



LEEDS
BECKETT
UNIVERSITY

Citation:

Patel, VK and Abhishek, K and Selvarajan, S (2024) Optimized recurrent neural network-based early diagnosis of crop pest and diseases in agriculture. *Discover Computing*, 27 (43). pp. 1-32. ISSN 2948-2992 DOI: <https://doi.org/10.1007/s10791-024-09481-2>

Link to Leeds Beckett Repository record:

<https://eprints.leedsbeckett.ac.uk/id/eprint/11574/>

Document Version:

Article (Published Version)

Creative Commons: Attribution-Noncommercial-No Derivative Works 4.0

© The Author(s) 2024

The aim of the Leeds Beckett Repository is to provide open access to our research, as required by funder policies and permitted by publishers and copyright law.

The Leeds Beckett repository holds a wide range of publications, each of which has been checked for copyright and the relevant embargo period has been applied by the Research Services team.

We operate on a standard take-down policy. If you are the author or publisher of an output and you would like it removed from the repository, please [contact us](#) and we will investigate on a case-by-case basis.

Each thesis in the repository has been cleared where necessary by the author for third party copyright. If you would like a thesis to be removed from the repository or believe there is an issue with copyright, please contact us on openaccess@leedsbeckett.ac.uk and we will investigate on a case-by-case basis.

Research

Optimized recurrent neural network-based early diagnosis of crop pest and diseases in agriculture

Vijesh Kumar Patel¹ · Kumar Abhishek¹ · Shitharth Selvarajan²

Received: 26 August 2024 / Accepted: 18 October 2024

Published online: 11 November 2024

© The Author(s) 2024 [OPEN](#)

Abstract

The productivity of agriculture plays a critical role in the Indian economy. Growing crop production is a critical responsibility nowadays to accommodate citizen demand and provide farmers with greater rewards. Therefore, a machine learning (ML) technique is employed to more precisely identify diseases and pests on leaves and other crop parts. This paper introduces a machine learning-based system in early crop disease and pest detection using image processing and optimization. Initially, the data is collected from the CCMT plant disease Dataset. Image augmentation techniques such as rotation, flipping, and zooming are utilized to make the dataset wholesome. After amplification, the pre-processing is carried out on these images. Noise reduction as well as enhancing quality are done by Adaptive Bilateral Filter. Lanczos interpolation technique resized it and normalization is done so that the analysis can proceed. Kapur's Entropy-based Whale Optimization is introduced for the segmentation of the image efficiently by dividing diseased areas into segments. The features are extracted using the Gray Level Co-occurrence Matrix, which assesses relationships among the pixels and produces an appropriate feature matrix for color images. This processed data then feeds into a Moth-Flame Optimized Recurrent Neural Network for crop disease and pest detection. These results achieved high accuracy levels at 98.4% for cashews, 98.3% for cassava, 98.5% for maize, and 96.8% for tomato crops, outperforming all the reported techniques.

Keywords Agriculture · Crop pest · Diseases · Early diagnosis · Machine learning

1 Introduction

Crop production is substantially raised if stresses are identified at the earliest possible time to facilitate the adoption of required prevention measures [1]. One of the most difficult jobs in agriculture is identifying plant diseases early on. Early disease detection was crucial for increasing agricultural yield [2]. With the application of ML and deep learning (DL) techniques, this issue has been resolved. Large crop farms were now automatically detecting plant illnesses, which is advantageous because it cuts down on monitoring time [3]. The demand for additional food increased due to the population's rapid rise [4]. Agricultural enterprises are adapting their methods to ML technologies to attain enhanced capabilities to fulfill this need. ML might be crucial in helping to boost output and expand the market for agricultural goods [5]. Insect pests and crop diseases are the biggest obstacles to agricultural productivity and the sustainable growth of the agricultural industry [6].

✉ Shitharth Selvarajan, ShitharthS@kdu.edu.et; Vijesh Kumar Patel, vijeshp.phd20.cs@nitp.ac.in; Kumar Abhishek, kumar.abhishek@nitp.ac.in | ¹Department of CSE, NIT Patna, Patna 800005, Bihar, India. ²Department of Computer Science, Cyber Security & Digital Forensics, School of Built Environment, Engineering and Computing, Leeds Beckett University, LS6 3QS Leeds, United Kingdom.



Farmers frequently turn to the indiscriminate application of synthetic fertilizers and plant protection agents to fulfill the increasing demand [7]. Plant disease and pest identification must be done quickly and accurately to avoid reduced agricultural output and/or decreased quantities of agricultural products. One possible strategy to get the answer is through ML techniques. In the current era, DL has significantly outperformed earlier methods in the development of image processing [8]. Agricultural contexts are among the many applications of ML, notwithstanding its specialized nature as an analytics system. This research aimed to develop a model for pest identification and classification to facilitate the process of recognizing and labeling the insects observed in the images [9].

