maidis* and the aggregated count as the categories in which Kriging's advantage was most consistent.

Taken together, these results indicate that Kriging's justification within AgroInsect does not rest on superior point-prediction accuracy—under the current sparse real-field sampling density, a well-tuned IDW

Table 4
Leave-one-out cross-validation (LOO-CV) comparison between Ordinary Kriging and IDW ($p = 1$) using real field detection data (Serra Azul Farm, $n = 79$ georeferenced points), by species and for the aggregated pest count ($n_{valid} = 79$ in all cases). RMSE, MAE, and Bias are reported in counts per point; R^2 is dimensionless. Preserved variance and hotspot capture are reported as a percentage of the corresponding observed (true) value.

Species	Method	n_{valid}	RMSE	MAE	R^2	Bias	Preserved variance (%)	Hotspot captured (%)
Diabrotica speciosa	IDW ($p = 1$)	79	0.631	0.492	−0.176	−0.008	4.0	6.3
	Kriging	79	0.630	0.490	−0.175	−0.004	2.5	3.4
Dalbulus maidis	IDW ($p = 1$)	79	2.519	2.031	−0.081	0.175	1.8	9.2
	Kriging	79	2.620	2.183	−0.170	0.026	7.5	13.9
Diceraeus spp.	IDW ($p = 1$)	79	0.681	0.541	−0.260	−0.014	4.3	6.0
	Kriging	79	0.693	0.543	−0.305	−0.004	6.9	5.5
Spodoptera frugiperda	IDW ($p = 1$)	79	0.471	0.323	−0.132	−0.019	8.7	4.1
	Kriging	79	0.471	0.307	−0.130	−0.002	11.4	4.2
Total count	IDW ($p = 1$)	79	2.055	1.565	−0.072	0.133	2.1	18.6
	Kriging	79	2.108	1.678	−0.128	0.016	8.3	23.1

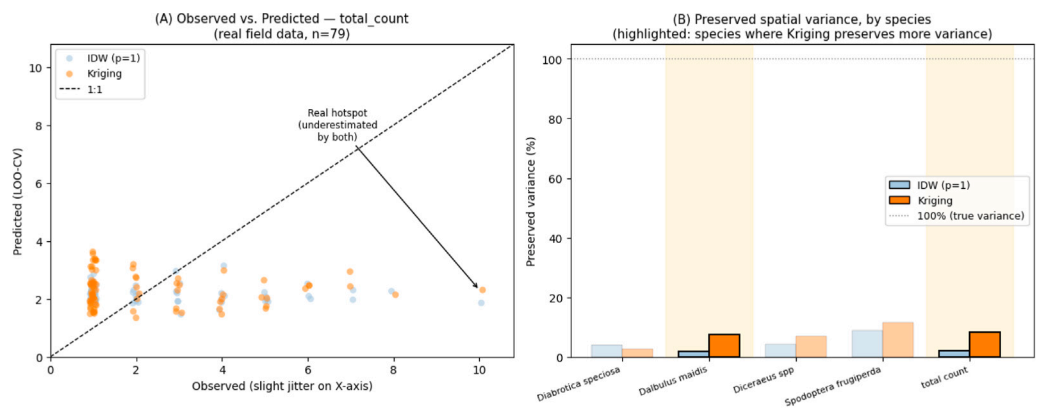


Fig. 8. Leave-one-out cross-validation comparison between Ordinary Kriging and IDW ($p = 1$) using real field detection data (Serra Azul Farm, $n = 79$ georeferenced points). (a) Observed vs. predicted aggregated pest counts (total count); both methods show comparable behavior and similarly underestimate the observed hotspot, indicating that this limitation stems from sparse sampling density rather than from the choice of interpolation method. (b) Percentage of observed spatial variance preserved by each method, by species; Kriging retains substantially more spatial contrast than IDW for *Dalbulus maidis* and the aggregated count (highlighted), producing a more informative infestation map despite comparable point-prediction error.

provides comparable predictive performance. Instead, the rationale for adopting Kriging lies in the practical utility of the resulting infestation maps, as it better preserved spatial contrast and hotspot representation for *Dalbulus maidis* and for the aggregated pest count, the categories of greatest agronomic and operational relevance for the application. Although this advantage was not observed uniformly across all species, the present field evaluation should therefore be interpreted as a feasibility demonstration of the proposed spatial analysis workflow within AgroInsect rather than as definitive validation of large-scale field performance. Nevertheless, the current validation dataset ($n = 79$) remains limited, and additional georeferenced observations collected across different dates, fields, environmental conditions, and acquisition devices will be essential to further assess the robustness and generalizability of the proposed approach.

3.3. Evaluation of automated field detections

The evaluation stage aimed to quantify the performance of the insect recognition model using a smartphone application. Each detected object was manually reviewed and classified into three categories: (i) Correct detection (D), when the insect was correctly identified; (ii) Class adjustment (A), when the detection was valid, but the species classification required correction; and (iii) Exclusion (E), applied to false positives or objects without adequate morphological definition.

Table 5 summarizes the detections performed by the model across the 79 evaluated images. Out of 183 identified objects, 92.4% were correctly classified, 6.6% required class adjustment, and 1.1% were excluded due to detection errors. These results evidence consistent model performance, with a low occurrence of false positives and limited need for manual intervention.

Table 5
Summary of detections and class adjustments.

Category	Quantity	Description
Correct detections (D)	169	Detections with correct classification.
Class adjustments (A)	12	Detections requiring manual correction.
Exclusions (E)	2	Objects discarded due to detection error.
Total valid	**183**	**Total objects considered after exclusions.**

Fig. 9 presents the confusion matrix derived from cross-validation, synthesizing the classifier's performance in discriminating the four target insect species and the background class. High agreement between predicted and true labels is observed, reflected in high sensitivity values per class. The few identified inconsistencies are concentrated in specific class pairs and may be associated with variations present in the images used during the validation process.

