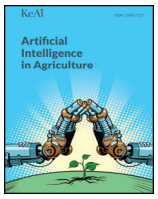
Artificial Intelligence in Agriculture 15 (2025) 377–394

Contents lists available at ScienceDirect

Artificial Intelligence in Agriculture

journal homepage: http://www.keaipublishing.com/en/journals/artificial-intelligence-in-agriculture/



Improving the performance of machine learning algorithms for detection of individual pests and beneficial insects using feature selection techniques

Rabiu Aminu [a,b,*], Samantha M. Cook [c], David Ljungberg [a], Oliver Hensel [b], Abozar Nasirahmadi [a,b]

[a] Department of Energy and Technology, Swedish University of Agricultural Sciences, Box 7032, Uppsala 75007, Sweden
[b] Department of Agricultural and Biosystems Engineering, University of Kassel, Witzenhausen D-37213, Germany
[c] Protecting Crops & Environment Division, Rothamsted Research, Harpenden, United Kingdom

ARTICLE INFO

Article history:
Received 24 November 2024
Received in revised form 27 March 2025
Accepted 28 March 2025
Available online 04 April 2025

Keywords:
Feature screening
Explainable artificial intelligence
Targeted pest control
Sustainable agriculture

ABSTRACT

To reduce damage caused by insect pests, farmers use insecticides to protect produce from crop pests. This practice leads to high synthetic chemical usage because a large portion of the applied insecticide does not reach its intended target; instead, it may affect non-target organisms and pollute the environment. One approach to mitigating this is through the selective application of insecticides to only those crop plants (or patches of plants) where the insect pests are located, avoiding non-targets and beneficials. The first step to achieve this is the identification of insects on plants and discrimination between pests and beneficial non-targets. However, detecting small-sized individual insects is challenging using image-based machine learning techniques, especially in natural field settings. This paper proposes a method based on explainable artificial intelligence feature selection and machine learning to detect pests and beneficial insects in field crops. An insect-plant dataset reflecting real field conditions was created. It comprises two pest insects—the Colorado potato beetle (CPB, *Leptinotarsa decemlineata*) and green peach aphid (*Myzus persicae*)—and the beneficial seven-spot ladybird (*Coccinella septempunctata*). The specialist herbivore CPB was imaged only on potato plants (*Solanum tuberosum*) while green peach aphids and seven-spot ladybirds were imaged on three crops: potato, faba bean (*Vicia faba*), and sugar beet (*Beta vulgaris* subsp. *vulgaris*). This increased dataset diversity, broadening the potential application of the developed method for discriminating between pests and beneficial insects in several crops. The insects were imaged in both laboratory and outdoor settings. Using the GrabCut algorithm, regions of interest in the image were identified before shape, texture, and colour features were extracted from the segmented regions. The concept of explainable artificial intelligence was adopted by incorporating permutation feature importance ranking and Shapley Additive explanations values to identify the feature set that optimized a model's performance while reducing computational complexity. The proposed explainable artificial intelligence feature selection method was compared to conventional feature selection techniques, including mutual information, chi-square coefficient, maximal information coefficient, Fisher separation criterion and variance thresholding. Results showed improved accuracy (92.62 % Random forest, 90.16 % Support vector machine, 83.61 % K-nearest neighbours, and 81.97 % Naïve Bayes) and a reduction in the number of model parameters and memory usage (7.22×10^7 Random forest, 6.23×10^3 Support vector machine, 3.64×10^4 K-nearest neighbours and 1.88×10^2 Naïve Bayes) compared to using all features. Prediction and training times were also reduced by approximately half compared to conventional feature selection techniques. This demonstrates a simple machine learning algorithm combined with an ideal feature selection methodology can achieve robust performance comparable to other methods. With feature selection, model performance can be maximized and hardware requirements reduced, which are essential for real-world applications with resource constraints. This research offers a reliable approach towards automatic detection and discrimination of pest and beneficial insects which will facilitate the development of alternative pest control approaches and other targeted pest removal methods that are less harmful to the environment than the broad-scale application of synthetic insecticides.
© 2025 The Authors. Publishing services by Elsevier B.V. on behalf of KeAi Communications Co., Ltd. This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/).

* Corresponding author at: Department of Agricultural and Biosystems Engineering, University of Kassel, Witzenhausen D-37213, Germany.
E-mail address: rabiu.aminu@uni-kassel.de (R. Aminu).

https://doi.org/10.1016/j.aiia.2025.03.008
2589-7217/© 2025 The Authors. Publishing services by Elsevier B.V. on behalf of KeAi Communications Co., Ltd. This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/).

1. Introduction

The Food and Agriculture Organization (FAO, 2021) estimates that of the total $220 billion in annual economic losses attributed to plant diseases, at least 31.8 % are caused by insect pests, including invasive species. The Colorado potato beetle (CPB, *Leptinotarsa decemlineata*) and green peach aphids (*Myzus persicae*) are two of the most damaging insect pests of potato crops due to direct feeding damage and disease transmission, respectively (Gao et al., 2024a). Larvae of the CPB can cause up to 100 % defoliation of the leaves and infestation may reduce the yield of potato tubers by more than 50 % (Sablon et al., 2013; Balaško et al., 2020; Bitkov and Lykov, 2024). Green peach aphids (*Myzus persicae*) cause even more damage; in Europe, viruses, particularly potato virus Y (PVY) transmitted by aphid feeding activity cause total annual losses of 180 million Euros (Dupuis et al., 2023). Environmentally sustainable solutions for insect pest control, particularly in arable crops, are therefore crucial for protecting yield and ensuring food security. However, current pest control methods, which usually involve the detection of pests in the field via scouting followed by the application of synthetic insecticides as blanket treatments across fields, negatively impact the environment, ecosystem, and biodiversity (Richard, 2010; Chaudhary et al., 2021; Beaumelle et al., 2023). Additionally, manual detection of insect pests via plant scouting is labour-intensive and time-consuming (Chen et al., 2021). Thus, there has been growing interest in automated insect detection and classification in recent years (Xia et al., 2018; Kirkeby et al., 2021; Hasan et al., 2024; Suresh et al., 2025), which could help detect pest presence in the field quickly, easily and accurately, and enable the determination of their spatial distribution for more targeted applications (Bick et al., 2024).

Advances in computer vision and machine learning have enabled researchers to develop object detection systems capable of detecting small objects with high accuracy (Don et al., 2023; Tan et al., 2023; Wan et al., 2023). These developments have made innovative processes such as selective weed control a reality and are now finding other practical applications in agriculture, for example, in plant disease detection and classification (Bhosale et al., 2023; Kini et al., 2023; Singh et al., 2023). These technologies have also been explored for insect detection and classification, where digital image processing techniques have demonstrated great potential (Espinoza et al., 2016). Improvements in high-resolution imaging technology have increased the capability of image processing techniques to extract detailed features from high-resolution images for accurate insect segmentation (Alkan and Aydın, 2023). Li et al. (2015) used multifractal analysis to segment whitefly images based on local singularity and global image characteristics, demonstrating superior performance over traditional image thresholding methods, achieving a true-positive detection rate of 86.9 % and a false-positive rate of 8.2 %. Using a different approach, Xia et al. (2015) used watershed segmentation and Mahalanobis distance to identify common greenhouse pests on sticky traps; their method achieved a coefficient of determination of 0.945 for aphid counting. In contrast, Agrawal et al. (2018) used clustering and pseudo-colour image processing for pest region detection, including CPB in rice cultivation. Kasinathan et al. (2020) combined foreground segmentation and a GrabCut algorithm to segment different insect species in popular public datasets and achieved impressive results.

Image processing techniques are often considered the pre-processing steps for targeted object detection and semantic segmentation in deep learning (Wang et al., 2019; Miranda et al., 2023; Sahin et al., 2023). While these techniques are computationally efficient and have excelled in simple image segmentation tasks such as distinguishing objects against simple backgrounds, their performance deteriorates with increasing lighting variations and object background complexity. This inconsistency is demonstrated by their excellent performance with certain images and poor performance in others, and this does not improve with additional data, as no supervised learning is involved (Cserni and Rovid, 2023; He et al., 2023). Given the unpredictable nature of the real field, these techniques cannot be relied upon for practical applications (Cuevas et al., 2023).

Machine learning has proved more effective than traditional image processing for insect detection and classification. The existing works based on machine learning can be broadly grouped into classical and deep learning. The concept of classical machine learning is to design hand-crafted features based on object descriptors such as shape, texture, colour, edges, and other relevant features. These features are either extracted from the whole image or the region of interest (ROI) in the segmented image and subsequently fed into the machine learning classifier for training and evaluation (Gao et al., 2024a). Liu et al. (2016) used histogram of oriented gradients (HOG) features extracted from positive and negative sample images to train a support vector machine (SVM) classifier, then employed a maximally stable extremal region descriptor to detect aphids in the classified images, achieving identification and error rates of 86.81 % and 8.91 %, respectively. In a similar approach, Kasinathan et al. (2020) identified 9 and 24 insect classes in the Wang dataset and 9 and 24 insect classes in the Xie dataset by extracting the shape features and training machine learning techniques such as artificial neural networks (ANN), SVM, k-nearest neighbours (KNN), Naïve Bayes (NB), Random forest (RF) and CNN models, achieving the highest classification rate of 91.5 %. Kasinathan and Uyyala (2021), combined different feature descriptors to train traditional and ensemble classifiers for insect classification. They experimented with various feature combinations and improved the classification accuracy by 2.6 % using majority voting. To count rice planthoppers (*Sogatella furcifera*) in rice paddy fields, Yao et al. (2014) proposed a three-layer detection method: the first layer used an AdaBoost classifier based on Haar features, an ensemble machine learning method that combines multiple decision trees into a stronger classifier. The second layer used an SVM classifier based on HOG features, and the third layer applied a threshold judgment of the three features. Their experimental results show an 85.2 % detection rate and a 9.6 % false detection rate. Others used a different approach; first applying image segmentation to extract ROI in the image before extracting features for training machine learning classifiers. For example, Lucero et al. (2015) detected CPB with an 85 % recognition rate by using a contour orientation histogram to extract features from the ROI and fed them as inputs to the random subspace classifier. Similarly, Remboski et al. (2018) developed an insect classification system by extracting ROIs and transforming them into feature vectors using a bag-of-words model. For the classification, they trained SVM, KNN, Decision Tree, and Gaussian Naïve Bayes, with SVM achieving the highest accuracy of 86.38 %. In contrast, deep learning uses convolutional layers to automatically extract relevant features of the target object and fully connected layers for classification. With rapid developments in computing power, imaging equipment, and its superiority for instance-segmentation, deep learning has become popular for tasks such as aphid detection and counting (Xu et al., 2023; Gao et al., 2024a), two-spotted spider mite (*Tetranychus urticae*) detection, (Zhou et al., 2024), fall armyworm *(Spodoptera frugiperda)* detection in field crops (Kasinathan and Uyyala, 2023) and detection of brown marmorated stink bug (*Halyomorpha halys*) (Betti Sorbelli et al., 2023).

Methods based on digital image processing are ideal for insect detection in simple scenarios but fail in complex situations because they are sensitive to lighting, colour, and other variations in the object background (Gao et al., 2024a). In contrast, methods based on deep learning may achieve high accuracy. However, these depend on large amounts of data and huge computational resources, making them computationally expensive and difficult to implement on affordable hardware for real-world applications (Cserni and Rovid, 2023). Due to the rapid advancements in deep learning, researchers have not fully explored the potential of classical machine learning. Despite dependence on manual feature design, classical machine learning techniques have the potential to achieve acceptable insect detection and classification accuracy (Gao et al., 2024b) with a small amount of data, a short training time, and less computational power compared to deep learning methodologies.

Owing to their less computational resource requirements, they can easily be implemented with low-cost hardware and deployed for practical applications. Using an optimal feature set, these methods can achieve results comparable to deep learning methods while maintaining lower computational complexity. However, no single feature is suitable for all tasks and challenges remain in identifying the feature subset that optimizes model performance while reducing complexity (Ye et al., 2023). To our knowledge, no study has yet assessed the impact of feature selection techniques on colour, shape, texture, and HOG features on a model's performance in insect detection tasks. This gap is significant because including irrelevant features can result in multidimensionality and redundancy among features (Sumesh et al., 2021), which increases computational time and reduces the model's generalization ability and classification accuracy (El-Kenawy et al., 2024). Explainable artificial intelligence (XAI) offers an alternative approach to addressing this. XAI refers to machine learning models that are transparent and easily interpretable, which is essential for building confidence, acceptability and trust, especially in settings where understanding the reasoning behind a model's decision-making process holds equal importance to its prediction accuracy (Dave et al., 2020; Zhang et al., 2021). This approach has also been used to interpret machine learning models for agricultural data analysis (Ryo, 2022). The XAI incorporate the concept of permutation feature importance (PFI) to assess the impact of permuting each feature on the model's prediction outcomes and Shapley Additive explanations (SHAP) values to explain the contributions of individual features to a model's prediction results (Lundberg et al., 2020). Therefore, a method based on XAI to determine the optimal combination of features for training machine learning algorithms, such as SVM, RF, KNN, and NB, was proposed for the current study. Unlike conventional feature selection methods, this approach integrates PFI ranking and SHAP values to identify the most relevant features in the dataset, thereby reducing redundancy, optimizing performance and decreasing computational overload associated with machine learning. The proposed method was evaluated and compared to conventional feature selection techniques. This method was designed for efficient implementation on low-cost hardware to detect harmful insects (CPB and aphids) and beneficial insects (ladybirds) in arable crops. The main objective was to balance computational efficiency with detection accuracy while limiting computational complexity through feature selection.

2. Materials and methods

2.1. Data collection and pre-processing

Two insect pests (the Colorado potato beetle (CPB, *Leptinotarsa decemlineata*) and green peach aphids, *Myzus persicae*), one beneficial insect (seven-spot ladybird, *Coccinella septempunctata*), and three crops (potato, *Solanum tuberosum*; faba bean, *Vicia faba;* and sugar beet, *Beta vulgaris* subsp. *vulgaris*) were selected for this study. The crop plants were grown in the greenhouse of the University of Kassel, Germany, while the insects were collected from three experimental farms, all located in the Kassel region. Four stages of CPB and seven-spot ladybird (eggs, larva, pupa and adults) and adult aphids were collected from the focus crops and kept in a greenhouse, with each species maintained separately inside 33 × 33 × 5 cm; 570 g ventilated plant insect net cages. CPBs were maintained on potato plants, and aphids on the three species; ladybirds were supplied with aphids as food. All insects were provided with water using moistened filter paper. The greenhouse temperature ranged between 14 and 31 °C and insects were maintained for 1–3 weeks before being replaced with freshly collected individuals. Data collection occurred from April to July, with insects being transferred from the greenhouse maintenance cages to our laboratory setup for image acquisition.