The manual visual inspection approach, which involves skilled experts observing plants with unaided eyes, was still a widely used technique for detecting plant diseases. To achieve this, a sizable team of specialists and ongoing observation are needed. For impoverished farmers, this is expensive and time-consuming when their farm is huge [10]. However, in some nations, farmers lack the necessary tools or are unaware that they can seek advice from specialists. Plant disease identification by hand is a more time-consuming procedure that is prone to human error and has only been carried out in a few places [11]. The integration of ML in agriculture fields enables to identification of plant leaf diseases and accurately reporting such findings to the appropriate persons within appropriate ranges. The challenges that modern farmers face daily have made farming and agriculture unappealing to the masses. To live a safe life and escape such obstacles in the agricultural area, all young people were moving to modern cities [12].

Early diagnosis of crop pests and diseases in agriculture is crucial for mitigating potential damage to crops and optimizing yield. The proactive identification of anomalies in crop health allows timely intervention, enabling farmers to implement targeted and efficient pest and disease management strategies [13]. Leveraging Optimized RNNs in this context can significantly enhance the accuracy and efficiency of early diagnosis. These models are designed to be fine-tuned to address the specific challenges posed by agricultural data, facilitating improved decision-making in pest and disease control. The pressing need for efficient models to overcome existing challenges in crop pest and disease diagnosis underscores the significance of adopting advanced technologies [14]. Optimized RNNs, with their unique ability to combine DL capabilities with tailored optimizations for agricultural data intricacies, emerge as a compelling solution. Their potential to revolutionize early diagnosis practices in agriculture promises a more reliable and timely approach to pest and disease management, ushering in a new era of precision agriculture [15].

The foremost contributions of this paper are as follows:

- This paper focuses on developing an optimized RNN-based model for the early diagnosis of crop pests and diseases in agriculture.
- The paper offers a preprocessing unit that utilizes the Adaptive Bilateral Filter to lessen noise and improve the quality of the collected images. This step is essential for ensuring that the input data is clean and ready for further analysis.
- KEWO is proposed for efficient image segmentation. A reliable segmentation is critical for accurate feature extraction and subsequent analysis.
- The paper introduces the MFRNN as the core model for detecting crop diseases and pests which is the combination of the Moth flame-optimization (MFO) and Recurrent Neural Network (RNN). This choice of model suggests a focus on capturing temporal dependencies in the data, which can be crucial for identifying patterns indicative of pests and diseases.

The rest of the section is systematized as Sect. 2 of this study offers a literature review of the techniques that have been done previously on crop pest and disease detection in agriculture environments, and Sect. 3 offers the proposed methodology of the suggested model. The result and discussion of the paper are offered in Sects. 4, 5 offers a conclusion of the research study.

2 Literature review

2.1 This section evaluated some of the most recent studies on crop pests and disease detection in agriculture settings

In 2021, Wang, et. al. [16] proposed the ADSNN-BO model, which was built on an enhanced attention mechanism and a mobile net structure. In addition, the Bayesian optimization technique was used to modify the model's

hyperparameters. Cross-validated classification studies are performed using a four-category public rice disease dataset. The experimental results demonstrate that our mobile-friendly ADSNN-BO model achieves a test accuracy of 94.65%, outperforming all analyzed state-of-the-art models.

Rahman, et. al. [17] have discussed, that farmers could apply timely treatments to their rice plants and significantly lower their economic losses by accurately and promptly detecting diseases and pests. The accuracy of picture classification has significantly increased due to recent advancements in DL-based convolutional neural networks (CNN). This research developed DL-based methods for identifying illnesses and pests from images of rice plants, inspired by CNN's remarkable performance in image categorization. The accuracy of this existing model is 93.2%.

In 2023, Ahmed and Yadav [18] have discussed the "Plant Village" dataset, which contains 17 common ailments. There are four bacterial infections, two viral infections, two mold infections, and one ailment associated with mites on exhibit. For twelve crop species, there are also images of unharmed leaves. Prediction models were developed using the ML techniques of CNN, GLCMs, and support vector machines (SVMs). Artificial intelligence for classification has advanced along with the introduction of backpropagation neural networks. To diagnose disease, a K-mean clustering algorithm is also applied based on the collected real-time images of the leaves and the accuracy achieved is 92%.

In 2023, Islam, et. al. [19] suggested a DL-based technique for cotton leaf disease identification that makes use of existing TL algorithms' layers and parameters to be fine-tuned. In addition, has looked into how well other fine-tuning TL models including VGG-16, VGG-19, Inception-V3, and Exception performed for cotton disease prediction utilizing publically accessible datasets. According to the research, the Exception model has the highest accuracy rate (98.70%) and was chosen to create a web-based smart application that predicts cotton diseases in real time, helping farmers grow more cotton. As a result, the model would open up novel possibilities for the automatic identification of leaf illnesses in other plants and may correctly identify diseases affecting cotton leaves.

In 2024, Jessie, et. al. [20] suggested pre-processing the increased images of paddy leaves using a Contrast Limited Adaptive Histogram Equalization (CLAHE) method. The characteristics were taken out of an image of the paddy leaves that have already been processed using the GLCM model. The five disease classes of paddy leaves that were ultimately identified were Sheath Rot (SR), Bacterial Leaf Blight (BLB), Brown Spot (BS), Narrow Brown Leaf Spot (NBLS), and Leaf Smut (LS), utilizing the hybrid CNN approach. Platforms for MATLAB manage implementation tasks. The suggested model demonstrates the superiority of paddy leaf classification (93.5%) when compared to earlier methods.