The quantitative performance metrics are presented in Table 6. The model achieved an overall accuracy of 95.1%, with precision, recall, and F1-score values exceeding 90% for all classes of agronomic interest. The species *Dalbulus maidis* presented an F1-score of 98%, indicating an excellent balance between sensitivity and precision. Equally robust performance was observed for *Diabrotica speciosa* (98%) and *Spodoptera frugiperda* (96%). Occasional errors, such as sporadic false negatives where individuals of *Diabrotica speciosa* were classified as background,

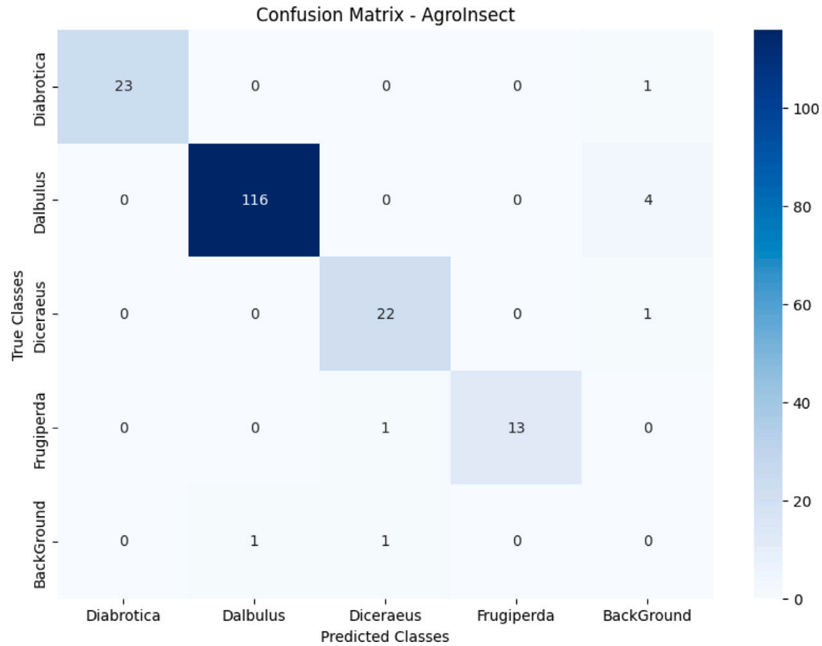


Fig. 9. Confusion matrix of the AgroInsect model evaluated with field-collected images.

Table 6
Classification performance of the model for the evaluated species.

Class	Precision	Recall	F1-score	Samples
Diabrotica speciosa	1.00	0.96	0.98	24
Dalbulus maidis	0.99	0.97	0.98	120
Diceraeus spp.	0.92	0.96	0.94	23
Spodoptera frugiperda	1.00	0.93	0.96	14
Overall accuracy		95.1%		

explain the slight reduction in the recall value for this class (96%), without compromising the model's overall performance.

The combined results of the automated evaluation and manual review reinforce the potential of the AgroInsect system as a tool to support automated pest monitoring. The high accuracy and low requirement for corrections indicate that the model can significantly reduce human effort during the screening stage while maintaining consistency and reliability in records. For future work, the use of active learning strategies and iterative refinement is recommended to enhance robustness against variability in morphology and lighting conditions within field images.

3.4. Spatial analysis of pest occurrences

Fig. 10 presents the results generated by the system after applying filters based on the user's associated farms, date range, and pest species. The interface returns three complementary components: **(a)** a quantitative dashboard showing the total number of insects detected on the property within the selected period; **(b)** a spatial distribution map displaying the geographical location of each detection; and **(c)** a Kriging map representing the interpolation of insect density within the field.

Together, these elements allow for the evaluation of not only the intensity of occurrences but also the spatial dynamics of the pests, providing objective data to support decision-making in site-specific management.

3.5. Results of the integrated spatial dashboard of the AgroInsect system

Fig. 11 presents the set of maps derived from the spatial analysis performed by the AgroInsect system. The dashboard integrates all stages of the processing workflow, from detections produced by the model to spatial interpolation using Kriging, allowing a clear and quantitative evaluation of pest distribution throughout the monitored field.

The aggregated map (Fig. 11a) synthesizes the combined occurrence of all species, providing a general overview of the insect pressure during the analyzed period and highlighting regions of higher concentration that prompted individual assessments for each pest.

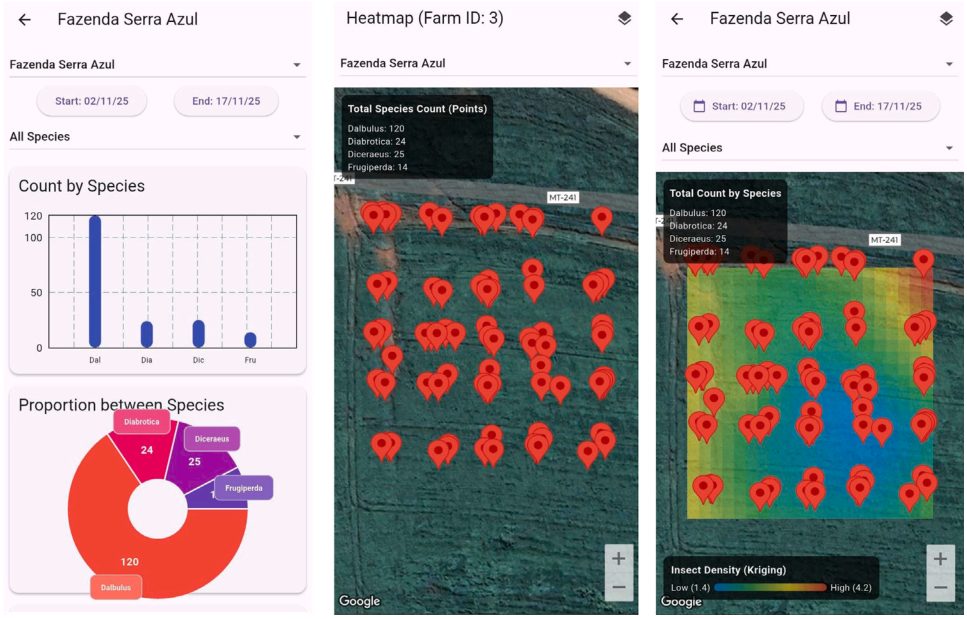
For *Diabrotica speciosa* (Fig. 11b), the spatial projection reveals a highly localized distribution pattern. Areas of higher intensity are concentrated in the immediate vicinity of the sampling points, particularly in the central region of the field. The absence of broader dispersal gradients suggests localized infestation foci, consistent with the lower mobility of this species and records of restricted spatial aggregation mentioned in the literature.