To create a working dataset reflecting the real field conditions, images of insects on the crop plants were collected from an imaging tent in the laboratory and from fields on three commercial farms. The imaging tent (120 × 120 × 200 cm) was made from black Mylar fabric, and was artificially lit using light-emitting diodes (LED-LE1200-E03L-1S) with a spectrum of 450 nm +470 nm, 660 nm, 730 nm, 6500 K, providing various illumination levels and lighting intensities. In each imaging session, 3–5 aphid-infested plants were transferred to the imaging tent, and individual CPBs and ladybirds were placed by hand on specific areas of the plants to simulate natural infestations. Approximately, 300 individual CPBs, 250 ladybirds and the same number of aphid colonies were used. CPB was imaged only on potato plants because it does not infest faba bean and sugar beet. Individual insects on plants were imaged more than once; however, to ensure data diversity, they were regularly replaced. This approach combined with images of insects on crop plants captured from real farm fields improved the quality and diversity of our dataset. Images of CPB on potato, and aphids and ladybirds on potato, faba bean and sugar beet crops were collected at three different times of the day (morning, afternoon, and evening). A total of 2000 images were captured, with 1000 images taken in laboratory settings and 1000 in the field. These include 500 images of individual CPB on potato plants, 500 seven-spot ladybirds, 500 aphids on a mixture of the three crop plants, and 500 leaves without insects across three crops. For ladybirds and aphids, images were evenly distributed across the three crop types, with approximately 167 samples per crop per insect. Two cameras were used for data acquisition: The Canon EOS 2000D and Sony α6400, collecting 1000 images each. This dual-camera approach enhanced dataset diversity, ensuring reliability for model training, evaluation and generalizaation. Samples of insect plant images in our dataset are shown in Fig. 1.

The original image resolution of 6000 × 4000 pixels acquired from the two cameras was converted to a lower resolution of 800 × 500 pixels because training models with high-resolution images is time-consuming and requires substantial computing power. Unwanted image backgrounds were cropped using the crop-and-select, method adopted from (Gao et al., 2024a). This involves manually dividing the original image into 4 smaller parts and carefully selecting the portion containing the insects and background leaf while discarding portions that do not have insects or leaves. The final insect-plant dataset has 1000 images: 250 CPB on potato plants, 250 for both ladybirds and aphids, divided approximately equally between the three crop plants, and 250 of the three crop plant leaves without insects, equally distributed across crop types. Images of plant leaves without insects were included because the model's ability to detect the absence of insects in the image holds equal importance to a farmer as positive detection of insects, as this means no treatment is needed. The complete workflow of the insect-plant dataset pre-processing and labelling is given in Fig. 2. The labelImg annotation tool, an open-source software designed for image labelling, was used to annotate the images. This enabled the labelling of insects by drawing rectangular bounding boxes around them and saving the annotations in extensible markup language file format (Fig. 3). The images and labels are inputs to the machine learning algorithms.

Our final dataset combined images collected in real field conditions with laboratory-captured images under varied lighting conditions to enhance generalization. Field data captured at different times of the day account for natural field variations, while laboratory data acquired under varied lighting wavelengths simulate diverse environmental conditions. This ensures that the proposed method generalizes well across different real-world scenarios, mitigating the impact of lighting variations on detection performance.

2.2. Overview of the framework

This study combined image segmentation, feature selection, and classical machine learning to detect and identify pests and beneficial insects. In the first step, a GrabCut algorithm was developed in Python to segment the potential ROI in the image, which are areas likely to contain the target insects. In the second step, features such as shape descriptors

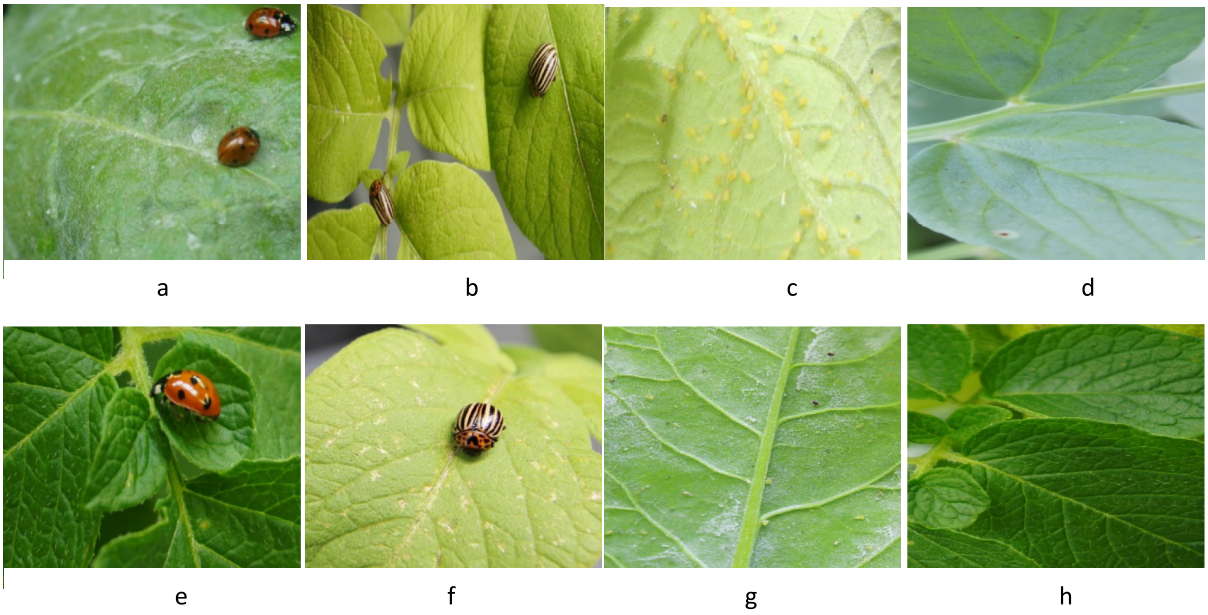


Fig. 1. Sample images from our insect plant dataset- a and e: Seven-spot ladybird (*Coccinella septempunctata*) on sugar beet (*Beta vulgaris* subsp. *vulgaris*) and potato (*Solanum tuberosum*) plants, respectively; b and f: Colorado potato beetle (*Leptinotarsa decemlineata*) on potato plant; c and g: green peach Aphids (*Myzus persicae*) on potato and sugar beet plants; d and h: faba bean (*Vicia faba*) and potato plant leaves without insects. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

(e.g. area, compactness, elongation and aspect ratio), texture features (e.g. correlation, entropy and contrast), colour features (e.g. skewness, energy and standard deviation derived from colour histogram) and multiple HOG features are extracted from the segmented ROIs for feature selection. These handcrafted features capture critical information about texture patterns, colour distributions and morphology in the image, which are key factors for discriminating objects. Incorporating feature selection in our methodology helps improve model performance by retaining the most relevant features and reducing computational time by removing redundant features in our image dataset. In step three, the performance of seven feature selection techniques based on standard performance metrics and model computational complexity was evaluated. To improve the feature selection process, an optimized feature selection method was developed using XAI feature importance ranking to identify top-performing features. In the final step, four machine learning algorithms (SVM, RF, KNN and NB) were trained using our new feature selection method, and then a performance comparison was conducted.

An overview of our framework is shown in Fig. 4; the algorithm first inputs the image into the GrabCut algorithm to segment potential insect regions, which are areas likely to contain insects in the image. It then extracts all relevant features from segmented insects and stores them as a feature vector. Our optimized feature selection technique identifies the optimal features from the created feature vector and feeds them to the machine learning algorithm for training. The model identifies if the leaf has insects before classifying the insects into three classes (Ladybirds, CPB, and aphids). If the classification result is a ladybird, it is considered a beneficial insect. On the contrary, if the classification result is CPB or aphids, the identified insect is pest-CPB or pest-aphids. Lastly, if the leaf has no insect the model detects that at the initial classification stage. The steps to determine the optimal features are explained in detail in Section 2.2.5.

2.2.1. Region of interest segmentation

Image segmentation is a fundamental step in computer vision and machine learning tasks, essential for partitioning an image into distinct regions or scenes based on pixel similarities. In this work, the GrabCut algorithm was used to segment insects from the backgrounds in the dataset. The GrabCut algorithm was originally introduced by Rother et al. (2004) and remains the most widely used lightweight image segmentation method because of its computational efficiency and ability to produce results comparable to deep-learning-based approaches without requiring extensive computational resources (Zhang et al., 2017; Liang and Palaoag, 2024). Since this study aims to introduce a novel method that balances computational efficiency with practical applicability, the GrabCut algorithm was incorporated into the machine learning pipeline due to its lightweight processing advantages over deep learning-based segmentation, which typically requires specialized

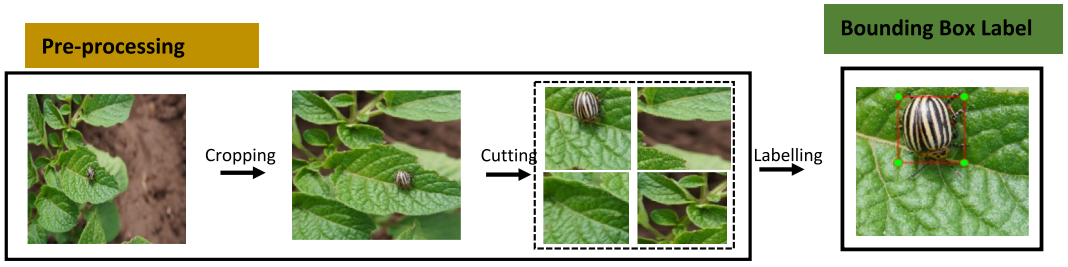


Fig. 2. Workflow of image pre-processing and labelling for an insect-plant dataset- The right end shows the zoom-out labelled Colorado potato beetle image with the insect inside the bounding box. Image pre-processing is an important step in machine learning.

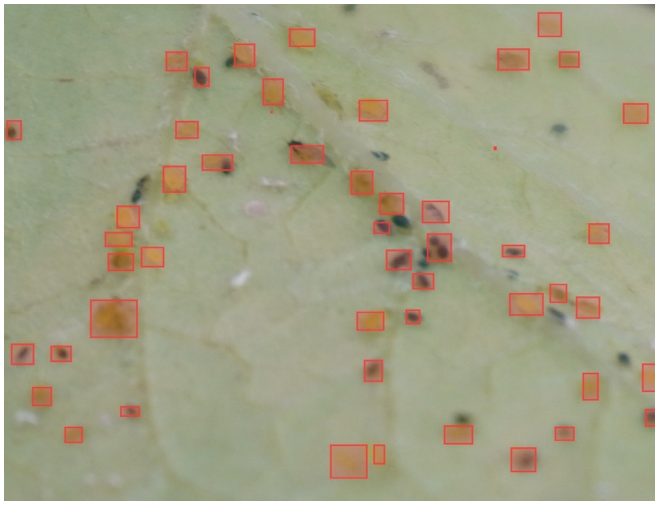


Fig. 3. Images of aphids on a potato leaf- with manually annotated bounding boxes (red rectangles). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

hardware and extensive training. The GrabCut image segmentation process consists of three key steps:

(a) Bounding box initialization:

- A bounding box is manually or automatically placed around the region of interest (ROI), defining the area containing the object to be segmented in the image.
- Everything outside the bounding box is initially considered the background.

(b) Gaussian Mixture Models (GMM) clustering:

- The GMM clusters pixel values into foreground and background distributions based on probability models. Initially, the image is classified using a trimap model (T) with three-pixel groups:

i. TB (Background pixels): pixels outside the bounding box, initially labelled as background.
ii. TU (Unknown pixels): pixels within the bounding box, requiring further classification.
iii. TF (Foreground pixels): initially empty ($TF = \varnothing$), representing the object region to be segmented.

- Alpha values (αn) are introduced to optimize the classification into the background ($\alpha n = 0$) and foreground ($\alpha n = 1$) GMM models.

(c) Iterative segmentation and energy minimization:

- The segmentation is refined iteratively using an energy minimization framework, optimizing the Gibbs energy function described in Eq. (1)

$$E(Gibbs) = U(data) + V(smooth) \quad (1)$$

Where the data term (U(data)) measures how well the segmentation fits the observed image data and the smoothness term (V(smooth)) identifies unexpected changes in segmentation labels across neighbouring pixels. The GrabCut algorithm being an iterative method has advantages over traditional image segmentation algorithms, such as thresholding methods, template matching and watershed segmentation, particularly in refining object boundaries and handling complex image backgrounds.

2.2.2. Feature extraction

Features are unique characteristics or attributes that discriminate different objects in the same image or environment. Machine learning algorithms are generally trained on features and labels to predict output; their performance depends on how well features are extracted and prepared before training. Therefore, feature extraction is a critical step in data pre-processing and forms the basis of machine learning training. The goal is to extract relevant information from the data, suitable for the classifier to distinguish between the different classes in the dataset. Features previously used in the literature and proven effective for insect classification tasks were considered in this study. The histogram of oriented gradient (HOG) features has achieved excellent results in extracting shape and edge features for insect pest classification and identification (Kasinathan and Uyyala, 2021). The extraction of HOG feature vectors from our image data includes dividing the image into cells, histogram generation, gradient calculation, and block normalization (Eqs. (2), (3) and (4)).

$$G_x(x,y) = H(x,y+1) - H(x,y-1) \quad (2)$$

$$G(x,y) = \sqrt{G_x(x,y)^2 + G_y(x,y)^2} \quad (3)$$

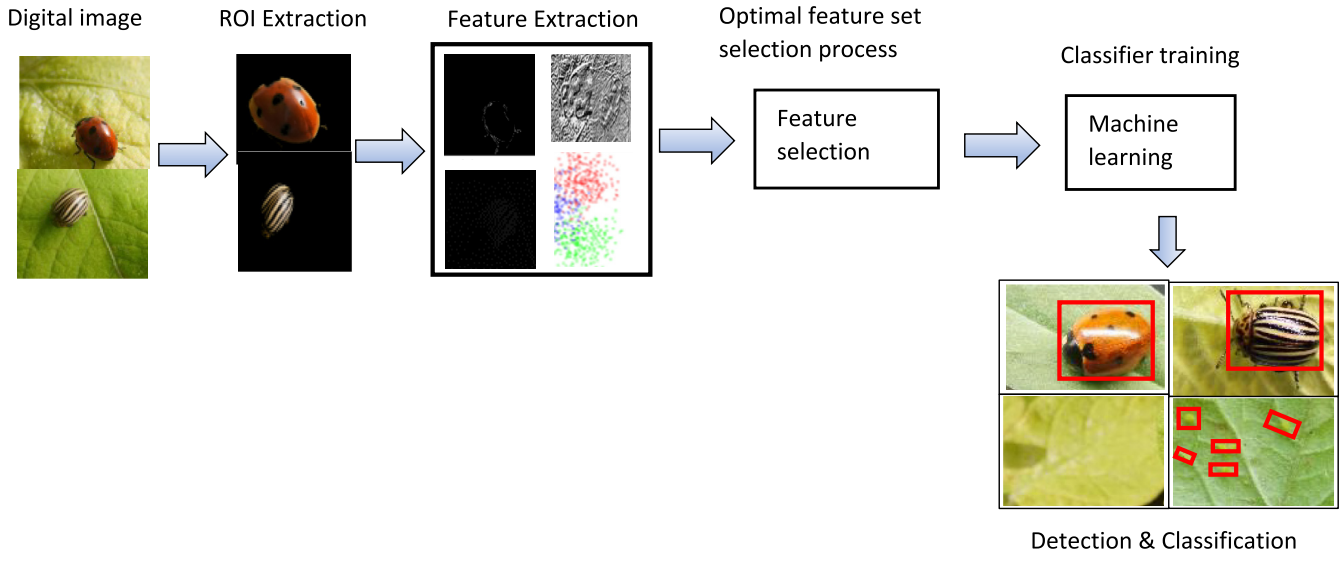


Fig. 4. Framework for the improved feature selection technique- extracted region of interest (ROI), shape, texture, edges and colour features.

$$\alpha(x,y) = tan^{-1}\left(\frac{G_y(x,y)}{G_x(x,y)}\right) \quad (4)$$

Where x, y are pixel coordinates and $H(x,y)$ are pixel values at this location and $G(x,y)$ represents the gradient magnitude. $G_y(x,y)$ and $G_x(x,y)$ are the vertical and horizontal gradients, respectively, and $\alpha(x,y)$ denotes the gradient orientation.