Sourav and Wang [21] have created an intelligent model based on deep convolutional neural networks (DCNN) and transfer learning (TL) for the identification of jute pests to address this real-world issue. The proposed DCNN model can accurately and quickly identify jute pests automatically based on images. TL explicitly used the ImageNet resource to train the VGG19 CNN model. Furthermore, an organized image dataset of the four most prevalent jute bugs is produced. The increased accuracy rate (91.8%) and reliable indicators of other performance parameters support the model's validity for use in real-world scenarios. The suggested concept was integrated into iOS and Android apps to achieve the objectives of smart agriculture development.

Adil, Et. al. [22] have used an architecture based on convolutional neural networks to pinpoint the cause of leaf disease. To identify leaf conditions, several models including CNN, VGG-16, VGG-19, and ResNet-50 structures were used. By using images of the plant leaves, the suggested web tool seeks to help farmers diagnose plant illnesses. The suggested application classifies the current illness kind and distinguishes between healthy and infected leaves using the ResNet50 transfer learning model and achieved an accuracy rate of 94.2%. By early detection and adequate treatment, the aim is to assist farmers in conserving resources and averting financial loss.

In 2022, Liu, et. al. [23] suggested using IoT data to directly sense crop field ecological factors to anticipate ailment attack probability early on. The life cycles of plant diseases are significantly influenced by environmental conditions. Plant disease incidence was forecasted based on agricultural field environmental factors. Multiple Linear Regression (MLR) is used as the machine learning model when there is a linear relationship between disease attack and environmental variables. When crop field environmental parameters are based on the Internet of Things (IoT), the ML approach has more accurately predicted the existence of plant diseases. The effectiveness of the recommended remedy was (92.5%) evaluated by implementing the suggested model for tea plant blister blight prediction.

2.2 Problem statement

The crop pests and diseases in the agriculture sector affect substantial production and financial losses. Information on crop health and disease detection can enhance productivity and help control diseases through appropriate management

techniques including using fungicides, pesticides, or disease-specific chemicals to control vectors. Many farmers were stuck in a cycle of low production and poverty due to a lack of access to fundamental crop knowledge and instruction. This is due to the adoption of current technology and management methods being hampered by the knowledge and skill gap written by various authors which are shown in Table 1.

3 Proposed methodology

In agriculture, early detection of crop pests and diseases guarantees prompt intervention, averting extensive damage and maximizing resource use. Integrating Optimized RNN technology enhances accuracy in predicting threats, enabling farmers to adopt proactive measures. This approach not only improves crop health but also promotes sustainable and efficient farming practices, contributing to increased yields and reduced environmental impact.

Early Diagnosis of Crop Disease and Pests Utilizing an Optimized RNN model is developed by following the below-mentioned steps (i) Data Collection (ii) Image Augmentation (iii) Data Pre-processing (iv) Image Segmentation (v) Feature Extraction (vi) MFRNN-based Crop Diseases and Pests Detection. Figure 1 depicts the suggested model's overall structure.

3.1 Data collection

The CCMT Plant Disease Dataset is where the raw data was gathered. Leveraging Artificial Intelligence (AI) in agriculture is made possible in large part by the CCMT Dataset for crop pest and disease identification, especially when it comes to tackling issues that the agricultural industry faces in underdeveloped nations such as Ghana. The dataset, which includes 102,976 enhanced images and 24,881 raw images from nearby farms, focuses on four important crops: cashew, cassava, maize, and tomato. Each crop has a different class associated with pest and disease problems.

To ensure privacy and ethical use, the dataset is carefully divided into 22 classifications, certified by knowledgeable plant virologists, and de-identified. It encompasses a variety of settings and backgrounds, with different sizes including 400×400 , 487×1080 , 1080×518 , 3024×4032 , and 4032×3024 . The dataset is more resilient for real-world settings when a variety of backdrops, including white, dark, lighted, and genuine backgrounds, are included. Owing to its extensive scope, the CCMT Dataset not only bridges the knowledge gap between farmers and technology, but it also offers a useful tool for agricultural pest and disease detection researchers and practitioners using AI. The availability of the dataset to the scientific community may facilitate the creation of more precise and effective artificial intelligence models, which might enhance crop productivity, manage disease, and lower agricultural costs. The CCMT dataset and its corresponding classifications are displayed in Fig. 2 [24].

The collected raw data are moved on to the image augmentation stage.

3.2 Image augmentation

Image augmentation is the process of adding different changes to the original images in the context of agricultural pest and disease identification to enhance the size of the dataset and add diversity to the training data. Zooming, flipping, and rotations are used to obtain this augmentation.

3.2.1 Rotations

Rotations entail turning images at various angles, including 90, 180, or 270 degrees, to alter their orientation. This makes the model more resilient to changes in how features or objects are positioned within the images. Rotations improve the model's ability to generalize and perform well on unseen data with diverse orientations by exposing it to images from various angles.