The Kriging applied to the *Dalbulus maidis* species (Fig. 11c) resulted in the broadest and most continuous spatial pattern among all evaluated species. The formation of smooth gradients and higher intensity regions is observed, distributed predominantly in the eastern and southwestern portions of the field. This behavior is coherent with the higher displacement capacity associated with the species, resulting in an interpolated surface that is more homogeneous and features more progressive transitions between zones of low and high occurrence probability.

The species *Diceraeus* spp. (Fig. 11d) presents an intermediate pattern. Although most of the area maintains low values, an elevated intensity core is observed in the central portion of the field. This focus may be associated with specific microenvironmental characteristics that favor its presence. Nonetheless, the distribution as a whole remains fragmented, suggesting moderate population pressure during the evaluated period.

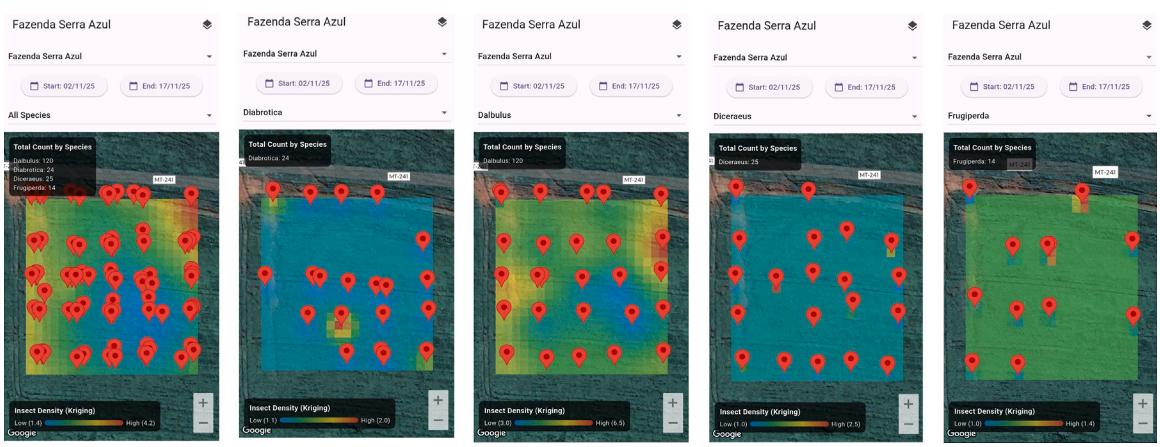
Finally, *Spodoptera frugiperda* (Fig. 11e) exhibited the most homogeneous spatial pattern with the lowest absolute magnitude. The interpolated surface shows minimal variation across the field, with only small areas of slight increase and no formation of robust infestation patches. This behavior confirms the low counts observed during the detection stage and suggests that the species was dispersed and at low density during the monitored interval.

In addition to spatial visualization and decision-support capabilities, the practical deployment of the AgroInsect system also depends on



(a) Detection count (b) Spatial distribution (c) Kriging interpolation

Fig. 10. Integrated spatial dashboard of the AgroInsect system: (a) total detected insect occurrences; (b) geographic distribution of detections across the monitored agricultural area; (c) kriging-based spatial interpolation representing the estimated insect occurrence density.



(a) All species (b) *Diabrotica* (c) *Dalbulus* (d) *Diceraeus* spp. (e) *S. Frugiperda*

Fig. 11. Kriging incidence maps: (a) aggregated distribution of all species; (b) *Diabrotica speciosa*; (c) *Dalbulus maidis* (d) *Diceraeus* spp.; (e) *S. frugiperda*.

efficient on-device inference performance, which is evaluated in the following section.

3.6. Model comparison and on-device performance

Tables 7, 8, and 9 present the on-device performance results obtained on the Motorola Edge 30 smartphone across all evaluated model configurations.

It is important to note that the mAP@0.5:0.95 values obtained in this study (37.6% for the selected YOLOv11n configuration) are considerably lower than the corresponding mAP@0.5 values (79.3%). This gap is expected: mAP@0.5:0.95 imposes a substantially stricter localization criterion, requiring accurate bounding box overlap across ten IoU thresholds (0.5 to 0.95), whereas mAP@0.5 tolerates comparatively loose overlap. A comparable gap between these two metrics has been reported in other recent agricultural object detection studies employing lightweight YOLO architectures (Chung et al., 2026), indicating that this behavior is not specific to the present model but rather a broader characteristic of nano-scale detectors evaluated under stricter IoU criteria. Small, morphologically irregular targets such as insects are particularly prone to a more pronounced version of this effect: Cheng et al. (2026) show that the conventional CIoU loss function, employed by default in the YOLOv5n, YOLOv8n, and YOLOv11n architectures used in this study, degrades when predicted and ground-truth boxes share similar proportions but differ in residual dimensional details, an error pattern that penalizes mAP@0.5:0.95 far more heavily than mAP@0.5. Given that the operational requirement of AgroInsect is pest presence detection and counting rather than sub-pixel bounding box precision, mAP@0.5 is adopted as the primary metric for deployment assessment, while improving bounding box regression accuracy through alternative loss formulations is identified as a direction for future work.

Table 7 shows that YOLOv11n-Float32 achieved the lowest mean inference latency among all evaluated configurations (2056 ms, 0.5 FPS), outperforming both YOLOv5n-Float32 (2415 ms) and YOLOv8n-Float32 (3076 ms). Notably, INT8 quantization of YOLOv11n preserved

Table 7
On-device benchmark results for YOLOv5n-Float32, YOLOv8n-Float32, YOLOv11n-Float32, and YOLOv11n-INT8 evaluated on a Motorola Edge 30 smartphone (Snapdragon 778G+, 8 GB RAM) over 120 images (30 per class). Inference latency and frames per second (FPS) were measured programmatically using the AgroInsect Technical Evaluation module with the XNNPack delegate (2 threads for Float32; 4 threads for INT8). Model size refers to the TFLite file size after post-training quantization.