The next important feature for insect classification is the texture feature. Texture describes patterns, pixel arrangements and spatial distributions of tones within specific images or bands in satellite images. This important pictorial information about the structural arrangement of image surfaces and their relationships with neighbouring pixels can help to discriminate objects or regions of interest in digital, aerial and satellite images (Haralick et al., 1973). Statistical texture features such as contrast, correlation, entropy, variance and angular moment, calculated in the spatial domain were used in this study. These descriptors can effectively quantify texture information in digital and aerial photos. Alkan and Aydın (2023) found Grey level co-occurrence matrix (GLCM) to be effective for extracting statistical texture features from digital and unmanned aerial vehicle images, respectively. GLCM extracts texture features by using statistical measures to analyze the spatial relationship of pixels in an image. The first four basic and eight advanced GLCM components that quantify texture features in our data were extracted, and the corresponding mathematical equations of these descriptors, as described in Haralick et al. (1973) are as follows:

$$Angular\ sec.moment = \sum_i \sum_j \{p(i,j)\}^2 \quad (5)$$

$$Contrast = \sum_{n=0}^{N_g} n^2 \left\{ \sum_{\substack{i=1 \\ |i-j=n|}}^{N_g} \sum_{j=1}^{N_g} p(i,j) \right\} \quad (6)$$

$$Correlation = -\sum_i \sum_j p(i,j)\log(p(i,j)) \quad (7)$$

$$Entropy = \frac{\sum_i \sum_j (ij)p(i,j) - \mu_x\mu_y}{\sigma_x\sigma_y} \quad (8)$$

$$Variance = \sum_i \sum_j (1-\mu)^2 p(i,j) \quad (9)$$

$$Sum\ average = \sum_{i=2}^{2N_g} i p_{x+y}(i) \quad (10)$$

$$Sum\ variance = \sum_{i=2}^{2N_g} (i - sum\ entropy)^2 p_{x+y}(i) \quad (11)$$

$$Sum\ Entropy = \sum_{i=2}^{2N_g} p_{x+y}(i)\log(p(i,j)) \quad (12)$$

$$Difference\ of\ Variance = Variance\ of\ p_{x+y^*} \quad (13)$$

$$Difference\ of\ Entropy = \sum_{i=0}^{N_g-1} p_{x-y}(i)\log\left\{p_{x-y}(i)\right\} \quad (14)$$

$$Max.correletion\ coeff. = (2nd\ largest\ eigen\ value\ of\ Q)^{1/2} \quad (15)$$

$$Where\ Q = \sum_k \frac{p(i,j)p(j,k)}{p_x(j)p_y(k)}$$

$$Information\ Measure\ Correlation = \frac{HXY - HXY1}{max\{HX, HY\}} \quad (16)$$

Note: i,j is pixel location and $p(i,j)$ is pixel value at this location and N_g represents number of grey levels in the image. $\mu_x, \mu_y, \sigma_x, and\ \sigma_y$ are the means and standard deviations at p_x *and* $p_{y,}$ and HX, HY are entropies of $p_x(i)$and $p_y(i)$, respectively.

After the GLCM computation, the next feature extractor is a colour histogram, commonly used for computing colour features. For an RGB image, the histogram is the plot of intensity values of each colour component against the frequency of pixels at that value, with the bins providing information about the colour distribution in the image, which might correspond to different objects or regions in the image. For example, a narrow bin indicates a low contrast image while a wider bin signifies a high contrast image. Colour feature has been used previously for insect classification (Espinoza et al., 2016; Kasinathan and Uyyala, 2021). From the GrabCut segmented images, the mean, standard deviation, skew, energy and entropy statistics-based histogram features were extracted. These statistics-based features give a better estimation of colour features, especially for greyscale or segmented images. They provide information about the general intensity distribution in the image and help in extracting important characteristics such as brightness, contrast, asymmetry, energy levels and data quality from the image. Considering the probability distribution $P(g)$, grey level pixel locations g and total number of intensity levels in image L, Sergyán (2008) summarizes the five statistics-based colour feature descriptors as follows:

$$\overline{g} = \sum_{g=0}^{L-1} gP(g) \quad (17)$$

$$\sigma_g = \sqrt{\sum_{g=0}^{L-1} (g-\overline{g})^2 P(g)} \quad (18)$$

$$\alpha_3 = \frac{1}{\sigma_g^3} \sum_{g=0}^{L-1} (g-\overline{g})^3 P(g) \quad (19)$$

$$Energy = \sum_{g=0}^{L-1} [P(g)]^2 \quad (20)$$

$$Entropy = \sum_{g=0}^{L-1} P(g)\log_2[P(g)] \quad (21)$$

Where $\overline{g}$, σ_g and α_3 are mean, standard deviation and skew, respectively.

The next widely used feature in the literature for object detection is the shape feature, which has demonstrated a high success rate for weed detection and insect classification (Kasinathan et al., 2020; Pathak et al., 2023). Geometry-based shape information such as area, circularity, elongation and many advanced shape descriptors, can distinguish objects in digital images. For example, in digital and aerial images area can help discriminate small objects from large ones while circularity can distinguish round objects from irregular ones. Contour detection is one of the most efficient methods for extracting geometric shape features from digital images (Figueiredo et al., 2016). However, basic shape features such as area, perimeter, minor and major axis depend on image dimensions and their values change with image resolution. Due to this limitation, these parameters are ineffective for image classification but can be used to derive advanced features independent of image dimension changes (Pathak et al., 2023). The contour method was used to compute 21 advanced geometric shape descriptors applied by Pathak et al. (2023) in their work on weed detection (Table 1).

2.2.3. Techniques for feature selection

Selecting suitable features is crucial for optimizing the performance of machine learning classifiers and reducing computational complexity.

Often, most extracted features do not contribute to model improvement. Therefore, exploratory data analysis was applied to thoroughly analyze all features to uncover hidden patterns, relationships, anomalies and redundancy by checking their statistical summaries, and univariate, bivariate, and multivariate interactions. The correlation matrix shows the relationship and strength of features with target classes in our dataset "CPB, ladybird and aphids" (Fig. 5). A similar plot for GLCM features highlighted that difference in variance and difference in entropy features are outliers and therefore dropped. No issue was observed for colour histogram and HOG features and thus all were included in the feature selection process.

Feature selection aims to automatically identify the most relevant features (Sosa-Cabrera et al., 2023). However, these features are not easily identifiable as no single feature selection method is universally applicable (da Silva et al., 2015). Therefore, feature selection techniques were used to sift through many features and find the optimal set that maximizes the classification rate. Although several feature selection methods exist, filter methods are model-independent and can identify relevant features based on intrinsic data properties without overfitting, making them ideal for our image datasets. In this work, filter-based feature selection techniques were experimented with to ensure fair comparison and reliability in our analysis.

2.2.3.1. Mutual information (MI). The MI technique measures the relevance and redundancy among features and estimates how much information one random variable has about another. This is valuable in feature selection as it measures how relevant a feature subset is to the target output (Estévez et al., 2009; Vergara and Estévez, 2014). If X and Y are two discrete random variables with joint probability mass function $p(x,y)$ and marginal probabilities $p(x)$ and $p(y)$, their MI can be expressed as follows:

$$I(X;Y) = \sum_{x \in X} \sum_{y \in Y} p(x,y)\, log \frac{p(x,y)}{p(x)p(y)} \tag{22}$$

The MI becomes zero when the two variables are statistically independent, which means $p(x,y) = p(x)p(y)$

2.2.3.2. Chi-square coefficient. The Chi-squared (X^2) coefficient is a statistical feature selection method that measures the relevance between a feature t and category C. If the feature t and category C are independent, t cannot be used to determine whether the object belongs to category C (Zhai et al., 2018). The Chi-squared coefficient was used to estimate the relevance of all our extracted features for the target output. The statistical expression for X^2 is defined:

$$X^2 = \sum \frac{(Observed\ frequency - Expected\ frequency)^2}{Expected\ Frequency} \tag{23}$$

2.2.3.3. Fisher separation criterion. When used for feature selection, the Fisher separation criterion identifies the best combinations of features that group similar objects in the same class while maximizing the margin between objects in different classes. da Silva et al. (2015) summarized Fisher separation criterion equations as:

$$F^d = \underset{F}{argmax} J\left(F^d\right) \tag{25}$$

Where, F is the feature set containing all the extracted features, F^d is the feature subset consisting of d features and J is found by finding the

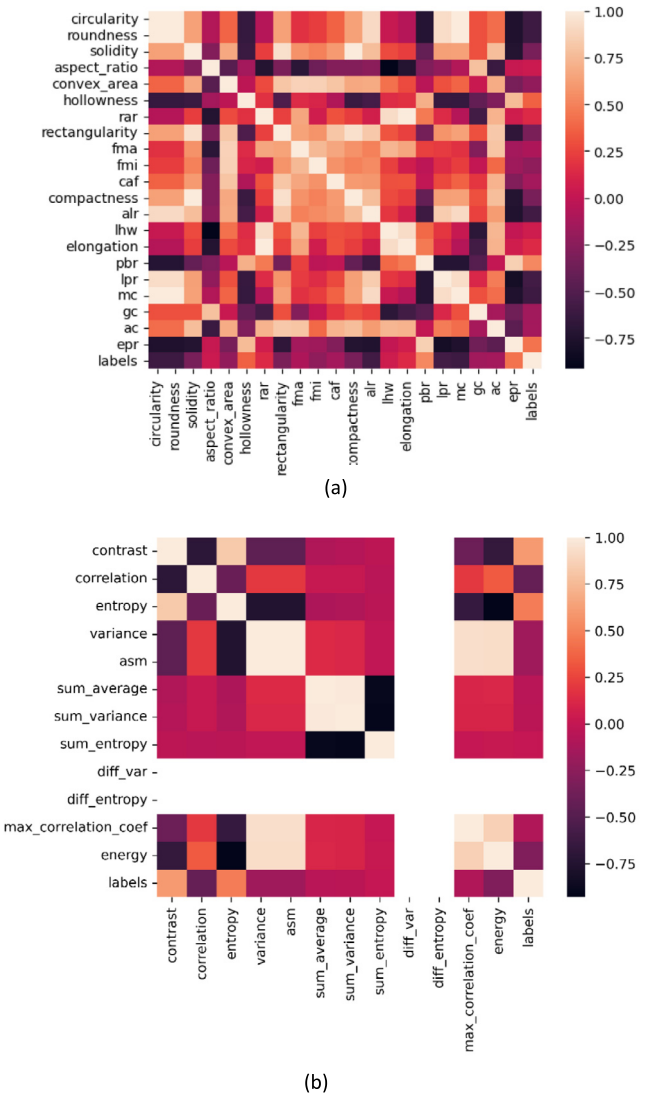


Fig. 5. Correlation matrix of extracted sample features with three labelled insect classes-showing relationships among different (a) shape features and (b) texture features and their correlation with labelled insect classes. Positively correlated features like circularity, roundness and solidarity show potential for discriminating between insect classes. While negatively correlated features such as aspect ratio, and elongation may contribute less to improving classification performance.

Table 1
Geometric shape feature descriptors used for image analysis and object detection tasks. These features quantify various aspects of the object's geometry and are essential for accurate object detection and classification.

Shape feature descriptors	Shape feature descriptors
$Convex\ area\ ferret\ ratio = \frac{Convex\ area}{Ferret\ diameter^2}$	$Circularity = \frac{4\pi \times Area}{Perimeter^2}$
$Log\ height\ width\ ratio = log\left[\frac{Height}{Width}\right]$	$Aspect\ ratio = \frac{Major\ axis}{minor\ axis}$
$Elongation = \frac{Major\ axis\ -\ Minor\ axis}{Major\ axis + Minor\ axis}$	$Roundnes = \frac{4 \times Area}{\pi \times Major\ axis^2}$
$Perimeter\ broadness\ ratio = \frac{perimeter}{2 \times \pi(Width + Height}$	$Solidity = \frac{Area}{Convex\ area}$
$Length\ perimter\ ratio = \frac{Major\ axis}{perimeter}$	$Convex\ area = \frac{Area}{Solidity}$
$Modified\ circularity = \frac{4 \times area}{perimeter \times ferret\ diameter}$	$Conpactness = \frac{Area}{Ferret\ diameter}$
$Grum\ circularity = \frac{16 \times Area^2}{4\pi \times perimter \times ferret^3}$	$Area\ lenght\ ratio = \frac{Area}{Major\ axis^2}$
$Rectangularity = \frac{Area}{Bounding\ rectangle\ area}$	$Hollowenes = \frac{Convex\ area\ -\ Area}{Convex\ area}$
$Ferret\ major\ axis\ ratio = \frac{Ferret\ diameter}{Major\ axis}$	$Area\ compact = \frac{4 \times Area}{\pi Ferret\ diameter^2}$
$Ferret\ minor\ axis\ ratio = \frac{Ferret\ diameter}{Minor\ axis}$	$Inverse\ aspect\ ratio = \frac{1}{aspect\ ratio}$
$Equivalent\ perimeter\ ratio = \frac{perimeter}{Equivalent\ perimeter}$	*where,*
	$Equivalent\ perimeter = 2.0 \times \sqrt{\pi \times Area}$

Note: the dimensions for calculating area and perimeter are measured in pixel^2 and pixels, respectively.

value of projective vector W that maximizes the following Fisher criterion (Lei et al., 2012):

$$J = \frac{\left|W^T S_b W\right|}{\left|W^T S_W W\right|} \tag{26}$$

Where W^T, denotes the transpose of W, S_b, is the between-class scatter matrix and $S_{\boldsymbol{w}}$ is the within-class scatter matrix. The optimum value of W is obtained by solving the Eigenvalues problem. The values of W^T determine the features that contribute most to class separation, thus achieving high scores in the Fisher criterion.