Table 1 Reviews by various authors about the research gaps

Author name	Aim	Methodology	Advantage	Dataset	Parameters
Proposed	A machine learning (ML) technique is employed to more precisely identify diseases and pests on leaves and other crop parts	Kapur's Entropy-based Whale Optimization	The segmentation procedure becomes even more effective as it can identify and respond to the image with diverse intensities along with textures with the help of WOA	CCMT dataset	Accuracy-98.4% for cashews, 98.3% for cassava, 98.5% for maize, and 96.8% for tomato crops
Wang, et. al. [16]	To provide fast and precise illness detection with AI support, the ADSNN-BO model, which is based on mobile net structure and enhanced attention mechanism, has been proposed	• The model's hyper parameters were adjusted using the Bayesian optimization technique	• Using pesticide treatments is the primary strategy for managing plant diseases	Rice Diseases Dataset	Accuracy-95.6%, Precision-94.58%, recall-96.1%
Rahman, et. al. [17]	Farmers can significantly lower their financial losses by treating their rice plants promptly with the assistance of an accurate and timely identification of diseases and pests. The accuracy of image categorization has significantly increased due to recent advancements in DL-based CNN	• CNN • DL	• The classification of images using DL approaches has shown a lot of promise	Plant village dataset	Accuracy-93.2%, precision-92.5%, f1 score-92.7%, recall-93%
Ahmed and Yadav [18]	to demonstrate automated techniques for segmenting and classifying photos in real time in order to identify plant diseases. Neural network modification, SVM, CNN, ANN, and K-means segmentation are some of the best methods for identifying and categorizing ill plants	• ML • DL	• Low-intensity photos might be used to identify the damaged area; • the method can separate the affected leaves portion from a complex background	Plant Doc dataset	Accuracy-92%
Islam, et. al. [19]	To show our model's adaptability in a range of situations, resolve its shortcomings, and raise its precision and efficacy for practical cotton disease forecasting in agriculture	• DL-based method	• By adding non-linearity to the model, the ReLU function enables the model to recognize intricate patterns and representations in the data. Its advantage is that it requires less computing, as it consists simply of basic thresholding operations	Cotton plant disease dataset	Accuracy-94.7%, precision-93.8%, recall-95.2%, f1 score-94.5%

Table 1 (continued)

Author name	Aim	Methodology	Advantage	Dataset	Parameters
Jessie, et. al. [20]	Crop productivity can rise with early identification of leaf disease. With the aid of automated disease classification models, farmers can manage the spread of leaf disease in their fields	<ul style="list-style-type: none"> • CNN • DNN • DBN • RNN • LSTM 	<ul style="list-style-type: none"> • The core concept of RNNs is the application of sequential knowledge 	Paddy leaf disease dataset	Accuracy-93.5%, recall-94.0%, f1 score-93.5%
Sourav and Wang [21]	To address this real-world issue, an intelligent model based on transfer learning (TL) and DCNN was created for the identification of jute pests. Based on photos, the suggested DCNN model can provide quick and precise automatic identification of jute pests	<ul style="list-style-type: none"> • TL • DCNN 	<ul style="list-style-type: none"> • There is an urgent need for a thorough investigation into jute because environmental concerns are becoming more and more common 	Jute plant disease dataset	Accuracy-91.8%
Adil, Et. al. [22]	By examining images of the plant leaves, the suggested web tool seeks to help farmers diagnose plant illnesses. The suggested application classifies the current illness kind and distinguishes between healthy and infected leaves using the ResNet50 transfer learning model	<ul style="list-style-type: none"> • ML • SVM • DL • Image processing 	<ul style="list-style-type: none"> • The recommended web tool aims to assist farmers in diagnosing plant diseases by looking at pictures of the plant leaves. The proposed application uses the ResNet50 transfer learning model to classify the present sickness kind and differentiate between healthy and infected leaves 	Plant village dataset	Accuracy-94.2%, precision-93.4%, recall-94.8%
Liu, et. al. [23]	To put forth a machine learning (ML) strategy based on directly sensed crop field environmental data from the Internet of Things (IoT) for the early prediction of the likelihood of a disease attack. Environmental factors have a substantial correlation with the life cycles of plant diseases. Plant disease incidence is predicted by the environmental factors in crop fields	<ul style="list-style-type: none"> • IOT • ML 	<ul style="list-style-type: none"> • By adopting proactive measures to prevent illness attacks, early disease prediction helps to effectively control the condition 	Agriculture dataset	Accuracy-92.5%, precision-91.7%, recall-92.0%

Table 1 (continued)