Metric	YOLOv5n Float32	YOLOv8n Float32	YOLOv11n Float32	YOLOv11n INT8
mAP@0.5 (%)	79.6	75.1	79.3	79.3
mAP@0.5:0.95 (%)	38.9	34.7	37.6	37.6
Precision (%)	84.4	82.9	84.4	84.4
Recall (%)	81.0	78.9	81.0	81.0
F1-Score (%)	82.6	80.8	82.6	82.6
Mean inference latency (ms)	2415	3076	2056	2056
FPS	0.4	0.3	0.5	0.5
Model size (MB)	10.09	12.05	10.42	2.98

Table 8
Average Precision (AP, %) per insect class at IoU threshold of 0.5 for all evaluated model configurations, measured on the 120-image evaluation set (30 images per class) on a Motorola Edge 30 smartphone (Snapdragon 778G+, 8 GB RAM).

Species	YOLOv5n Float32	YOLOv8n Float32	YOLOv11n Float32	YOLOv11n INT8
Diabrotica speciosa	90.9	90.9	90.9	90.9
Diceraeus spp.	86.7	76.2	85.9	85.9
Spodoptera frugiperda	85.5	77.9	90.9	90.9
Dalbulus maidis	55.4	55.2	49.6	49.6
mAP@0.5	**79.6**	**75.1**	**79.3**	**79.3**

Table 9
Thirty-minute stress test results for all evaluated model configurations on a Motorola Edge 30 smartphone (Snapdragon 778G+, 8 GB RAM). Battery consumption and thermal metrics were recorded by the AgroInsect stress test module under continuous inference cycling over the 120-image evaluation set.

Metric	YOLOv5n Float32	YOLOv8n Float32	YOLOv11n Float32	YOLOv11n INT8
Battery consumed (%)	2	8	2	4
Estimated drain rate (% h^{-1})	4.0	16.0	4.0	8.0
Completed inference cycles	75	309	95	175
Initial temperature (°C)	33.0	39.0	36.0	29.0
Final temperature (°C)	30.0	39.0	28.0	30.0
Thermal variation ΔT (°C)	−3.0	0.0	−8.0	+1.0

detection accuracy without degradation relative to its Float32 counterpart across all metrics, while reducing the model size by 71.4% (from 10.42 MB to 2.98 MB). This result demonstrates that post-training INT8 quantization is a viable strategy for reducing storage requirements on resource-constrained devices without sacrificing detection performance.

The stress test results (Table 9) reveal substantial differences in energy efficiency across configurations. YOLOv8n-Float32 exhibited the highest battery drain rate (16.0% h^{-1}), four times greater than both YOLOv5n-Float32 and YOLOv11n-Float32 (4.0% h^{-1} each). The YOLOv11n-INT8 configuration consumed an intermediate drain rate of 8.0% h^{-1}. Thermal behavior remained stable across all configurations, with thermal variation ΔT ranging from −8.0 °C to +1.0 °C, indicating no thermal throttling risk during sustained field operation.

Per-class Average Precision results (Table 8) indicate that *Diabrotica speciosa* achieved the highest and most consistent AP across all models (90.9%), reflecting its visually distinctive morphology and the higher number of training samples available for this species (591 images). *Dalbulus maidis* presented the lowest AP values across all configurations (49.6%–55.4%), which may be attributed to higher intraclass visual variability and the relatively limited number of training samples (177 images) for this species. These findings highlight the need for targeted data collection efforts for underrepresented species in future dataset expansions.

To complement the fixed-threshold metrics reported in Tables 6 and 7, Fig. 12 presents the Precision-Recall and F1-Score curves across the full range of confidence thresholds for each evaluated model configuration, illustrating the trade-off between precision and recall that underlies the mAP@0.5 and per-class AP values summarized above.

Consistent with the lower AP values discussed above, the Precision-Recall curves in Fig. 12 show that *Dalbulus maidis* exhibits the earliest and steepest decline in precision as recall increases across all four configurations, resulting in the lowest peak F1-Score among all classes (0.59–0.63, compared to 0.85–0.95 for the remaining species). The F1-Score curves further support the choice of a fixed confidence threshold of 0.5 as a practical operating point for the AgroInsect application. For *Diabrotica speciosa*, *Diceraeus* spp., and *Spodoptera frugiperda*, the threshold that maximizes F1-Score consistently falls within the 0.27–0.52 range across the four model configurations, and the corresponding F1-Score curves remain nearly flat in the neighborhood of conf = 0.5, indicating that operating at this threshold results in a negligible performance loss relative to each class's individual optimum. For *Dalbulus maidis*, although its optimal threshold varies more widely across configurations (0.26–0.63), its F1-Score curve is comparatively flat over a broad low-to-moderate confidence range before declining sharply beyond conf ≈0.7–0.8, meaning that conf = 0.5 still falls within, or close to, this near-plateau region rather than in a steeply penalized segment of the curve. Consequently, adopting a single global threshold of 0.5 represents a reasonable compromise across species with heterogeneous detection difficulty, avoiding the additional operational complexity of per-class threshold calibration without disproportionately penalizing the model's weakest-performing class.

Overall, YOLOv11n in both Float32 and INT8 configurations demonstrated the best balance between detection accuracy, inference latency, model compactness, and energy efficiency, confirming its suitability as the selected architecture for on-device pest monitoring in resource-constrained agricultural environments.

4. Discussion

An important aspect of the AgroInsect framework is the integration of a human-in-the-loop mechanism through manual annotation. This functionality allows users to correct or complement automatic detections, enabling continuous dataset curation and mitigating potential model bias over time. As a result, the system supports the progressive improvement of detection performance under real-world conditions.