2.2.3.4. Variance thresholding. This feature selection method uses a threshold to remove all features with low variance in the feature vector. It assumes that features with lower variance are less informative while features with higher variance might have more useful information about the target object. The variance of all the features is calculated, and a threshold is set. The selection of this threshold is critical; a high value might lead to elimination of the relevant features and a low threshold might result in including redundant features.

2.2.3.5. Maximal information coefficient (MIC). The MIC, introduced by Reshef et al. (2011), is a dependency measure based on information theory. It is an important tool for identifying non-linear relationships between pairs of variables in large datasets. MIC can identify both correlated and non-correlated relationships and for correlated relationships provides a score similar to the coefficient of determination (R^2) obtained from fitting data to the regression line. The basic steps in computing MIC are: (I) For every pair of coordinates (x, y), the algorithm identifies the $x \times y$ grid with the highest induced MI; (II) The algorithm normalizes the MI scores and compiles a matrix that stores, for each resolution, the best grid at that resolution and its normalized score; (III) The normalized scores form the characteristic matrix, which can be visualized as a surface; MIC corresponds to the highest point on this surface.

2.2.3.6. Principal components analysis (PCA). Although not a feature selection technique, PCA is a widely used statistical method in machine learning for dimensionality reduction. PCA reduces the dimensionality of the feature vector, containing all the extracted features to a lower dimensional vector thereby reducing computational complexity and improving performance. The new feature set, called principle components is the linear combination of the original features in the dataset. However, PCA suffers from major limitations including information loss due to data compression. Additionally, it is computationally intensive, because it transforms all the features including irrelevant ones into new features (Xiao, 2024).

2.2.3.7. Proposed explainable artificial intelligence (XAI) feature selection method. Despite decades of research on feature selection, challenges remain in identifying the optimal feature subset that optimizes model performance while reducing complexity. Some methods require massive computational resources while others struggle to distinguish between important and redundant features, especially for high-dimensional data such as images. Therefore, feature selection remains an active research area in machine learning and data mining (Sosa-Cabrera et al., 2023). This work proposes using XAI to identify the most important and contributing features for model development. The concept of permutation feature importance (PFI) (Fisher et al., 2019) and Shapley Additive explanations (SHAP) (Lundberg et al., 2017) were adopted to identify the most relevant features in our dataset. The PFI evaluates the effect of permuting a feature on the model's predictive ability. Let $\hat{f} : \mathbb{R}^p \longrightarrow \mathbb{R}$, denotes a machine learning hypothesis function, where $\hat{f}(x)$ denotes the predicted output of the model for a feature vector $x \in \mathbb{R}^p$, which has p dimensions. Suppose the observed feature vector is denoted as $x_j \in \mathbb{R}^n$ and the *j-th* feature is considered as a random variable X_j. The permutation importance of the *j-th* feature is then defined as:

$$PFI_j = \mathbb{E}\left[L\left(y, \hat{f}\left(\tilde{X}_j, X_{j\text{-}1}\right)\right)\right] - \mathbb{E}\left[L\left(y, \hat{f}\left(X_j, X_{-j}\right)\right)\right] \tag{27}$$

Where L is the loss function, y is the positive output, X_j is the original feature vector, $\tilde{X}_j$ is the permuted feature and X_{-j} represents all unchanged features (Molnar, 2022).

In contrast, SHAP explains the contributions of individual features to a model's prediction outcomes using the concept of Shapley values, an additive feature attribution similar to a linear model (Lundberg et al., 2020). Let f represent the original prediction model and g explanatory model. By using local methods (Ribeiro et al., 2016), the prediction $f(x)$ for a given input x can be interpreted through a simplified input Z', a binary vector mapping the original features through a function where each feature is either included to be part of $g(z)$ model or excluded. Lundberg et al. (2017) expressed an additive feature attribution method for a linear explanatory model g as follows:

$$g(Z') = \phi_0 + \sum_{i=1}^{N} \phi_i z'_i \tag{28}$$

Where N represents the number of simplified input features and $Z' \in \{0,1\}^N$, $\phi_0 \in \mathbb{R}$ is the null model output, $\phi_i \in \mathbb{R}$ X_j is the feature attribution for a feature i, that is the Shapley value. The variable z'_i are the observation outcomes, with $(z'_i = 1)$ included or $(z'_i = 0)$ excluded. Additionally, the Shapley value ϕ for a feature i is computed by summing over all the possible feature subsets S that exclude i. Mathematically, this can be expressed as:

$$\phi_i = \sum_{S \subseteq N\{i\}} \frac{|S|!(N - |S| - 1)!}{N} \left[f(S \cup \{i\}) - f(S)\right] \tag{29}$$

Where, N is the total number of features, $f(S)$ is the prediction outcome of the model for the feature subset S and the expression $|S|!(N - |S| - 1)!$ is the weight of feature i in the subset S.

The proposed XAI feature selection method uses the intersection of features selected by both PFI and SHAP to achieve better accuracy while reducing model complexity and execution time. In this work, tree-based SHAP models were implemented as presented in Lundberg et al. (2017). Finally, the classifiers discussed in Section 2.3.4 were trained using our method and the six other conventional feature selection techniques described in Sections 2.2.3.1–6. The performance was evaluated and compared based on key performance parameters: accuracy, model complexity, training time, execution time, and number of features.

2.2.4. Machine learning algorithms

To evaluate the feature selection methods in Section 2.2.3, four machine learning algorithms, successfully used in previous studies for insect classification, were selected. These algorithms were chosen for their lower computational complexity and ability to produce competitive results with fewer computational resources. Unlike deep learning methods, which automatically extract and use features, classical machine learning enables controlled feature selection to optimize performance. Given the objective of this study is to balance accuracy with computational efficiency using feature selection, these algorithms were deemed appropriate, as supported by recent studies (Kasinathan et al., 2020; Srisuradetchai and Suksrikran, 2024). These classifiers are trained on feature matrix X', consisting of the selected feature subset and the corresponding labels y. The goal is to improve performance and reduce computational overload in reduced feature space. Each classifier has a unique method of estimating the decision functions from the

labelled dataset. In our experiment, there are three class labels (corresponding to CPB, ladybirds and aphids) and each feature value represents shape, colour, texture and HOG as described in Section 2. A brief description of these algorithms follows:

2.2.4.1. Support vector machine (SVM) algorithm. SVM is a supervised learning algorithm that separates classes in a dataset by finding the optimal hyperplane. Points in the dataset closer to the hyperplane are referred to as *support vectors* and are critical in defining the margin between classes (Guo and Song, 2018). Let the training dataset be $\{(x_i, y_i), i = 1, 2, 3, \ldots n\}$, where $y_i \epsilon \{+1, -1\}$ which corresponds to class labels and $X' \in R^d$ are the feature vector derived from d-dimensional input vectors. In linear SVM, a hyperplane separating two classes can be represented by the set of points x that satisfy:

$$w.x - b = 0 \tag{30}$$

Where w is the normal vector perpendicular to the hyperplane, x feature vector and b is the bias term. In Eq. (30), the classification problem is equivalent to minimizing the magnitude of w subject to constraint no data points fall within the margin. This means minimizing $\frac{1}{2}\|w\|^2$ subject to the constraint $y_i(w.x_i - b) \geq 1$ for all training points.

For non-linear separable training samples, SVM maps the input vectors into a higher dimensional feature space, where a linear separating hyperplane can be created. This involves the use of a kernel function to reduce computational complexity of the high dimensional feature space. Common kernel functions include linear, Gaussian, sigmoid, and polynomial.

2.2.4.2. Random forest (RF) algorithm. RF is an ensemble machine learning algorithm that combines the strength of multiple decision trees to achieve better predictions. RF algorithms can be used to solve both regression and classification problems. The working principle involves building multiple decision trees and merging their predictions to achieve more accurate and stable results. During training, sampling is performed at both the sample and feature levels. At the sample level, subsets of samples are determined by the bootstrap sampling method to train individual decision trees. At the feature level, feature subsets are randomly selected for information gain computation before splitting the decision tree nodes. The RF model reduces the variance effects of a single decision tree model by synthesizing results of multiple decision trees, although it may not correct bias effectively but ensures no underfitting occurs in each decision tree. Additionally, RF decision trees are independent, this allows the training and prediction processes to be executed simultaneously (Boruah and Biswas, 2023). After training, the final prediction is through majority voting for classification tasks and averaging the predicted probabilities of all the trees for regression tasks. The majority voting process in RF classifier is expressed as follows (Breiman, 2001):

$$f(x) = \underset{y \in Y}{argmax} \sum_{j=1}^{J} I(y = h_j(x)) \tag{31}$$

Where $f(x)$ is the predicted class with the highest votes, y is an actual class, Y is the set of all possible class labels, and $h_j(x)$ represents the *jth* base learner, and $I(.)$ is an indicator function that returns a value of 1 if the prediction is true and 0 if not.

2.2.4.3. K-nearest neighbour (KNN) algorithm. The KNN classifier has been applied in many applications including insect classification and has achieved excellent results in classifying insects on public datasets (Yao et al., 2014; Remboski et al., 2018; Kasinathan et al., 2020). KNN is an instance-based classifier that uses distance metrics such as Euclidean, and Manhattan to estimate similarity between data points. To do classification, each data sample is considered as a point in a two-dimensional plane. The distance between data points is computed and the class of a sample is determined based on the majority votes of its neighbours (Venkateswarlu and Gangula, 2024). For regression tasks, the mean, or median of the k-neighbours can be calculated or a solution to the linear regression problem can be found using the neighbours. The number of nearest neighbour K is always an odd number to avoid any ties in the voting process. Cross-validation is used to choose the best value of k to minimize prediction error (Srisuradetchai and Suksrikran, 2024).

2.2.4.4. Naïve Bayes (NB) algorithm. The NB, derived from probabilistic reasoning is a commonly used classifier in machine learning that assumes feature independence given the class label (Russell and Norvig, 2010). This model considers the "class" variable C to be predicted as the root node and the "Features" variable Xi as the leaf nodes. The features are assumed to be conditionally independent, hence the name Naïve. To perform classification, let c be the number of classes in our dataset, x feature vector values, and p number of features, the probability of a new sample belonging to class c can be summarized as follows (da Silva et al., 2015):

$$P(C = c|x) = P(C = c) \prod_{j=1}^{p} P(F_j = X_j|\mathrm{C} = \mathrm{c}) \tag{32}$$

Where $P(C = c)$ is the probability of an observation belongs to class c, $P(F_j = X_j|\mathrm{C} = \mathrm{c})$ is the probability of feature X_j having value x_j given class c.

2.3. Evaluation metrics

To analyze our results, both model complexity and performance were evaluated. The model complexity was determined by computing the number of learnable parameters needed to construct the model, training and execution time. While complex models can achieve higher accuracy, they often require huge hardware resources and longer training time (Lee and Chen, 2020). Training time, usually measured in seconds, is the time needed to train a model from start to end and mostly depends on model complexity, hardware resources, and data size. Execution time, which reflects how long a model takes to predict a new sample, is crucial for real-world applications. Model performance was also evaluated using standard metrics such as accuracy, precision, recall, and F1 score. These metrics are calculated based on false positive (FP), false negative (FN), true positive (TP) and true negative (TN) as shown in Eqs. (33) to (36). Precision is the percentage of correctly predicted classes. The accuracy of the classifier gives the percentage of the correctly classified positive and negative samples. Recall estimates the fraction of the classes correctly predicted as positives out of the total prediction while the F1 score is the harmonic mean between precision and recall.

$$Precision = \frac{TP}{TP + FP} \tag{33}$$

$$Recall = \frac{TP}{TP + FN} \tag{34}$$

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{35}$$

$$F1\ score = 2\frac{Precision \times Recall}{Precision + Recall} \tag{36}$$

2.4. Model fine-tuning

To improve the performance of our model, parameters were fine-tuned using a 10-fold grid search and 5-fold stratified cross-validation.

Grid search systematically tries a range of hyperparameter values to find the combination that gives optimal performance. Stratified cross-validation is an enhancement of k-fold cross-validation, where the dataset is divided such that each fold maintains the same proportion of class labels as in the original dataset, thus providing a more robust evaluation of the model's performance. This technique prevents overfitting, which occurs when a model performs well on the training set but poorly on the testing set or when predicting new data samples (Russell and Norvig, 2010). Table 2 summarizes the range of parameters used and the final optimal parameters of each classifier.

2.5. Experiments

To evaluate feature selection techniques for insect detection and classification using the insect-plant dataset, we: (1) Trained and evaluated SVM, RF, KNN, and NB classifiers based on standard performance metrics Eqs. (31)–(34) and computational complexity measures, such as the number of parameters, training, and prediction times; (2) identified the top-performing features using XAI feature importance ranking, a collection of consistent, valuable features that contributed to model development; (3) compared the performance of the proposed technique with the six conventional techniques (Section 2.2.3–6) in terms of model complexity and detection accuracy. To evaluate model complexity, training time, execution time, and the number of features selected were computed. Standard metrics (Section 2.3) were used to assess performance. The accuracy and practical suitability of each model for real-world applications were evaluated. All evaluations were conducted using Python version 3.8 on a Dell Optiplex 3050 desktop computer with an Intel Core i5–6500 CPU (3.20 GHz, 4 cores, 4 threads) and 4 GB RAM. These requirements are similar to most single-board computers and edge devices.

3. Results and discussion

3.1. Insect segmentation and region of interest identification

Accurate segmentation of ROI is crucial for feature extraction and insect classification tasks. The GrabCut algorithm was used to segment regions containing insects from the background, enabling feature extraction specifically from insects only. The algorithm performed extremely well in segmenting images of CPB and seven-spot ladybirds with all instances correctly identified (Fig. 6a-f). However, the GrabCut algorithm improperly segmented images where the background closely matched the insect, especially in images containing aphids (Fig. 6g-i). This could also be due to their smaller size as well as colour similarity to the background leaves. To address this, maximally stable extremal regions post-processing was applied to improve segmentation accuracy. This technique focuses on the maximally stable regions in the image, to effectively identify and isolate small objects that may otherwise be challenging to segment due to their size and similarity in colour to the background. Applying this method improved GrabCut segmentation accuracy, correctly identifying almost all the ROIs in the aphids' images (Fig. 6j-l).

A plot of pixel distributions of the foreground and background models shows that an insect represents less than 5 % of the original image (Fig. 7). This highlights the challenge of accurately segmenting small objects. Consequently, this study focuses on the overall performance of the complete machine learning pipeline rather than individual pre-processing steps. However, future work may include a comparative evaluation of image segmentation methods for specific insect species.

Although detecting individual aphids may present challenges, high precision can be achieved in estimating aphids' population and identifying infested leaves or patches of plants by analysing the overall distribution of aphids within an image. In practice, pinpointing individual aphids might not be necessary, especially in the context of on-the-spot spraying methods, where the focus is on targeting infested leaves or areas rather than individual insects. Thus, for aphid detection, it may be more effective to assess the algorithm's ability to estimate the number of aphids and identify the infested leaves for treatment instead of induvial insects, as explored in previous works (Xu et al., 2023; Gao et al., 2024a). This consideration is important for integrating the technique into targeted pest control strategies.