Author name	Aim	Methodology	Advantage	Dataset	Parameters
Jayakumar, et. al. [27]	The purpose of this research is to create an autonomous illness prediction and classification system for melons utilizing a stacked Recurrent Neural Network (RNN)-based deep learning model	RNN	<ul style="list-style-type: none"> This methodology helps promote precision agriculture by giving farmers actionable insights that allow them to adopt targeted interventions according to early disease identification 	Plant disease dataset	Execution time
Albanese, et. al. [28]	The fundamental goal of this study is to establish automated pest identification utilizing Deep Neural Networks (DNN) installed on edge devices, especially for precision farming	DNN	<ul style="list-style-type: none"> DNN integration enables high accuracy in pest identification, contributing to sustainable agriculture practices by minimizing dependency on chemical pesticides via early and precise interventions 	Pest disease dataset	Accuracy-96.1%, recall-94.9%, f1 score-97.2%
Waleed, et. al. [29]	Artificial Neural Networks (ANN) will be used in this project to classify vegetative pests that harm cucumber plants	ANN	<ul style="list-style-type: none"> The automated nature of the system reduces human error with pest identification, assuring that actions are based on correct evaluations, which can lead to increased crop health and productivity 	Plant disease dataset	Accuracy-91.07%
Resti, et. al. [30]	The purpose of this study is to identify corn plant illnesses and pests using digital photos and a combination of Multinomial Naive Bayes as well as K-Nearest Neighbor (KNN) algorithms	KNN	<ul style="list-style-type: none"> The ability to use the characteristics of both categorization systems results in better accuracy and reliability in disease as well as pest identification 	Real time dataset	Accuracy-98.07%, precision-88.57%, f1 score-94.3%