The results demonstrate that the AgroInsect system efficiently integrates modern computer vision techniques, geoprocessing, and spatial modeling, offering a complete workflow for field pest monitoring. This type of architecture, which combines mobile data capture, automated geographic validation, and server-side Kriging execution to return interactive graphical outputs to the mobile application, remains sparsely explored in the literature, where most solutions address only isolated components of the monitoring process. For instance, Zhou et al. (2024) present a YOLO-based system for orchards; however, its mapping stage depends on external services such as ArcGIS Server, which increases operational complexity and integration costs.

A notable difference was observed between the on-device benchmark results (mAP@0.5 = 79.3%) and the field evaluation accuracy (95.1%). This discrepancy can be partially attributed to the distinct image acquisition conditions inherent to each evaluation protocol. In the benchmark, inference was performed over a fixed set of 120 images comprising varied lighting conditions, angles, and focal distances, without operator control over image quality. In contrast, during field evaluation, the user retains full control over image capture, naturally

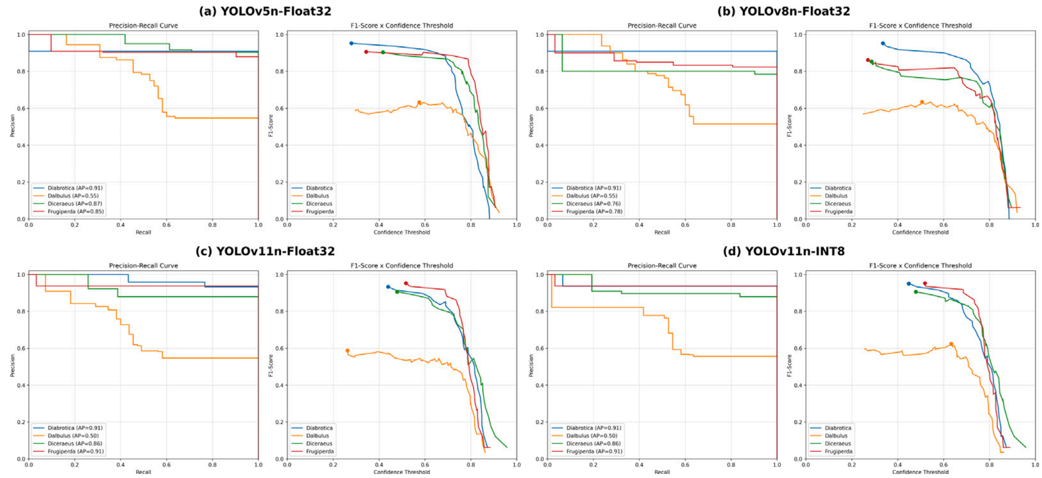


Fig. 12. Precision-Recall and F1-Score curves for the four evaluated on-device model configurations: (a) YOLOv5n-Float32, (b) YOLOv8n-Float32, (c) YOLOv11n-Float32, and (d) YOLOv11n-INT8. For each configuration, the left panel shows the Precision-Recall curve with the Average Precision (AP) per insect class, and the right panel shows F1-Score as a function of confidence threshold, with dots marking each class's optimal operating point.

selecting the most favorable angle, distance, and focus before triggering detection. This human-in-the-loop acquisition behavior effectively filters out low-quality inputs prior to inference, contributing to higher classification consistency under real operational conditions. These findings suggest that reported benchmark metrics should be interpreted as a conservative lower bound of expected field performance, rather than a direct predictor of accuracy in assisted monitoring workflows.

The classification model performance, with an overall accuracy of 95.1%, is consistent with recent advances in lightweight neural network architectures designed for mobile devices. Xu et al. (2025) describe the GBiDC-PEST model, optimized for Android smartphones, achieving overall high precision while significantly reducing computational cost. Comparable results were obtained with AgroInsect, reinforcing that compressed models adapted for TFLite can maintain competitive performance even on hardware-constrained devices. This characteristic is particularly relevant for adoption by rural producers operating in low-infrastructure environments.

Furthermore, Verma et al. (2021) present a detailed performance evaluation of different YOLO versions (v3, v4, and v5) for soybean insect detection, achieving a mean Average Precision (mAP) of up to 99.5% and an F1-score of 96% with YOLOv5. While these results highlight the effectiveness of YOLO models for precise detection in agricultural scenarios, their approach remains dependent on high-end GPU processing and is limited to the detection stage. In contrast, AgroInsect relies on standard smartphones for user interaction and data capture while delegating heavy spatial processing to a server-side environment, eliminating the need for specialized field hardware and integrating detection, automatic geospatial validation, Kriging interpolation, and infestation map generation into a single automated workflow. Consequently, the proposed system advances beyond isolated detection toward a functional spatial monitoring solution aligned with the practical demands of precision agriculture.

A relevant comparison can also be made with cloud-based approaches. Karar et al. (2021) propose a pipeline in which the smartphone functions only as an interface, while inference using Faster R-CNN is performed on remote servers. Although this approach achieves overall high accuracy, it depends on continuous network connectivity, automatic data retransmission, and scalable hosting infrastructure. In agricultural environments, where connectivity is often limited, such dependencies can compromise practical adoption. In contrast, AgroInsect performs detection locally on the device, ensuring immediate, low-latency visual feedback during field sampling without requiring continuous network coverage. Server-side processing is reserved for generating aggregated spatial maps when connectivity is available, providing a balanced workflow adapted to low-infrastructure environments.

Recent studies have also reported significant advances in detecting small insects in heterogeneous agricultural environments. Tang et al. (2025) introduce S-YOLOv5m, an enhanced YOLOv5m variant for tomato crops that incorporates SPD-Conv, Ghost modules, CBAM, and the AEM block. These structural modifications yielded clear improvements in detection accuracy, model compression, and inference speed. Despite these performance gains, the model remains focused exclusively on the detection task, without incorporating spatial validation or distribution mapping. In contrast, AgroInsect combines on-device detection with automated spatial validation and server-side geostatistical modeling within a unified framework, offering an accessible end-to-end monitoring tool for low-infrastructure contexts.

Complementary progress is observed in the work of Kargar et al. (2025), who propose an ultra-lightweight model for insect segmentation and counting on microcontrollers with severely constrained memory resources. Although the model achieves satisfactory segmentation performance with a substantially reduced parameter count, its scope is limited to local segmentation and counting tasks, without geographic validation or spatial analysis. In this respect, AgroInsect extends local processing by combining on-device detection and georeferencing with server-side interpolation and automated map generation, delivering a complete monitoring pipeline accessible via conventional smartphones.