3.2. Explainability and feature importance ranking

The overall contribution of each feature in our dataset was assessed to identify the most influential features in insect classification. This analysis is important for improving the model's performance and generalization ability.

3.2.1. Permutation feature importance ranking

The ranking of features (Table 3) is based on the contribution of each feature; hog_1757 with a standard deviation of 0.020 and mean average of 0.030 had the highest influence on the model's performance followed by compactness. The influence decreases down the table with lower-ranked features, suggesting that the model's performance is less affected by the random permutations of these features.

3.2.2. Shapley additive explanations

The area length ratio (alr) feature had the highest average impact on the model's performance (Fig. 8). Other features such as roundness, length perimeter ratio (lpr), compactness, circularity, and standard deviation histogram (std_hist) also strongly contributed to model outputs. On the other hand, features like the energy, rectangularity, hollowness and histogram of oriented gradients (hog_889) had lower average impacts, suggesting that the model's performance is less affected by these features. This indicates their contribution to the overall model's performance is relatively minimal. The rule of thumb is that features

Table 2
Hyperparameters ranges and optimal values for principal components analysis (PCA), mutual information (MI) and our proposed method. Linear kernel optimized support vector machine model (SVM) performance across all the techniques while parameter ranges varied for k-nearest neighbour (KNN) and random forest (RF) models.

Model	Range of parameters	Optimal parameters	Techniques
SVM	Kernel = [linear, poly, rbf, sigmoid], C = [0.1, 1, 10, 100],	C = 10, kernel = 'linear', probability = True	All + PCA
		C = 100, kernel = 'linear', probability = True	MI
		C = 10, kernel = 'linear', probability = True	Proposed
RF	n_estimators = [100,200,300], max_depth = [None, 10, 20, 30], min_samples_split = [2, 5, 10], min_samples_leaf = [1, 2, 4]	n_estimators = 200, max_depth = None, min_samples_split = 2, min_samples_leaf = 1	All + PCA
		n_estimators = 300, max_depth = None, min_samples_split = 5, min_samples_leaf = 4	MI
		n_estimators = 100, max_depth = None, min_samples_split = 2, min_samples_leaf = 1	Proposed
KNN	n_neighbors = [3, 5, 7], weights = ['uniform', 'distance'], metric = ['euclidean', 'manhattan']	n_neighbors = 3, weights = 'uniform', eetric = 'manhattan'	All + PCA
		n_neighbors = 5, weights = 'distance', metric = 'manhattan'	MI
		n_neighbors = 7, weights = 'uniform', metric = 'manhattan'	Proposed

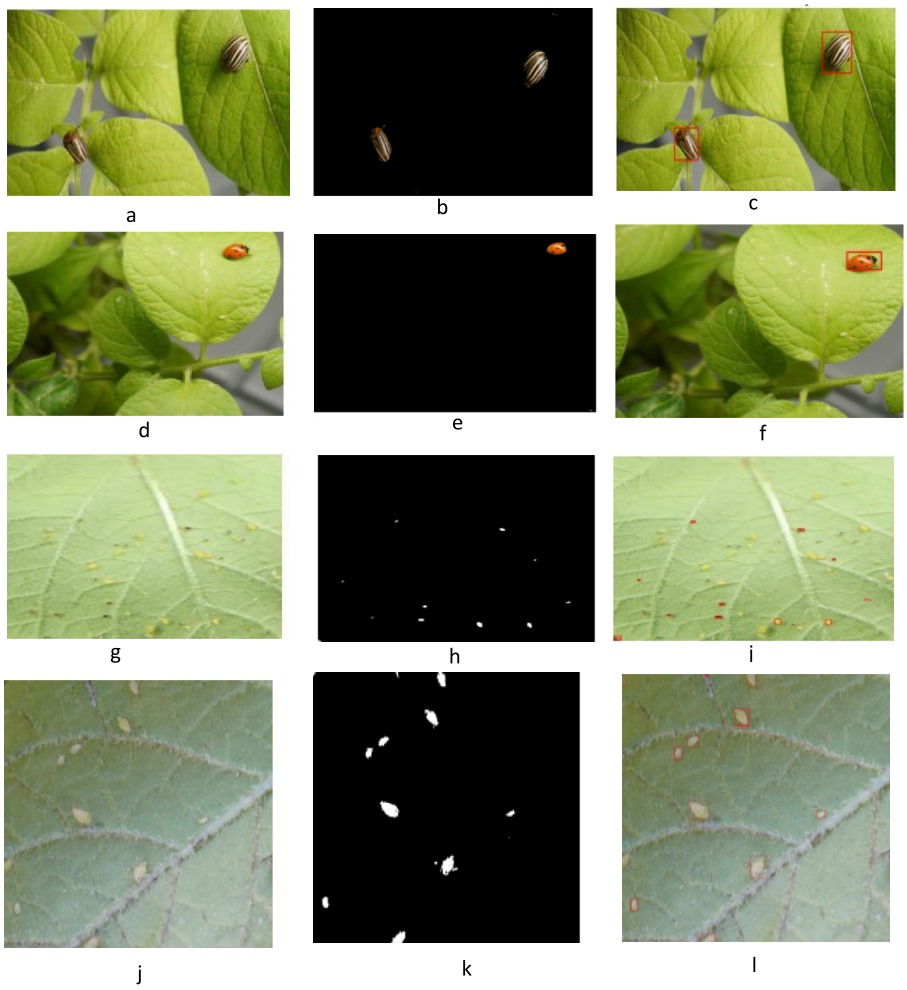


Fig. 6. (a) Original image of Colorado potato beetles on a potato plant, (b) segmented background and foreground regions, (c) identified region of interest (ROI); (d) original image of a seven-spot ladybird on a potato plant, (e) segmented background and foreground regions, (f) identified ROI; (g) original image of aphids on a potato plant, (h) segmented background and foreground regions, (i) identified ROIs (j) original image of aphids (k) Correctly identified background and foreground regions (l) Correctly identified ROIs. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

with higher SHAP values like alr and roundness are likely to have stronger correlations with the target class and, hence, are more influential in determining the predicted outcome. Likewise, features with lower SHAP values such as energy and rectangularity have a lower impact, signifying that their random permutations have a negligible effect on the model's performance.

3.3. Evaluating the performance of feature selection techniques and classifiers

The RF model achieved high performance across all the feature selection techniques, with all metrics above 80 % (Fig. 9a). Our proposed method, MI and Fisher separation criterion produced the best results, with a Fisher separation criterion achieving above 90 % for all metrics and MI yielding the highest precision value at 94.44 %. The PCA and Chi-square were ineffective methods for RF algorithm despite being more computationally demanding than MI. On average, the two top-performing techniques exceeded the PCA with all features by approximately 1.77 % accuracy, 2.13 % precision, 2.34 % recall and 1.84 % F1 score.

The method developed in this study, MI and variance thresholding were the most effective feature selection techniques for the SVM classifier, achieving an average of 85 % for all metrics (Fig. 9b). While other techniques consistently performed well, exceeding 75 %, Fisher separation criterion underperformed, with below 60 % accuracy despite higher computational demands (Table 4). This further highlights the importance of feature selection, demonstrating that performance does not depend on the number of features and computational complexity of the model (Table 4) but on the relevance and suitability of the selected features for the specific algorithm. For example, the Fisher separation criterion proved effective for an RF algorithm but not for SVM.

Fisher separation criterion and our proposed method yielded the best results for the KNN model, with all metrics exceeding 81 % (Fig. 9c). Interestingly, Fisher separation criterion was consistent for both KNN and RF models (Fig. 9a). Both techniques achieved the same results for all metrics, with the highest value of 84.75 % for precision and the lowest value of 83.33 % for recall. The remaining five techniques showed consistently similar results with all metrics above 75 %.

Despite the NB algorithm being rarely used for insect classification, it performed fairly well in our experiment, with all metrics above 75 % across all the feature selection techniques (Fig. 9d). MI produced the best results exceeding 80 % for all metrics. This suggests that MI selects the most relevant features and seems to be the most suitable technique for the Naïve Bayes algorithm. Our proposed method and PCA were the next best-performing methods while chi-square and variance thresholding resulted in lower scores, particularly for recall (76.67 %). These results indicate that with a suitable combination of features, the performance of a classifier can be improved.

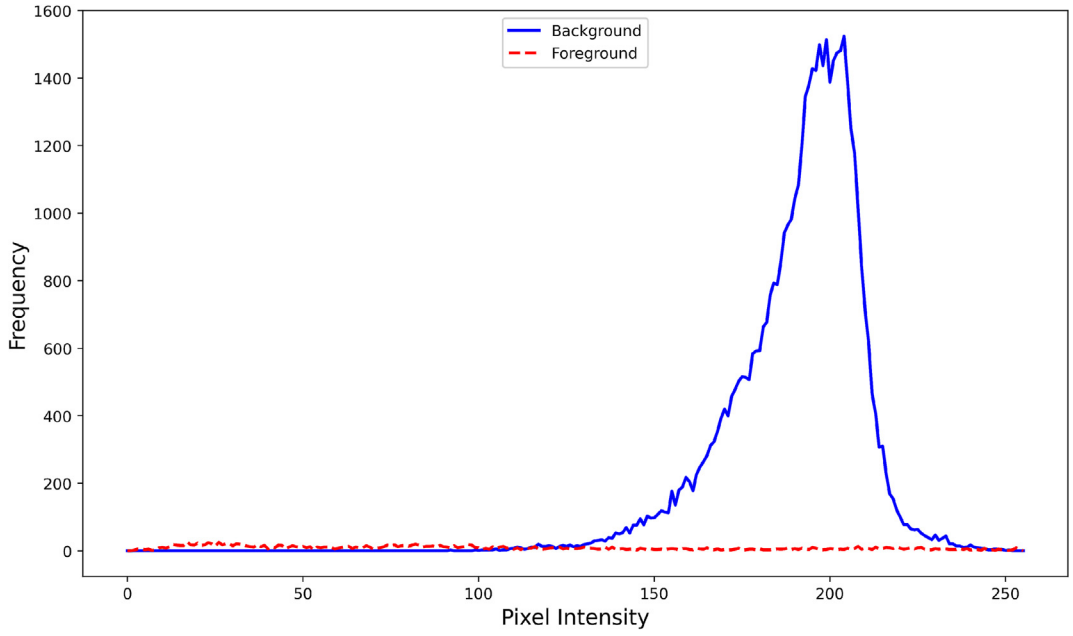


Fig. 7. Image pixels distribution- showing segmented regions of interest in the image, illustrating the distinction between background and insect regions. The red dotted line represents the insect region, capturing all its body parts, and the blue line outlines the background including the surrounding scene. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

3.4. Comparing the proposed method with conventional feature selection techniques

Our proposed method improved performance, producing the highest metric scores while reducing model parameters compared to traditional feature selection techniques (Table 4). This resulted in faster

Table 3
Top 20 features ranked by permutation importance for insect classification: the weight column shows each feature's contribution to classification performance, with higher values indicating greater influence. Features include a histogram of oriented gradients (HOG) for texture and edges (hog_1757, hog_611, hog_1661, etc.), geometric shape descriptors (compactness, area length ratio, hollowness) and colour features (standard deviation and entropy). Permutation importance evaluates the model's prediction error after randomly permuting a feature, providing insights into its significance in the model decision-making. This analysis identifies the most influential features in our dataset.

Weight	Feature
0.030 ± 0.020	hog_1757
0.025 ± 0.010	compactness
0.021 ± 0.022	hog_611
0.020 ± 0.008	hog_1294
0.019 ± 0.008	Alr
0.018 ± 0.007	hog_1661
0.016 ± 0.017	Std_hist
0.016 ± 0.010	hog_1643
0.016 ± 0.023	hog_1266
0.015 ± 0.012	hog_60
0.015 ± 0.007	hog_298
0.015 ± 0.012	hollowness
0.015 ± 0.019	hog_730
0.013 ± 0.008	hog_1578
0.013 ± 0.017	hog_504
0.013 ± 0.008	hog_598
0.013 ± 0.008	entropy
0.013 ± 0.017	hog_56
0.013 ± 0.013	hog_478
0.013 ± 0.008	hog_206

Decreasing influence of features on model predictions.

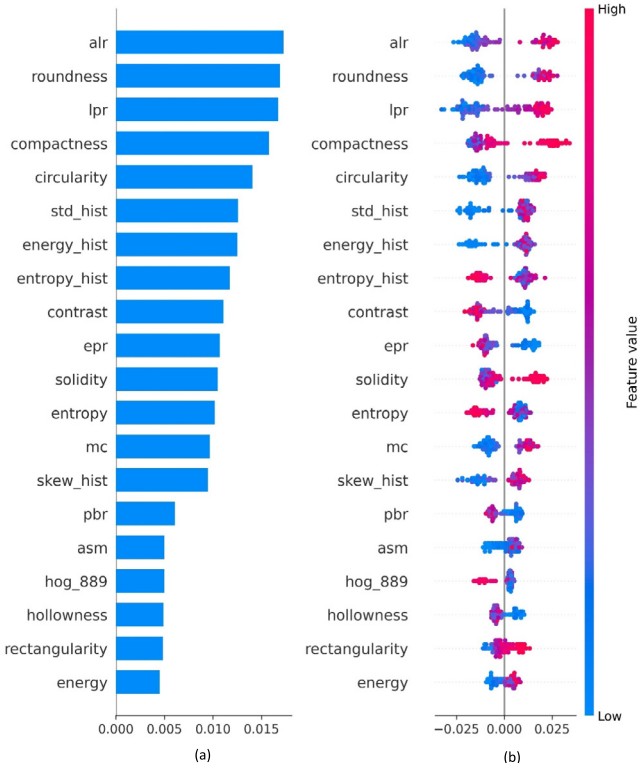


Fig. 8. Shapley values showing the impact of each feature on the random forest model for ranking and selection (a) average feature importance, where bar length represents the contribution of each feature to model performance, longer bars indicate higher influence (b) Feature influence on individual predictions, features ranked from most (area length ratio, alr) to least (energy). The horizontal axis indicates the direction and magnitude of each feature's impact on the model's outcomes, with positive values increasing predictions and negative values reducing them. The colour scale ranges from blue (low values) to red (high values), while the vertical grey line represents a neutral impact (value = 0). This analysis enables informed feature selection to improve model performance. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