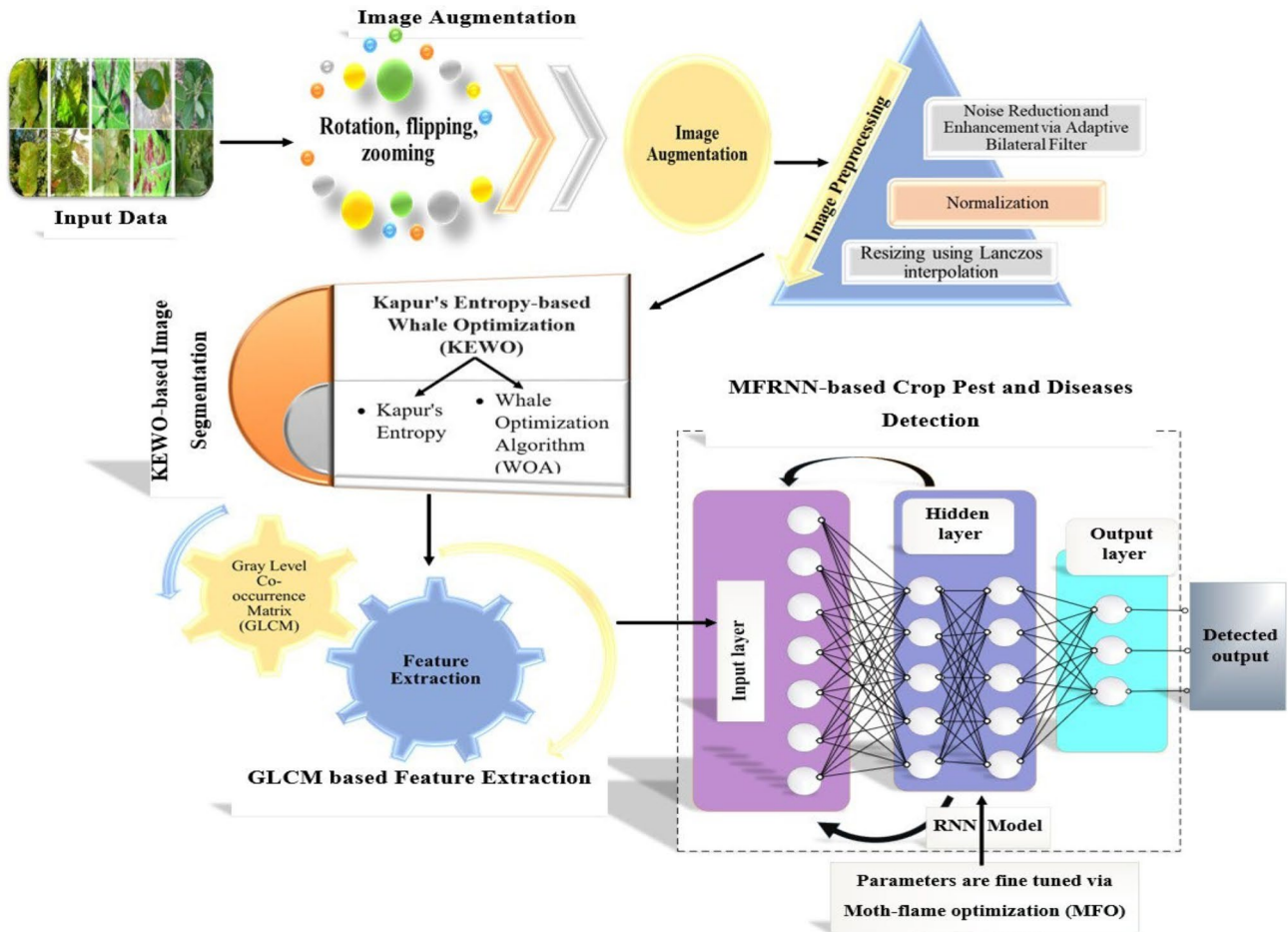


Fig. 1 Overall structure of the suggested model

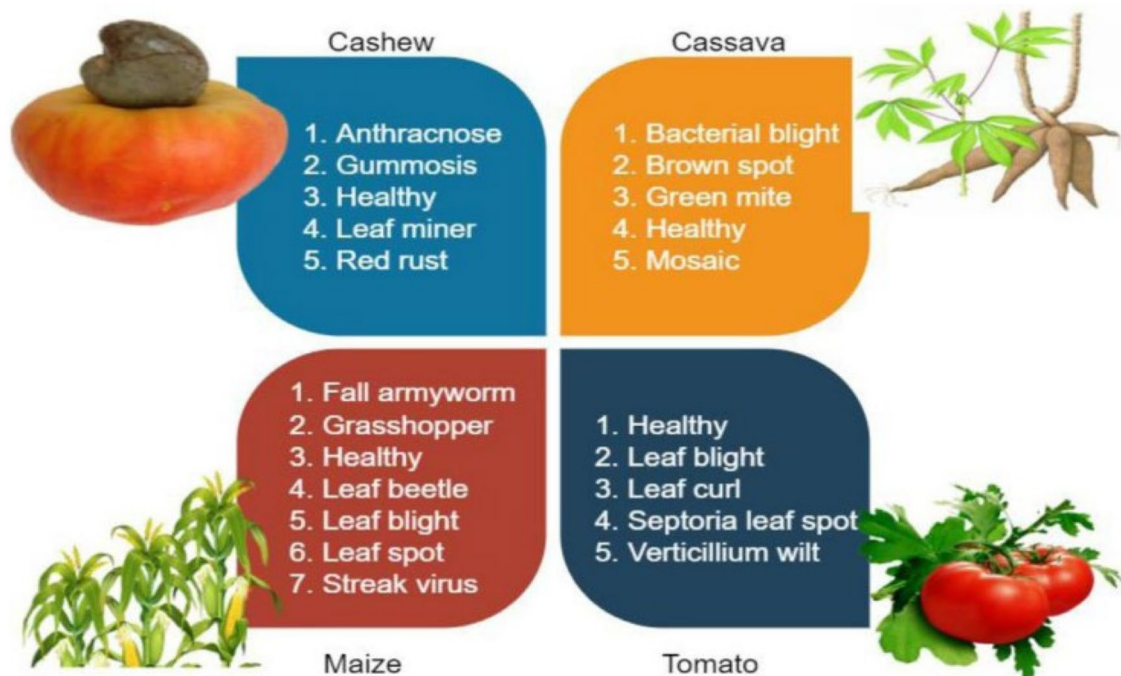


Fig. 2 CCMT dataset with its respective classes

3.2.2 Flipping

Mirroring the images vertically or horizontally is a part of flipping. Vertical flipping happens along the horizontal axis, whereas horizontal flipping entails flipping the image along its vertical axis. Flipping creates changes to the way items are arranged inside the images, which lessens the model's sensitivity to the precise orientation of features. This is particularly useful in situations where objects might show up with varying orientations in the real world [25].

3.2.3 Zooming

By zooming in to focus on certain elements or out to get a broader picture, one can adjust the scale of the images. By using this method, the model is better able to adapt to changes in the distances between the objects in the pictures and the camera. Zooming helps the model perform better overall on a variety of spatial configurations by enhancing its capacity to identify patterns and features at various scales. Figure 3 shows the image augmentation picture.