Similarly, Wang et al. (2025) introduce Insect-YOLO, which incorporates attention mechanisms to improve feature extraction in low-resolution images. While the model achieves a favorable balance between efficiency and accuracy, its application is described within IoT-based systems that rely on dedicated infrastructure and remote processing. The authors also report limitations related to environmental sensitivity, inter-species performance variability, and potential overfitting due to region-specific datasets. AgroInsect addresses these limitations by performing on-device inference directly on user-acquired field images, eliminating the need for fixed IoT sensors or specialized capture hardware while leveraging server-side spatial modeling for map generation.

In addition, smart trap systems equipped with computer vision and IoT capabilities, such as those described by Qin et al. (2024), enable large-scale automated data collection but require fixed installations, permanent infrastructure, and reliable connectivity. By contrast, AgroInsect democratizes pest monitoring by leveraging the farmer's own smartphone for image capture and on-device detection, while delegating spatial modeling to server-side infrastructure. This hybrid model substantially increases accessibility in regions where dedicated sensing hardware is economically or logistically unfeasible.

The geostatistical analysis conducted in this study revealed spatial patterns consistent with classical investigations applying Kriging to pest monitoring. The observed smoothing effects, including underestimation in high-density areas and overestimation near field boundaries, have also been reported in studies involving pests such as *Helicoverpa armigera*, billbugs, and grape root borer (Moral García, 2006). Moreover, the spatial differences observed among species, such as the localized distribution of *Diabrotica* and the more diffuse pattern of *Dalbulus*, are consistent with documented ecological behaviors related to mobility, aggregation, and dispersal.

Another relevant contribution of this work lies in the automated handling of spatial information. The georeferenced validation based on EXIF metadata and the operational radius of each property reduces positioning errors that are common in systems relying on manual coordinate input or external GPS devices. This approach addresses a recurring limitation in mobile agricultural applications and strengthens the spatial reliability of AgroInsect. Notably, none of the reviewed studies, including IoT-based systems, lightweight detection models, or cloud-dependent solutions, integrate geospatial validation, interpolation, and automated map generation within a unified framework.

Overall, the results demonstrate that AgroInsect delivers a complete, end-to-end monitoring workflow that seamlessly encompasses mobile image acquisition, on-device detection, human-in-the-loop annotation, spatial validation, and server-side geostatistical mapping. The proposed system provides a practical, accurate, and operationally efficient tool to support integrated pest management under real field conditions.

5. Conclusion and future work

This study presented AgroInsect, a hybrid edge-cloud framework designed for agricultural pest monitoring under real field conditions. By integrating on-device deep learning, georeferenced validation, and server-side geostatistical modeling, the proposed system addresses key limitations of existing approaches that rely on continuous cloud-dependent inference, external GIS platforms, or controlled acquisition environments. The main findings and contributions of this work are summarized below:

(1) Practical applicability: The proposed framework demonstrates the feasibility of performing accurate pest detection directly on standard smartphones using a lightweight YOLOv11n model deployed in TensorFlow Lite format. By enabling on-device inference for field image capture and sampling, the system provides low-latency visual feedback and operates independently of network connectivity during field navigation—critical requirements in rural and low-infrastructure agricultural settings. Heavy geostatistical interpolation is offloaded to the server side, eliminating the need for specialized field hardware while ensuring agricultural consultants and growers can use widely available mobile devices for spatial monitoring. In addition to in-field pest detection, the system functions as a practical tool for generating annotated image datasets through a human-in-the-loop strategy that allows users to correct or create annotations, supporting continuous dataset curation and model refinement in real-world scenarios.

(2) System effectiveness: Experimental results obtained under non-controlled field conditions, including variable illumination and complex crop backgrounds, indicate that the system achieves good detection accuracy while maintaining stable performance on resource-constrained devices. The subsequent geospatial validation based on EXIF metadata and property boundaries ensures spatial consistency of the collected data. Kriging-based interpolation enabled the generation of infestation maps that reveal meaningful spatial patterns consistent with known ecological behaviors of the analyzed pest species, supporting decision-making within Integrated Pest Management (IPM) programs.

(3) Limitations and future work: Despite its effectiveness, the current implementation is limited to four pest species and two major crops, reflecting constraints in the availability of annotated field data. The geostatistical component was validated both under a dense synthetic simulation and, more directly, through leave-one-out cross-validation against IDW on real field detections (Section 3.2.1); however, this real-field validation set remains limited (n = 79), and Kriging's advantage in map utility over IDW was not uniform across all species. Future work will focus on expanding the dataset to additional species and crops, enlarging the georeferenced field dataset to strengthen the real-field geostatistical validation, incorporating temporal analysis for population dynamics, and evaluating alternative geostatistical and spatio-temporal interpolation methods. Further investigation into automated uncertainty estimation and adaptive model updating is also planned to enhance robustness and scalability for large-scale agricultural deployment.

CRediT authorship contribution statement

Guilherme Pires Silva de Almeida: Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Leonardo Nazário Silva dos Santos:** Writing – review & editing, Supervision, Resources, Methodology, Investigation, Funding acquisition, Conceptualization. **Ruy de Oliveira:** Writing – review & editing, Writing – original draft, Validation, Software, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Marconi Batista Teixeira:** Writing – review & editing, Resources, Funding acquisition, Conceptualization. **Mario De Oliveira:** Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Pablo da Costa Gontijo:** Writing – review & editing, Supervision, Methodology. **Adriano Soares de Oliveira Bailão:** Writing – review & editing, Supervision, Methodology. **Heyde Francielle do Carmo França:** Writing – review & editing, Supervision, Methodology. **Sergio Souza Novak:** Writing – review & editing, Supervision, Software, Methodology. **Jéssica Lauanda Stirle:** Writing – review & editing, Supervision, Methodology.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This research received financial support from the Ministry of Science, Technology and Innovation (MCTI), the Brazilian Innovation Agency (FINEP), the Goiás State Research Foundation (FAPEG), the National Council for Scientific and Technological Development (CNPq), the Coordination for the Improvement of Higher Education Personnel (CAPES), the Center of Excellence in Exponential Agriculture (CEAGRE), the Federal Institute of Education, Science and Technology of Goiás (IF Goiano) — Campus Rio Verde, and the Federal Institute of Education, Science and Technology of Mato Grosso (IFMT).