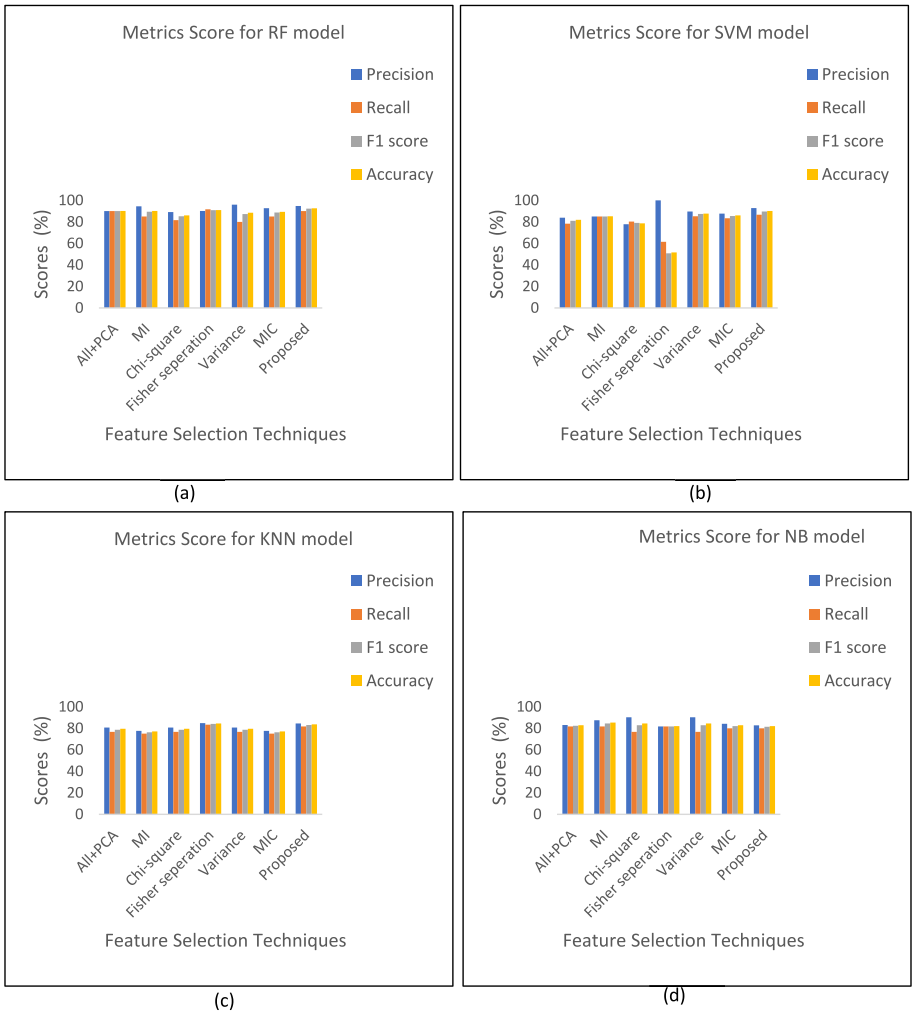


Fig. 9. Performance of classifiers (precision, recall, F1 score and accuracy) across feature selection techniques (principal component analysis (All+PCA), mutual information (MI), chi-squared coefficient, fisher separation criterion, variance thresholding, maximal information coefficient (MIC) and the proposed method). Metrics scores are expressed as percentages (a) random forest (RF) (b) support vector machine (SVM) (c) K-nearest neighbour (KNN) and (d) Naïve Bayes (NB) performance across all techniques. This figure compares the performance of machine learning algorithms used in this study.

training time, especially for the SVM classifier, requiring approximately half of the training time required by conventional feature selection techniques. Our proposed technique prediction time was comparable to top-performing feature selection techniques for all models (Table 4).

The RF classifier consistently achieved the highest accuracy across all feature selection techniques, though it required the highest number of parameters (Table 4). The proposed method, MI and Fisher separation criterion outperformed other feature selection methods for all classifiers, with the proposed method maintaining the best balance between accuracy and computational complexity. NB required the fewest number of parameters, and the fastest training and prediction times, while SVM had the longest training times regardless of feature selection technique used. Notably, the MIC technique required more training time for all algorithms while the KNN classifier consistently achieved lower accuracy and had the slowest prediction times.

Feature selection improved the performance of machine learning algorithms, (Table 4) SVM with XAI (90.16 %) and PCA (81.96 %). This shows that our proposed technique optimized SVM classifier accuracy while maintaining an acceptable number of parameters, training and prediction time. Likewise, our proposed technique optimized an RF classifier, with an accuracy of 92.62 %. The KNN classifier achieved its highest accuracy of 84.43 % using Fisher separation criterion, though have longer prediction times. The NB consistently performs better with feature selection techniques compared to PCA, with accuracy ranging from 81.97 % to 85.25 % and minimal computational demands.

The proposed XAI method reduces computational complexity and basic hardware requirements compared to conventional techniques (Table 5). By selecting only 18 top-performing features instead of 80 (PCA) or 40 (others), XAI minimizes redundancy while maintaining classification performance. Its smaller model size (50–150 MB) compared to (500–1000 MB) for PCA enhances scalability and deployment feasibility. Additionally, XAI reduces training time, memory footprint, and inference costs, making it a practical and efficient alternative to conventional feature selection techniques for insect classification tasks.

3.5. Sensitivity analysis

A sensitivity analysis was conducted to examine the impact of individual features on model performance. Features used in this study are derived from image properties (shape, texture, edges and colour), they are inherently independent of environmental changes such as temperature, humidity and pressure. To thoroughly evaluate our approach, an ablation study was conducted by successively adding features and assessing their impact on accuracy. The accuracy trend was examined for all the feature selection methods as features were added successively. It can be inferred that all feature selection techniques

Table 4
Comparison of accuracy and computational complexity of the proposed method with conventional feature selection techniques for insect classification. Machine learning models (support vector machine, SVM; random forest, RF; K-nearest neighbour, KNN; and Naïve Bayes, NB) were evaluated across various feature selection methods (All features + principal component analysis, All +PCA; mutual information, MI; Fisher separation criterion; chi-square coefficient; maximal information coefficient, MIC; variance thresholding and the proposed explainable artificial intelligence, (XAI) method). Columns show classification accuracy (%), number of model parameters, and training/ prediction times. This comparison evaluates the effectiveness of the XAI method in balancing accuracy and computational efficiency.

Models	Techniques	Accu (%)	Number_param	Training time (s)	Prediction Time (ms)
	All+PCA	81.96	8.96×10^3	1.95×10^4	$<1 \times 10^{-6}$
SVM	MI	85.25	4.04×10^3	9.48×10^3	3.29×10^{-2}
	Fisher	51.64	1.93×10^4	4.15×10^1	3.27×10^{-2}
	Chi-square	78.69	5.52×10^3	3.02×10^3	2.07×10^{-2}
	MIC	86.07	4.52×10^3	8.12×10^3	1.64×10^{-2}
	Variance	87.70	4.60×10^3	1.72×10^4	1.86×10^{-2}
	Proposed	**90.16**	$\mathbf{2.73 \times 10^3}$	$\mathbf{2.87 \times 10^2}$	$\mathbf{2.50 \times 10^{-3}}$
	All+PCA	90.16	7.77×10^7	2.64×10^2	3.53×10^{-1}
RF	MI	90.16	5.28×10^6	2.64×10^2	5.12×10^{-1}
	Fisher	90.98	1.88×10^6	2.88×10^2	2.56×10^{-1}
	Chi-square	86.07	9.09×10^6	2.82×10^2	1.28×10^{-1}
	MIC	89.34	3.91×10^6	2.73×10^2	2.56×10^{-1}
	Variance	88.52	1.45×10^6	2.77×10^2	5.12×10^{-1}
	Proposed	**92.62**	$\mathbf{5.46 \times 10^6}$	$\mathbf{6.02 \times 10^{-1}}$	$\mathbf{3.92 \times 10^{-1}}$
	All+PCA	79.51	4.85×10^4	5.03×10^{-1}	1.40×10^{-1}
KNN	MI	77.04	2.42×10^4	2.63×10^1	0.71×10^1
	Fisher	84.43	2.42×10^4	2.62×10^1	0.72×10^1
	Chi-square	79.51	2.42×10^4	2.86×10^1	0.77×10^1
	MIC	77.05	2.42×10^4	2.63×10^1	0.71×10^1
	Variance	79.51	2.42×10^4	2.97×10^1	0.91×10^1
	Proposed	**83.61**	$\mathbf{1.21 \times 10^4}$	$\mathbf{2.98 \times 10^{-3}}$	$\mathbf{1.31 \times 10^1}$
	All+PCA	82.79	3.20×10^2	$<1 \times 10^{-4}$	1.28×10^{-1}
NB	MI	85.25	1.60×10^2	1.56×10^{-1}	$<1 \times 10^{-6}$
	Fisher	81.97	1.60×10^2	$<1 \times 10^{-4}$	$<1 \times 10^{-6}$
	Chi-square	84.44	1.60×10^2	$<1 \times 10^{-4}$	$<1 \times 10^{-6}$
	MIC	82.79	1.60×10^2	$<1 \times 10^{-4}$	$<1 \times 10^{-6}$
	Variance	84.43	1.60×10^2	$<1 \times 10^{-4}$	$<1 \times 10^{-6}$
	Proposed	**81.97**	$\mathbf{1.32 \times 10^2}$	$\mathbf{3.99 \times 10^{-3}}$	$\mathbf{4.08 \times 10^{-2}}$

demonstrated similar performance; as the number of selected features increased from 1 to 5, accuracy improved (from 50 % to 75 %). However, adding more features (from 6 to 20) led to fluctuations in accuracy for all methods, demonstrating how feature selection methods influence model robustness (Fig. 10). This suggests that while adding informative features might initially improve performance, excessive features may introduce noise, leading to instability. The proposed XAI method achieved an accuracy of 90 % by selecting only the top 11 features and consistently outperformed other methods, suggesting stability across features (Fig. 10). The top 11 features are identified as follows: histogram of oriented gradients (HOG) (hog_1704, hog_680, hog_543, hog_216, hog_1385, hog_809), colour histogram (entropy_hist, energy_hist and std_hist) and shape (elongation and ferrat major axis) features. For the proposed method, adding the first three HOG features increased accuracy to 65 %, suggesting that HOG features are the most influential, followed by colour and shape features. In contrast, other methods exhibited greater fluctuations in performance, underlining the varying contributions of individual features to model effectiveness. These results highlight the importance of selecting optimal features to enhance model generalizability. This analysis provides valuable insights into the stability and reliability of feature selection techniques, thereby supporting the validity of our proposed methodology. A good feature selection method should be sensitive to the inclusion of additional features, indicating the first few features added resulted in a proportional improvement in the accuracy. However, there must be a saturation point beyond which the accuracy becomes stable and further additions of features might yield less accuracy and overfitting.

3.6. Comparison with existing works

To further assess the effectiveness of our proposed method, its performance was compared with existing works. The comparison focused on accuracy as most authors do not evaluate computational complexity. Using the IP102 public dataset (a large-scale benchmark public dataset for insect classification), Kasinathan and Uyyala (2021) achieved slightly higher accuracy, approximately 1.48 % than our RF and SVM models. This might be due to our reduction of the redundant features; a trade-off for faster computation. Additionally, their work is simply image classification, where the entire image is classified as one insect species, without considering the bounding box or the insect's exact location. Our RF model, however, produced a superior performance of 92.62 % compared to the 91.5 % reported by Kasinathan et al. (2020) using a CNN model, despite including all features in their work. This is evidence of the effectiveness of our feature selection method. Similarly, our SVM model, with 91.16 % accuracy outperformed the 85.2 % and 86.81 % accuracy obtained in previous studies (Yao et al., 2014; Liu et al., 2016), despite the simplicity of their task of identifying negative and positive samples and their models requiring longer computational time. This further highlights the advantage of integrating feature selection into the machine learning pipeline, as it improved model performance, reduced computational requirements and enhanced generalization ability. This is achieved by using feature selection to rank and identify the most relevant features in the dataset. Furthermore, deep learning methodologies may require more data and large computing power to produce effective and reliable models (Cserni and Rovid, 2023). However, insect-plant datasets from real in-field scenarios are highly scarce, which might pose difficulties in training reliable models for practical applications. Most researchers rely on data augmentation to artificially increase data samples, thus results obtained by models built using this technique should be interpreted with caution as they might not fully reflect the real field situations.

Samples of visual results (Fig. 11) demonstrate the effectiveness of our developed method in pinpointing the exact position of the insect in diverse image backgrounds. This is a very important requirement for an on-the-spot spraying or targeted pest removal approach. Three

Table 5
Comparison of the basic hardware requirements of the proposed explainable artificial intelligence (XAI) method with principal components analysis (PCA + All features), mutual information (MI), variance thresholding and maximal information coefficient (MIC) using the support vector machine (SVM) model. All experiments used a Dell Optiplex 3050 (Intel Core i5–6500, 4 GB RAM) desktop computer.

Metric	Techniques				
	XAI method	PCA + All features	MI	Variance thresholding	MIC
Training time (hours)	~ 0.08	~ 5.42	~ 2.63	~ 4.78	~ 2.26
Inference time (*ms/sample*)	2.50×10^{-3}	1.0×10^{-6}	3.29×10^{-2}	1.86×10^{-2}	1.64×10^{-2}
RAM usage (*GB*)	~ 0.5	~ 2.0	~ 1.0	~ 1.8	~ 1.9
CPU utilisation (%)	~ 50	~ 70	~ 65	~ 80	~ 84
Energy efficiency (*W*)	~ 5–40	~ 30–50	~ 45–55	~ 48–58	~ 49–60
Model size (*MB*)	~ 50–150	~ 500–1000	~ 500–750	~ 600–800	~ 800–950
Scalability	High	medium	medium	medium	medium

Note: scalability is assessed based on dataset and processing time (High scalability requires less computational resources and medium scalability requires high computational power).

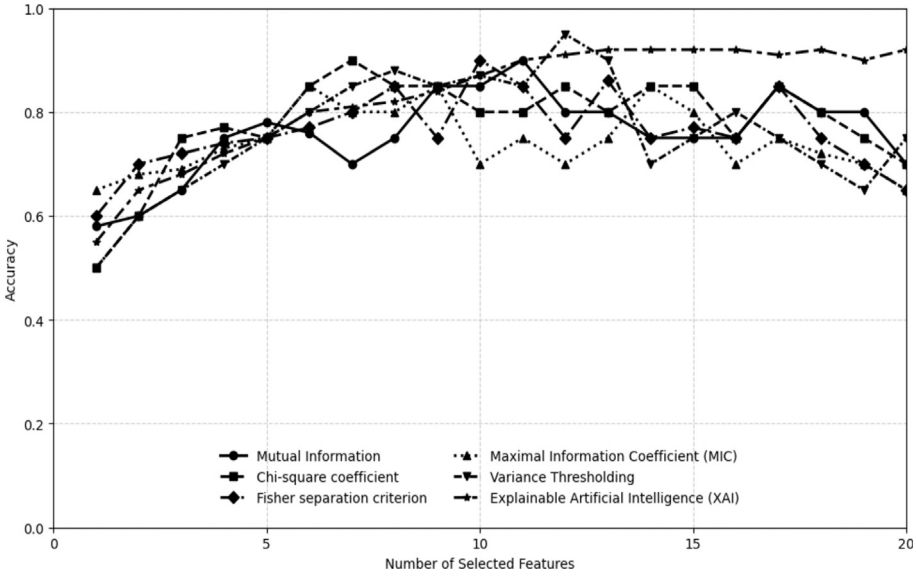


Fig. 10. Impact of individual features on model accuracy-comparing explainable artificial intelligence (XAI) method with existing feature selection techniques on accuracy in classifying insects.

crop plants were used in our experiments to increase data diversity. However, we have not analyzed the impact of individual host plants on the model performance. Future research could delve into this, as crop plant types might affect the performance of machine learning models. Additionally, investigating how having more than one insect species in a single image affects detection accuracy could also be useful, as overlapping objects might affect the performance of machine learning algorithms. In a broader sense, while deep learning methodologies might achieve higher insect identification accuracy, our proposed method offers comparable performance with relatively low hardware requirements. These requirements are important for developing an automatic targeted pest control system for practical applications. Hardware used in agricultural applications is often constrained by size, energy consumption, memory and portability. Agricultural hardware like robots may have limited computational capacity compared to desktop computers or high-end servers in other applications. Hence, keeping a balance between detection accuracy and computational efficiency is desirable. It is worth noting that, for agricultural applications near-perfection detection rate is not always necessary. In many cases, an accuracy of 80 % might be quite sufficient to effectively target individual insect pests and reduce their numbers below the economic threshold, without requiring complete eradication (Crowder et al., 2009). Our method achieves this level of precision while using less computational power, this is important for real-world deployment, as it makes the technique cheaper and accessible.