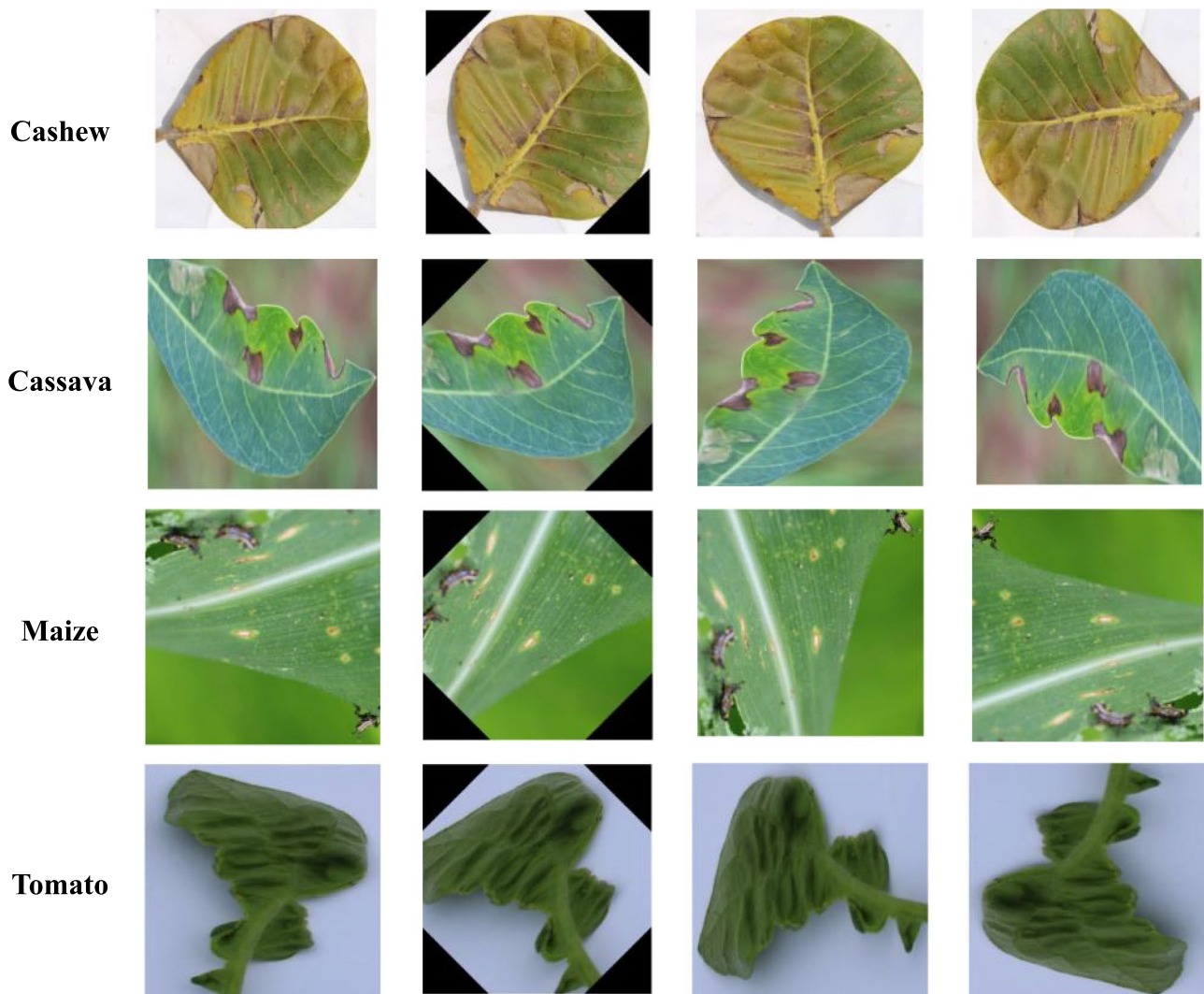


Fig. 3 Image augmentation from CCMT plant disease dataset

The dataset is successfully enhanced by implementing various augmentation strategies, increasing the diversity of training samples. After the data are enhanced, they proceed to the pre-processing phase.

3.3 Data pre-processing

The gathered raw data is carefully pre-processed to enhance its quality and usefulness for identifying crop diseases and pests. The procedure involves an adaptive bilateral filter to improve image quality and reduce noise, resizing using Lanczos interpolation, and normalization. Crop pest and disease identification can be done more effectively using a uniform format provided by the resized images, which are scaled to pixel values between 0 and 1.

3.3.1 Adaptive bilateral filter based noise reduction and enhancement

An essential part of the pre-processing phase of crop pest and disease detection is the Adaptive Bilateral Filter. The principal purposes of this filter are to reduce noise and improve image quality.

The purpose of the bilateral filter is smoothing; a damaged image's sharpness cannot be restored. We suggested a modified adaptive bilateral filter as a way to improve the bilateral filter's performance. The most common type of bilateral filter is the adaptive bilateral filter. The following significant adjustments were made to the bilateral filter in order to improve the denoised image's visual quality:

1. An offset ζ is introduced to the range filter.
2. Both ζ and σ_s i.e. the width of the range filter are made locally adaptive.
3. The color space utilized is CIE-Lab.

Assuming that $[n_0, o_0]$ represents the window's center pixel and that σ_e and σ_s represent the respective standard deviations of the domain and range Gaussian filters, Eq. (1) can be utilized to create the kernel weight function or normalization factor employed in the suggested method.

$$s_{n_0, o_0} = \sum_{n=n_0-O}^{n_0+O} \sum_{o=o_0-O}^{o_0+O} e^{\left(\frac{-(n-n_0)^2+(o-o_0)^2}{2\sigma_e^2}\right)} \times e^{\left(-\frac{\|(h[n, o]-h[n_0, o_0]-\zeta[n_0, o_0])\|^2}{2\sigma_s^2}\right)} \quad (1)$$

Through the adaptation of both ζ and σ_s and their joint optimization, the bilateral filter is enhanced to become a considerably more potent and adaptable filter. An exhaustive search is conducted in the parameter space $\phi = \{(\zeta, \sigma_s) : \zeta \in \phi_\zeta \& \sigma_s \in \phi_{\sigma_s}\}$, to identify the pair of parameters that minimizes the MSE.

where $\phi_{\sigma_s} = [5, 45]$ and $\phi_\zeta = [-60, 60]$. Empirically, the parameters' range and step size are selected to produce sufficient sharpening and smoothing for all kinds of images.