Data availability

Data will be made available on request.

References

Alcaide Zaragoza, C., González Perea, R., Fernández García, I., Camacho Poyato, E., Rodríguez Díaz, J.A., 2020. Open source application for optimum irrigation and fertilization using reclaimed water in olive orchards. Comput. Electron. Agric. 173, 105407. http://dx.doi.org/10.1016/j.compag.2020.105407.

Cheng, W., Liu, Z., Li, X., 2026. Enhanced lightweight architecture for real-time detection of agricultural pests and diseases. Comput. Mater. Contin. 87 (2), http://dx.doi.org/10.32604/cmc.2025.074250, URL: https://www.sciencedirect.com/science/article/pii/S1546221826002225.

Cho, K.H., Park, J.-S., Kim, J.H., Kwon, Y.S., Lee, D.-H., 2022. Modeling the distribution of invasive species (Ambrosia spp.) using regression kriging and Maxent. Front. Ecol. Evol. Volume 10 - 2022, http://dx.doi.org/10.3389/fevo.2022.1036816.

Chung, H., Kim, J.-H., Ahn, J., Chung, Y., Kim, E., Heo, W., 2026. Resource-constrained edge AI solution for real-time pest and disease detection in chili pepper fields. Agriculture 16 (2), http://dx.doi.org/10.3390/agriculture16020223, URL: https://www.mdpi.com/2077-0472/16/2/223.

de Almeida, G.P.S., dos Santos, L.N.S., da Silva Souza, L.R., da Costa Gontijo, P., de Oliveira, R., Teixeira, M.C., De Oliveira, M., Teixeira, M.B., do Carmo França, H.F., 2024. Performance analysis of YOLO and Detectron2 models for detecting corn and soybean pests employing customized dataset. Agronomy 14 (10), http://dx.doi.org/10.3390/agronomy14102194.

Evan, A., Sarosa, M., Asmara, R.A., Kusumawardani, M., Al Riza, D.F., Azis, Y.M., 2024. Detection and counting of grape leaves using YOLOv8 via TFLite on mobile applications. In: 2024 International Conference on Electrical and Information Technology. IEIT, pp. 246–251. http://dx.doi.org/10.1109/IEIT64341.2024.10763328.

Everingham, M., Van Gool, L., Williams, C.K.I., Winn, J., Zisserman, A., 2010. The pascal visual object classes (VOC) challenge. Int. J. Comput. Vis. 88 (2), 303–338. http://dx.doi.org/10.1007/s11263-009-0275-4.

Gill, Y.S., Afzaal, H., Singh, C., Randhawa, G.S., Angrish, K., Jaura, N., Qamar, Z., Farooque, A.A., 2026. Deep learning driven edge inference for pest detection in potato crops using the AgriScout robot. Comput. Electron. Agric. 244, 111492. http://dx.doi.org/10.1016/j.compag.2026.111492, URL: https://www.sciencedirect.com/science/article/pii/S0168169926000876.

Han, F., Guan, X., Xu, M., 2024. Method of intelligent agricultural pest image recognition based on machine vision algorithm. Discov. Appl. Sci. 6 (10), 536. http://dx.doi.org/10.1007/s42452-024-06224-2.

Heusler, J., Funk, J., Wagner, A., 2025. Spatial interpolation in applied insect ecology: A review, including guidelines and datasets for practical use. J. Appl. Entomol. n/a (n/a), http://dx.doi.org/10.1111/jen.13455.

Jocher, G., Qiu, J., 2024. Ultralytics YOLO11. URL: https://github.com/ultralytics/ultralytics.

Karar, M.E., Alsunaydi, F., Albusaymi, S., Alotaibi, S., 2021. A new mobile application of agricultural pests recognition using deep learning in cloud computing system. Alex. Eng. J. 60 (5), 4423–4432. http://dx.doi.org/10.1016/j.aej.2021.03.009.

Kargar, A., Zorbas, D., Gaffney, M., O'Flynn, B., Tedesco, S., 2025. Tiny deep learning model for insect segmentation and counting on resource-constrained devices. Comput. Electron. Agric. 236, 110378. http://dx.doi.org/10.1016/j.compag.2025.110378.

Khanam, R., Hussain, M., 2024. YOLOv11: An overview of the key architectural enhancements. arXiv:2410.17725. URL: https://arxiv.org/abs/2410.17725.

Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L., 2014. Microsoft COCO: Common objects in context. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (Eds.), Computer Vision – ECCV 2014. Springer International Publishing, Cham, pp. 740–755.

Moral García, F., 2006. Analysis of the spatio–temporal distribution of helicoverpa armigera Hb. in a tomato field using a stochastic approach. Biosyst. Eng. 93 (3), 253–259. http://dx.doi.org/10.1016/j.biosystemseng.2005.12.011.

Oumaima, S., Mohamed, L., Abderrahmane, B., Iyass, M., 2025. EnviScan+: AI-driven plant identification and eco-system management for sustainable agriculture. SoftwareX 30, 102182. http://dx.doi.org/10.1016/j.softx.2025.102182.

Pereira, G.W., Valente, D.S.M., Queiroz, D.M.d., Coelho, A.L.d.F., Costa, M.M., Grift, T., 2022. Smart-Map: An open-source QGIS plugin for digital mapping using machine learning techniques and ordinary kriging. Agronomy 12 (6), http://dx.doi.org/10.3390/agronomy12061350, URL: https://www.mdpi.com/2073-4395/12/6/1350.

Pimentel, D. (Ed.), 1997. Techniques for Reducing Pesticide Use: Economic and Environmental Benefits. John Wiley & Sons, Chichester, http://dx.doi.org/10.1002/9780470510505.