Our findings further suggest that model computational complexity depends largely on the number of features and algorithm complexity, while prediction time is affected by model parameters and memory requirements. Training time is determined by the feature selection technique and classifier type, not the number of model parameters. Most importantly, model accuracy is independent of the number of features. Furthermore, the recorded training and prediction times for the proposed method using SVM are approximately 4 h and 13 ms, respectively, with a model's size of around 50 MB and accuracy of (91.16 %) (Table 4). These modest hardware requirements indicate that the approach is computationally feasible for real-world deployment on single-board hardware like Raspberry Pi and standard desktop computers or cloud-based systems with similar or better specifications.

4. Conclusion

This study evaluates the significance of feature selection in improving the performance of machine learning algorithms for individual insect detection. Our proposed approach used explainable artificial intelligence (XAI) to identify the optimal feature set that maximized the performance of machine learning algorithms for the detection of individual pest and beneficial insects (Colorado potato beetles on potato plants and seven-spot ladybirds, aphids on potato, faba bean and sugar beet plants). Laboratory and field-collected insect-plant datasets were used to evaluate and compare several feature selection techniques, with XAI achieving an accuracy of 90.16 % and 92.62 for support vector machine and random forest, respectively. This approach improved accuracy compared to using a principal components analysis without feature selection and conventional top-performing feature selection techniques such as Fisher separation criterion, mutual information, and maximal information coefficient. Additionally, the XAI method reduced the model's computational complexity compared to traditional methods, resulting in shorter training and prediction times. For K-nearest neighbour and Naïve Bayes algorithms, our approach achieved results comparable to the top-performing feature selection techniques while reducing the number of model parameters, training and prediction times. The faster training and prediction time recorded highlights the potential of XAI method for real-time applications, making it a promising approach for deployment in low-cost hardware and edge computing devices for easy integration with mechanical parts such as robotic arms for targeted-pest removal or portable devices like mobile phones for automatic pest monitoring. The low hardware requirements and shorter prediction times make the proposed method suitable for integration with agricultural machinery, such as tractors and drones or deployment in edge devices for insect pest monitoring in field crops. This XAI-based feature selection approach advances machine learning techniques for pest detection, particularly under hardware resource constraints, while also managing computational complexity to facilitate real-time applications. The proposed method's ability to identify individual insects makes it suitable for targeted removal, minimizing ecological impacts on non-target species.

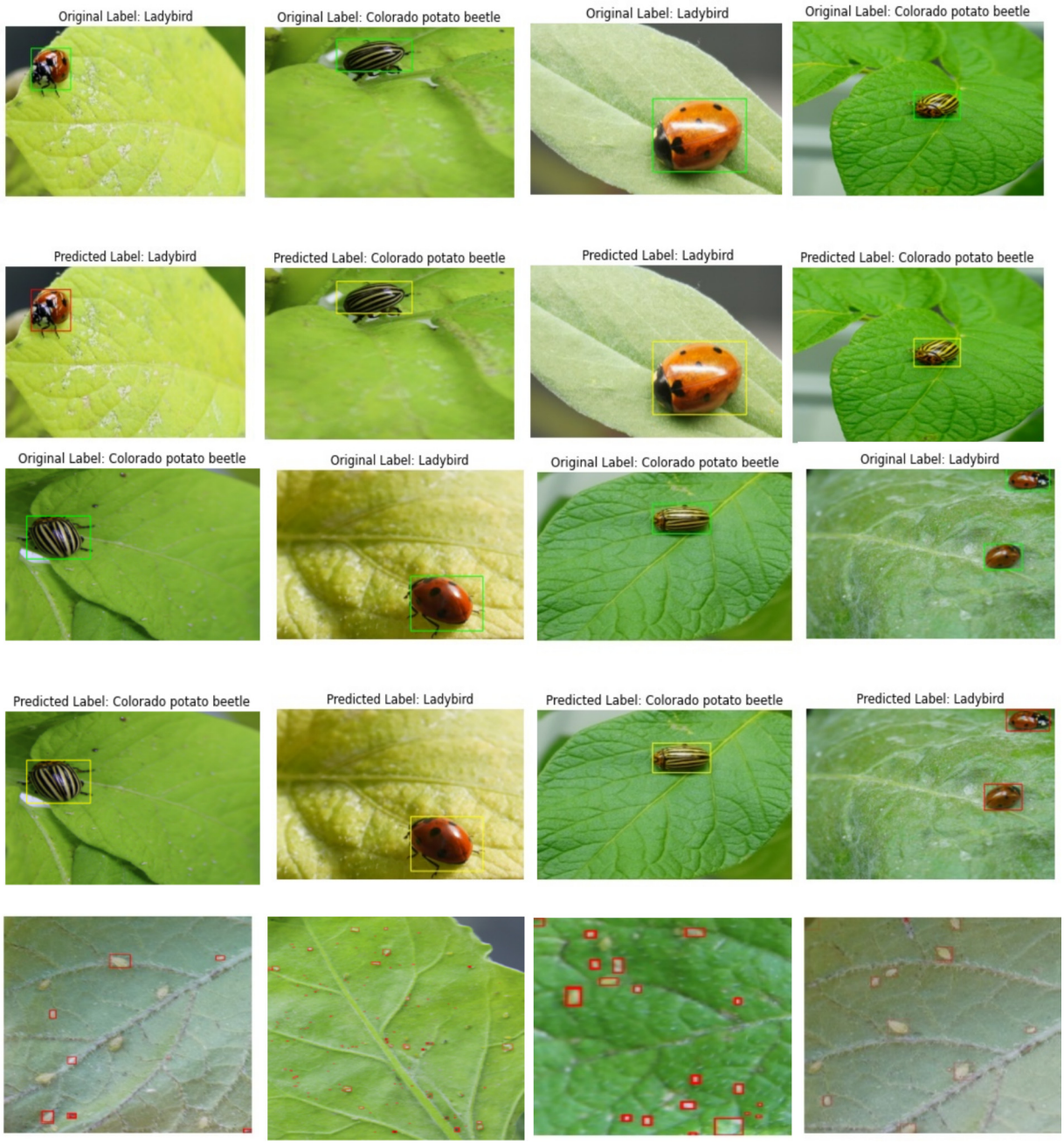


Fig. 11. Visual results; showing beneficial and pest insects (seven-spot ladybird and aphids on potato, faba bean and sugar beet plants; Colorado potato beetles on potato plant only;) with original labels, and predictions made by the developed model using explainable artificial intelligence (XAI) method. The model is trained on an insect-plant dataset containing images of the three insects and three crop plants taken in laboratory and field settings to improve data diversity and model robustness. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

Finally, our findings show that combining a machine learning algorithm with an ideal feature selection technique can achieve robust performance without requiring complex and computationally demanding methodologies. This is essential for real-world applications with resource constraints where hardware limitations must be considered. Our method offers comparable performance with relatively low hardware requirements. These are important for developing an automatic targeted pest control system for practical applications, as hardware used in agricultural applications is often constrained by size, energy consumption, memory and portability. Further research should focus on limitations posed by external factors such as lighting variations, which remain challenging in insect detection across machine learning techniques, especially for aphid detection on plant material in field conditions due to their smaller size and similarity to green leaves. Future research could explore combining feature selection with deep learning or the potential of artificial lighting and synthetic data to improve the accuracy of small insect detection in complex scenes.

CRediT authorship contribution statement

Rabiu Aminu: Writing – original draft, Visualization, Methodology, Formal analysis, Data curation, Conceptualization. **Samantha M. Cook:** Writing – review & editing, Validation, Supervision, Resources, Conceptualization. **David Ljungberg:** Writing – review & editing, Supervision,

Conceptualization. **Oliver Hensel:** Writing – review & editing, Supervision, Resources, Project administration, Funding acquisition, Conceptualization. **Abozar Nasirahmadi:** Writing – review & editing, Supervision, Software, Resources, Project administration, Methodology, Funding acquisition, Conceptualization.

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.

Acknowledgement

This research was conducted as part of the DigiPlus project "Digitale Experimentierfelder", funded by the German Federal Ministry of Food and Agriculture (BMEL) through the Federal Agency of Food and Agriculture (BLE) with grant number "28DE207A21". SMC's input was funded by the UK NERC/BBSRC AgZero+ Project NE/W005050/1. We sincerely thank the funding bodies for supporting this research work.

References

Agrawal, N., Urma, S., Padhan, S., Barik, R.C., 2018. Indian agro based Pest region detection by clustering and Pseudo-color image processing. Int. J. Eng. Res. Technol. 7 (3), 380–385. www.ijert.org.

Alkan, E., Aydın, A., 2023. Image processing techniques based feature extraction for insect damage areas. Eur. J. For. Eng. 9 (1), 34–40. https://doi.org/10.33904/ejfe.1320121.

Balaško, M.K., Mikac, K.M., Bažok, R., Lemic, D., 2020. Modern techniques in Colorado potato beetle (*Leptinotarsa decemlineata* say) control and resistance management: history review and future perspectives. Insects 11 (9), 1–17 MDPI AG. https://doi.org/10.3390/insects11090581.

Beaumelle, L., Tison, L., Eisenhauer, N., Hines, J., Malladi, S., Pelosi, C., Thouvenot, L., Phillips, H.R.P., 2023. Pesticide effects on soil fauna communities—a meta-analysis. J. Appl. Ecol. 60 (7), 1239–1253 John Wiley and Sons Inc. https://doi.org/10.1111/1365-2664.14437.

Betti Sorbelli, F., Palazzetti, L., Pinotti, C.M., 2023. YOLO-based detection of *Halyomorpha halys* in orchards using RGB cameras and drones. Comput. Electron. Agric. 213. https://doi.org/10.1016/j.compag.2023.108228.

Bhosale, J.D., Thorat, S.S., Pancholi, P.V., Mutkule, P.R., 2023. Machine learning-based algorithms for the detection of leaf disease in agriculture crops. Int. J. Recent Innov. Trends Comput. Commun. 11, 45–50. https://doi.org/10.17762/ijritcc.v11i5s.6596.

Bick, E., Sigsgaard, L., Torrance, M.T., Helmreich, S., Still, L., Beck, B., El Rashid, R., Lemmich, J., Nikolajsen, T., Cook, S.M., 2024. Dynamics of pollen beetle (*Brassicogethes aeneus*) immigration and colonization of oilseed rape (*Brassica napus*) in Europe. Pest Manag. Sci. 80 (5), 2306–2313.

Bitkov, M.P., Lykov, I.N., 2024. Efficacy of three bioinsecticides for control of Colorado potato beetle on potatoes. E3S Web of Conferences. 486. https://doi.org/10.1051/e3sconf/202448602033.

Boruah, A.N., Biswas, S.K., 2023. I-RF: A Transparent Decision-Making System. Alliance University, National Institute of Technology Silchar. https://doi.org/10.21203/rs.3.rs-2734607/v1.

Breiman, L., 2001. Random forests. Mach. Learn. 45 (1), 5–32.

Chaudhary, V.K., Arya, S., Singh, P., 2021. Effects of pesticides on biodiversity and climate change. Int. J. Environ. Sci. 12 (2). https://doi.org/10.53390/ijes.v12i2.1.

Chen, J., Chen, W., Zeb, A., Zhang, D., Nanehkaran, Y.A., 2021. Crop pest recognition using attention-embedded lightweight network under field conditions. Appl. Entomol. Zool. 56 (4), 427–442. https://doi.org/10.1007/s13355-021-00732-y.

Crowder, D.W., Onstad, D.W., Gray, M.E., 2009. Planting transgenic insecticidal corn based on economic thresholds: consequences for integrated Pest management and insect resistance management. J. Econ. Entomol. 99 (3), 899–907. https://doi.org/10.1603/0022-0493-99.3.899.

Cserni, M., Rovid, A., 2023. Combining classical and neural approaches for image segmentation. 2023 IEEE 21st World Symposium on Applied Machine Intelligence and Informatics, SAMI 2023 - Proceedings, pp. 33–38. https://doi.org/10.1109/SAMI58000.2023.10044498.

Cuevas, C., Berjón, D., García, N., 2023. A fully automatic method for segmentation of soccer playing fields. Sci. Rep. 13 (1). https://doi.org/10.1038/s41598-023-28658-1.

da Silva, F.L., Grassi Sella, M.L., Francoy, T.M., Costa, A.H.R., 2015. Evaluating classification and feature selection techniques for honeybee subspecies identification using wing images. Comput. Electron. Agric. 114, 68–77. https://doi.org/10.1016/j.compag.2015.03.012.

Dave, D., Naik, H., Singhal, S., Patel, P., 2020. Explainable AI meets Healthcare: A Study on Heart Disease Dataset. http://arxiv.org/abs/2011.03195.

Don, D.R., Aygun, R., Karakaya, M., 2023. A multistage framework for detection of very small objects. ACM Int. Conf. Proc. Ser., 9–14. https://doi.org/10.1145/3589572.3589574.

Dupuis, B., Nkuriyingoma, P., Ballmer, T., 2023. Economic impact of potato virus Y (PVY) in europe. Potato Res. 10. https://doi.org/10.1007/s11540-023-09623-xï.

El-Kenawy, E.M., Khodadadi, N., Mirjalili, S., Abdelhamid, A.A., Eid, M.M., Ibrahim, A., 2024. Greylag goose optimization: nature-inspired optimization algorithm. Expert Syst. Appl. 238 (Part E), 122147. https://doi.org/10.1016/j.eswa.2023.122147.