Pimentel, D., 2009. Pesticides and pest control. In: Peshin, R., Dhawan, A.K. (Eds.), Integrated Pest Management: Innovation-Development Process: Volume 1. Springer Netherlands, Dordrecht, pp. 83–87. http://dx.doi.org/10.1007/978-1-4020-8992-3_3.

Qin, K., Zhang, J., Hu, Y., 2024. Identification of insect pests on soybean leaves based on SP-YOLO. Agronomy 14 (7), http://dx.doi.org/10.3390/agronomy14071586.

Sapna, Gauttam, H., Chauhan, V., Pattanaik, K., Trivedi, A., Ghosh, H., 2026. A comprehensive review of edge computing empowered smart agriculture: Trends, opportunities and future directions. Comput. Electron. Agric. 241, 111252. http://dx.doi.org/10.1016/j.compag.2025.111252, URL: https://www.sciencedirect.com/science/article/pii/S0168169925013584.

Savary, S., Ficke, A., Aubertot, J.-N., Hollier, C., 2019. The global burden of pathogens and pests on major food crops. Nat. Ecol. Evol. 3 (3), 430–439. http://dx.doi.org/10.1038/s41559-018-0793-y.

Seethalam, M., Bapatla, K.G., Kumar, M., Nisa, S., Chandra, P., Mathyam, P., Sengottaiyan, V., 2021. Characterization of Helicoverpa armigera spatial distribution in pigeonpea crop using geostatistical methods. Pest. Manag. Sci. 77 (11), 4942–4950. http://dx.doi.org/10.1002/ps.6536.

Souza, W.S.R., Alves, A.N., Borges, D.L., 2019. A deep learning model for recognition of pest insects in maize plantations. In: 2019 IEEE International Conference on Systems, Man and Cybernetics. SMC, pp. 2285–2290. http://dx.doi.org/10.1109/SMC.2019.8914428.

Tang, Y., Luo, F., Wu, P., Tan, J., Wang, L., Niu, Q., Li, H., Wang, P., 2025. An improved YOLO network for small target insects detection in tomato fields. Comput. Electron. Agric. 239, 110915. http://dx.doi.org/10.1016/j.compag.2025.110915.

Tetila, E.C., Moro, B.L., Astolfi, G., da Costa, A.B., Amorim, W.P., de Souza Belete, N.A., Pistori, H., Barbedo, J.G.A., 2024. Real-time detection of weeds by species in soybean using UAV images. Crop. Prot. 184, 106846. http://dx.doi.org/10.1016/j.cropro.2024.106846.

Tkachenko, M., Malyuk, M., Holmanyuk, A., Liubimov, N., 2020. Label studio: Open source data labeling platform. https://github.com/heartexlabs/label-studio. Acessado em 20 de Janeiro de 2025.

Verma, S., Tripathi, S., Singh, A., Ojha, M., Saxena, R.R., 2021. Insect detection and identification using YOLO algorithms on soybean crop. In: TENCON 2021 - 2021 IEEE Region 10 Conference. TENCON, pp. 272–277. http://dx.doi.org/10.1109/TENCON54134.2021.9707354.

Vilar-Andreu, M., García, L., Garcia-Sanchez, A.-J., Asorey Cacheda, R., Garcia-Haro, J., 2024. Enhancing precision agriculture pest control: A generalized deep learning approach with YOLOv8-based insect detection. IEEE Access PP, http://dx.doi.org/10.1109/ACCESS.2024.3413979, 1–1.

Wang, N., Fu, S., Rao, Q., Zhang, G., Ding, M., 2025. Insect-YOLO: A new method of crop insect detection. Comput. Electron. Agric. 232, 110085. http://dx.doi.org/10.1016/j.compag.2025.110085.

Wenkel, S., Alhazmi, K., Liiv, T., Alrshoud, S., Simon, M., 2021. Confidence score: The forgotten dimension of object detection performance evaluation. Sensors 21 (13), http://dx.doi.org/10.3390/s21134350, URL: https://www.mdpi.com/1424-8220/21/13/4350.

Xu, X., Shi, J., Chen, Y., He, Q., Liu, L., Sun, T., Ding, R., Lu, Y., Xue, C., Qiao, H., 2023. Research on machine vision and deep learning based recognition of cotton seedling aphid infestation level. Front. Plant Sci. Volume 14 - 2023, http://dx.doi.org/10.3389/fpls.2023.1200901.

Xu, W., Yang, R., Karthikeyan, R., Shi, Y., Su, Q., 2025. GBiDC-PEST: A novel lightweight model for real-time multiclass tiny pest detection and mobile platform deployment. J. Integr. Agric. 24 (7), 2749–2769. http://dx.doi.org/10.1016/j.jia.2024.12.017.

Yang, Z., Liang, W.-Z., Lawrence, N., Qiao, X., Riggan, B., Harveson, R., Chiang, C.-E., Oboamah, J., Andjawo, D.P., 2026. WeedCAM: An edge-computing camera system for multi-species weed detection in sugar beet production fields. Comput. Electron. Agric. 244, 111498. http://dx.doi.org/10.1016/j.compag.2026.111498, URL: https://www.sciencedirect.com/science/article/pii/S0168169926000931.

Yu, C., Feng, J., Zheng, Z., Guo, J., Hu, Y., 2024. A lightweight SOD-YOLOv5n model-based winter jujube detection and counting method deployed on android. Comput. Electron. Agric. 218, 108701. http://dx.doi.org/10.1016/j.compag.2024.108701.

Zhang, Z., Yang, Y., Xu, X., Liu, L., Yue, J., Ding, R., Lu, Y., Liu, J., Qiao, H., 2024. GVC-YOLO: A lightweight real-time detection method for cotton aphid-damaged leaves based on edge computing. Remote. Sens. 16 (16), http://dx.doi.org/10.3390/rs16163046.

Zhou, C., Lee, W.S., Zhang, S., Liburd, O.E., Pourreza, A., Schueller, J.K., Ampatzidis, Y., 2024. A smartphone application for site-specific pest management based on deep learning and spatial interpolation. Comput. Electron. Agric. 218, 108726. http://dx.doi.org/10.1016/j.compag.2024.108726.