Espinoza, K., Valera, D.L., Torres, J.A., López, A., Molina-Aiz, F.D., 2016. Combination of image processing and artificial neural networks as a novel approach for the identification of *Bemisia tabaci* and *Frankliniella occidentalis* on sticky traps in greenhouse agriculture. Comput. Electron. Agric. 127, 495–505. https://doi.org/10.1016/j.compag.2016.07.008.

Estévez, P.A., Tesmer, M., Perez, C.A., Zurada, J.M., 2009. Normalized mutual information feature selection. IEEE Trans. Neural Netw. 20 (2), 189–201. https://doi.org/10.1109/TNN.2008.2005601.

FAO, 2021. Climate change fans spread of pests and threatens plants and crops, new FAO study. Food Agric. Organ. [Online]. Available at: https://www.fao.org/newsroom/detail/Climate-change-fans-spread-of-pests-and-threatens-plants-and-crops-new-FAO-study/zh.

Figueiredo, J.C., Medeiros Neto, F.G., de J'unior, I.C.P., 2016. Contour-based feature extraction for image classification and retrieval. 2016 35th International Conference of the Chilean Computer Science Society (SCCC). Valparaiso, Chile, pp. 1–7.

Fisher, A., Rudin, C., Dominici, F., 2019. All models are wrong, but many are useful: learning a variable's importance by studying an entire class of prediction models simultaneously. J. Mach. Learn. Res. 20. http://jmlr.org/papers/v20/18-760.html.

Gao, X., Xue, W., Lennox, C., Stevens, M., Gao, J., 2024a. Developing a hybrid convolutional neural network for automatic aphid counting in sugar beet fields. Comput. Electron. Agric. 220. https://doi.org/10.1016/j.compag.2024.108910.

Gao, Y., Xue, X., Qin, G., Li, K., Liu, J., Zhang, Y., Li, X., 2024b. Application of machine learning in automatic image identification of insects a review. Eco. Inform. https://doi.org/10.1016/j.ecoinf.2024.102539.

Guo, Y., Song, X., 2018. Support vectors classification method based on projection vector boundary feature. In: Jiang, X., Hwang, J.-N. (Eds.), Proc.SPIE, Tenth International Conference on Digital Image Processing (ICDIP 2018). SPIE, p. 312 https://doi.org/10.1117/12.2503318.

Haralick, R.M., Shanmugam, K., Dinstein, I., 1973. Textural_features_for_image_classification. IEEE Trans. Syst. Man Cybern. 3 (6), 610–621.

Hasan, M.M., Shaqib, S., Akter, M.S., Alam, R., Haque, A.U., Khushbu, S.A., 2024. Unleashing the power of transfer learning model for sophisticated insect detection: revolutionizing insect classification. ArXiv [Online]. Availabe at: https://arxiv.org/abs/2406.07716.

He, C., Li, K., Zhang, Y., Xu, G., Tang, L., Zhang, Y., Guo, Z., Li, X., 2023. Weakly-supervised concealed object segmentation with sam-based pseudo labeling and multi-scale feature grouping. 37th Conference on Neural Information Processing Systems (NeurIPS 2023). https://github.com/ChunmingHe/WS-SAM.

Kasinathan, T., Uyyala, S.R., 2021. Machine learning ensemble with image processing for pest identification and classification in field crops. Neural Comput. & Applic. 33 (13), 7491–7504. https://doi.org/10.1007/s00521-020-05497-z.

Kasinathan, T., Uyyala, S.R., 2023. Detection of fall armyworm (*spodoptera frugiperda*) in field crops based on mask R-CNN. SIViP 17 (6), 2689–2695. https://doi.org/10.1007/s11760-023-02485-3.

Kasinathan, T., Singaraju, D., Uyyala, S.R., 2020. Insect classification and detection in field crops using modern machine learning techniques. Inf. Process. Agric. 8 (3), 446–457. https://doi.org/10.1016/j.inpa.2020.09.006.

Kini, A.S., Reddy, P.K.V., Pai, S.N., 2023. Techniques of deep learning and image processing in plant leaf disease detection: a review. Int. J. Electr. Comput. Eng. 13 (3), 3029–3040. https://doi.org/10.11591/ijece.v13i3.pp3029-3040.

Kirkeby, C., Rydhmer, K., Cook, S.M., Strand, A., Torrance, M.T., Swain, J.L., Prangsma, J., Johnen, A., Jensen, M., Brydegaard, M., Græsbøll, K., 2021. Advances in automatic identification of flying insects using optical sensors and machine learning. Sci. Rep. 11 (1). https://doi.org/10.1038/s41598-021-81005-0.

Lee, R., Chen, I.Y., 2020. The time complexity analysis of neural network model configurations. Proceedings - 2nd International Conference on Mathematics and Computers in Science and Engineering, MACISE 2020, pp. 178–183. https://doi.org/10.1109/MACISE49704.2020.00039.

Lei, Z., Liao, S., Li, S.Z., 2012. Efficient feature selection for linear discriminant analysis and its application to face recognition. Proceedings of the 21st International Conference on Pattern Recognition (ICPR2012), Tsukuba, Japan, pp. 1136–1139.

Li, Y., Xia, C., Lee, J., 2015. Detection of small-sized insect pest in greenhouses based on multifractal analysis. Optik 126 (19), 2138–2143. https://doi.org/10.1016/j.ijleo.2015.05.096.

Liang, J., Palaoag, T.D., 2024. Developing a deep learning-based GrabCut target image automatic segmentation algorithm. Proceedings of the 2024 4th International Conference on Machine Learning and Intelligent Systems Engineering (MLISE), pp. 187–191. https://doi.org/10.1109/MLISE62164.2024.10674591.

Liu, T., Chen, W., Wu, W., Sun, C., Guo, W., Zhu, X., 2016. Detection of aphids in wheat fields using a computer vision technique. Biosyst. Eng. 141, 82–93. https://doi.org/10.1016/j.biosystemseng.2015.11.005.

Lucero, R.-S., Tetyana, B., Ernst, K., Alberto, E.-E., María, T., González, R., 2015. Recognition of pests on crops with a random subspace classifier. 2015 International Work Conference on Bio-Inspired Intelligence (IWOBI).

Lundberg, S.M., Allen, P.G., Lee, S.-I., 2017, November 25. A unified approach to interpreting model predictions. 31st Conference on Neural Information Processing Systems (NIPS 2017), Long Beach, CA, USA. https://github.com/slundberg/shap.

Lundberg, S.M., Erion, G., Chen, H., DeGrave, A., Prutkin, J.M., Nair, B., Katz, R., Himmelfarb, J., Bansal, N., Lee, S.I., 2020. From local explanations to global understanding with explainable AI for trees. Nat. Mach. Intell. 2 (1), 56–67. https://doi.org/10.1038/s42256-019-0138-9.

Miranda, J.C., Arnó, J., Gené-Mola, J., Lordan, J., Asín, L., Gregorio, E., 2023. Assessing automatic data processing algorithms for RGB-D cameras to predict fruit size and weight

in apples. Comput. Electron. Agric. 214. https://doi.org/10.1016/j.compag.2023.108302.
Molnar, C., 2022. Interpretable Machine Learning: A Guide for Making Black Box Models Explainable. Second edition. Christoph Molnar, Munich, Germany.
Pathak, H., Igathinathane, C., Howatt, K., Zhang, Z., 2023. Machine learning and handcrafted image processing methods for classifying common weeds in corn field. Smart Agric. Technol. 5. https://doi.org/10.1016/j.atech.2023.100249.
Remboski, T.B., De Souza, W.D., De Aguiar, M.S., Ferreira, P.R., 2018. Identification of fruit fly in intelligent traps using techniques of digital image processing and machine learning. Proc. ACM Symp. Appl. Comput., 260–267. https://doi.org/10.1145/3167132.3167155.
Reshef, D.N., Reshef, Y.A., Finucane, H.K., Grossman, S.R., Mcvean, G., Turnbaugh, P.J., Lander, E.S., Mitzenmacher, M., Pardis, Sabeti, C., 2011. Detecting novel associations in large data sets. Science (New York, N.Y.) 384, 1518–1524. https://doi.org/10.1126/science.1205438.
Ribeiro, M.T., Singh, S., Guestrin, C., 2016. "Why should I trust you?" explaining the predictions of any classifier. Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 13–17-August-2016, pp. 1135–1144. https://doi.org/10.1145/2939672.2939778.
Richard, I., 2010. Pesticides and the loss of biodiversity How intensive pesticide use affects wildlife populations and species diversity Written by Richard Isenring. www.pan-europe.info.
Rother, C., Kolmogorov, V., Blake, A., 2004. "GrabCut"-Interactive Foreground Extraction Using Iterated Graph Cuts.
Russell, S.J., Norvig, P., 2010. Artificial Intelligence a Modern Approach Third Edition. 3rd ed. Prentice Hall, United states.
Ryo, M., 2022. Explainable artificial intelligence and interpretable machine learning for agricultural data analysis. Artif. Intell. Agric. 6, 257–265. https://doi.org/10.1016/j.aiia.2022.11.003.
Sablon, L., Dickens, J.C., Haubruge, É., Verheggen, F.J., 2013. Chemical ecology of the Colorado potato beetle, *Leptinotarsa decemlineata* (say) (Coleoptera: Chrysomelidae), and potential for alternative control methods. Insects 4 (1), 31–54. https://doi.org/10.3390/insects4010031.
Sahin, H.M., Miftahushudur, T., Grieve, B., Yin, H., 2023. Segmentation of weeds and crops using multispectral imaging and CRF-enhanced U-net. Comput. Electron. Agric. 211. https://doi.org/10.1016/j.compag.2023.107956.
Sergyán, S., 2008. Color histogram features based image classification in content-based image retrieval systems. SAMI 2008 6th International Symposium on Applied Machine Intelligence and Informatics - Proceedings, pp. 221–224. https://doi.org/10.1109/SAMI.2008.4469170.
Singh, S., Roy, Y., Bhan, A., Sah, S., 2023. Computer based detection and classification of leaf diseases using hybrid features. International Conference on Sustainable Computing and Smart Systems, ICSCSS 2023 - Proceedings, pp. 788–793. https://doi.org/10.1109/ICSCSS57650.2023.10169167.
Sosa-Cabrera, G., Gómez-Guerrero, S., García-Torres, M., Schaerer, C.E., 2023. Feature selection: A perspective on inter-attribute cooperation. Int. J. Data Sci. Anal. 17 (2), 139–151 Springer Science and Business Media Deutschland GmbH. https://doi.org/10.1007/s41060-023-00439-z.
Srisuradetchai, P., Suksrikran, K., 2024. Random kernel k-nearest neighbors regression. Front. Big Data. 7. https://doi.org/10.3389/fdata.2024.1402384.
Sumesh, K.C., Ninsawat, S., Som-ard, J., 2021. Integration of RGB-based vegetation index, crop surface model and object-based image analysis approach for sugarcane yield estimation using unmanned aerial vehicle. Comput. Electron. Agric. 180. https://doi.org/10.1016/j.compag.2020.105903.
Suresh, M., Xin, T., Cook, S., Evans, D., 2025. Bugs and bytes: entomological biomonitoring through the integration of deep learning and molecular analysis for merged community and network analysis. Agric. For. Entomol. 27, e12667. https://doi.org/10.1111/afe.12667.
Tan, K., Ding, S., Wu, S., Tian, K., Ren, J., 2023. A small object detection network based on multiple feature enhancement and feature fusion. Sci. Program. 2023. https://doi.org/10.1155/2023/5500078.
Venkateswarlu, B., Gangula, R., 2024. Exploring the power and practical applications of k-nearest neighbours (knn) in machine learning. J. Comput. Allied Intell. 2 (1), 8–15. https://doi.org/10.69996/jcai.2024002.
Vergara, J.R., Estévez, P.A., 2014. A review of feature selection methods based on mutual information. Neural Comput. & Applic. 24 (1), 175–186. https://doi.org/10.1007/s00521-013-1368-0.
Wan, Y., Liao, Z., Liu, J., Song, W., Ji, H., Gao, Z., 2023. Small object detection leveraging density-aware scale adaptation. Photogramm. Rec. 38 (182), 160–175. https://doi.org/10.1111/phor.12446.
Wang, A., Zhang, W., Wei, X., 2019. A review on weed detection using ground-based machine vision and image processing techniques. Computers and Electronics in Agriculture. 158. Elsevier B.V, pp. 226–240. https://doi.org/10.1016/j.compag.2019.02.005.
Xia, C., Chon, T.S., Ren, Z., Lee, J.M., 2015. Automatic identification and counting of small size pests in greenhouse conditions with low computational cost. Eco. Inform. 29 (P2), 139–146. https://doi.org/10.1016/j.ecoinf.2014.09.006.
Xia, D., Chen, P., Wang, B., Zhang, J., Xie, C., 2018. Insect detection and classification based on an improved convolutional neural network. Sensors (Switzerland) 18 (12). https://doi.org/10.3390/s18124169.
Xiao, Z., 2024. Unsupervised feature selection via principal component analysis and possibillistic graph learning. 2024 5th International Seminar on Artificial Intelligence, Networking and Information Technology, AINIT 2024, pp. 1502–1509. https://doi.org/10.1109/AINIT61980.2024.10581792.
Xu, W., Xu, T., Alex Thomasson, J., Chen, W., Karthikeyan, R., Tian, G., Shi, Y., Ji, C., Su, Q., 2023. A lightweight SSV2-YOLO based model for detection of sugarcane aphids in unstructured natural environments. Comput. Electron. Agric. 211. https://doi.org/10.1016/j.compag.2023.107961.
Yao, Q., Xian, D. Xiang, Liu, Q. Jie, Yang, B. Jun, Diao, G. Qiang, Tang, J., 2014. Automated counting of rice planthoppers in paddy fields based on image processing. J. Integr. Agric. 13 (8), 1736–1745. https://doi.org/10.1016/S2095-3119(14)60799-1.
Ye, Z., Yang, K., Lin, Y., Guo, S., Sun, Y., Chen, X., Lai, R., Zhang, H., 2023. A comparison between pixel-based deep learning and object-based image analysis (OBIA) for individual detection of cabbage plants based on UAV visible-light images. Comput. Electron. Agric. 209. https://doi.org/10.1016/j.compag.2023.107822.
Zhai, Y., Song, W., Liu, X., Liu, L., Zhao, X., 2018. A chi-square statistics based feature selection method in text classification. 2018 IEEE 9th International Conference on Software Engineering and Service Science (ICSESS), Beijing, China, pp. 160–163.
Zhang, Y., Yuan, J., Liu, H., Li, Q., 2017. GrabCut image segmentation algorithm based on structure tensor. J. China Univ. Posts Telecommun. 24 (2), 38–47. https://doi.org/10.1016/S1005-8885(17)60197-3.
Zhang, D., Yang, S., Yuan, X., Zhang, P., 2021. Interpretable deep learning for automatic diagnosis of 12-lead electrocardiogram. IScience 24 (4). https://doi.org/10.1016/j.isci.2021.102373.
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. https://doi.org/10.1016/j.compag.2024.108726.