The strength of edges as assessed by a Gaussian Laplacian (LoG) operator, which is described as follows in Eqs. (2) and (3), is the feature that is used for pixel classification.

$$LoG[n, o] = \begin{cases} -\frac{1}{\pi\sigma_{LoG}^4} \left(1 - \frac{n^2+o^2}{2\sigma_{LoG}^2}\right) \exp\left(-\frac{n^2+o^2}{2\sigma_{LoG}^2}\right) - D, & |n|, |o| \leq O \\ 0, & \text{else} \end{cases} \quad (2)$$

$$\text{And } D = \frac{1}{(2O+1)^2} \sum_{n=n_0-O}^{n_0+O} \sum_{o=o_0-O}^{o_0+O} -\frac{1}{\pi\sigma_{LoG}^4} \left(1 - \frac{n^2+o^2}{2\sigma_{LoG}^2}\right) \exp\left(-\frac{n^2+o^2}{2\sigma_{LoG}^2}\right) \quad (3)$$

where $\sigma_{LoG} = 1.5$ and $O = 4$.

The following are characteristic attributes of the LoG operator [15]:

1. The filter is a high pass one.
2. It calculates the input image's second derivative.
3. Reaction: Near edges: a large reaction magnitude.

Smooth areas: the response's magnitude is little.

The response's magnitude is zero in the center of an edge.

The method used to compute the pixel class index is shown in Eq. (4)

$$M[n, o] = \begin{cases} [h_{LoG}[n, o]], & |h_{LoG}[n, o]| \leq M_{max} \\ M_{max}, & h_{LoG}[n, o] > M_{max} \\ -M_{max}, & h_{LoG}[n, o] < -M_{max} \end{cases} \quad (4)$$

where $h_{LoG}[n, o] = LoG[n, o] ** h[n, o]$, $M_{max} = 60$ and denotes arrounding $\lceil y \rceil$ to the nearest integer. ζ is roughly equivalent to the LoG class. The magnitude of the LoG reaction determines how much offset is wanted. Furthermore, the sign of ζ is identical to the LoG class's.

The breadth of the range filter σ_s in a simple bilateral filter is fixed. To improve the bilateral filter's strength and adaptability, offset ζ , as well as ζ and σ_s , are made locally adaptable. By carefully addressing noise, the filter contributes to refining the overall clarity and detail of the images, preparing them for subsequent stages of analysis.

3.3.2 Resizing

Lanczos interpolation is used in the pre-processing pipeline's essential step of resizing to modify the image's dimensions.

A mathematical procedure called the Lanczos interpolation function is used to interpolate a digital image's value smoothly between samples. Every sample of the provided image is mapped to a translated and scaled copy of the Lanczos kernel, which is a dilated sinc function with its central hump acting as a window. At the target pixel, the sum of these translated and scaled kernels is subsequently calculated. For geometric changes that do not require heavy down sampling, Lanczos interpolation offers the best qualities in terms of detail preservation and low aliasing artifact production. Equation (5) displays the Lanczos interpolation function of order n in one dimension.

$$M(y, o > 0) = \begin{cases} \sin c(y) \cdot \sin c(y/o) & \text{for } |y| \leq o \\ 0 & \text{otherwise} \end{cases} \quad (5)$$

where the normalized sinc function is presented in Eq. (6):









$$M(y, o > 0) = \begin{cases} 1 & \text{for } y = 0 \\ \text{sinc}(\pi y) / (\pi y) & \text{otherwise} \end{cases} \quad (6)$$

The use of Lanczos interpolation comes from its capacity to reduce artifacts and preserve image quality when resizing. By preserving the resized images' key characteristics and aesthetic integrity, this method makes for a more accurate and trustworthy analysis [26].

3.3.3 Normalization

Another essential component of the pre-processing process is normalization. It entails converting each image's pixel values to a uniform scale of 0 to 1. To enable consistent comparison and analysis of images, this normalization phase is essential for maintaining consistency in data representation. Variations in intensity are adjusted by scaling pixel values, which makes it possible to understand characteristics more reliably in later phases of crop pest and disease detection. After pre-processing, the processed data moves to the Image Segmentation stage. Figure 4 shows the input image and normalized image.

Fig. 4 Input and normalized image

Names	Input image	Normalized image
Cashew		
Cassava		
Maize		
Tomato		

3.4 Image segmentation through KEWO

The main aim of image segmentation is to precisely identify and isolate regions of interest, specifically focusing on areas affected by diseases or infestations, and distinguish them from the healthy sections within the images. In the segmentation process applied to the pre-processed data, the KEWO approach is utilized. This model integration enhances the accuracy of identifying and isolating relevant areas, ensuring a thorough and effective crop pest and disease detection process.

The purpose of proposing KEWO is to enhance the accuracy of crop disease and pest detection image segmentation along with improving efficiency. KEWO has been specifically selected to get maximum entropy at the time of segmentation so that the infected region separated in an image comes out to be much more diversified from the healthy region. This method provides an optimized thresholding technique that improves the quality of segmentation and leads toward a better extraction of features and improvement in overall detection accuracy.

3.4.1 Multilevel thresholding utilizing Kapur's entropy

Two primary types can be distinguished in image threshold segmentation: bi-level thresholding techniques and multilevel thresholding techniques. Using a threshold value, the bi-level thresholding approach divides a picture into a foreground and background to handle simple images. An essential unsupervised image processing technique that can handle complicated picture segmentation problems and produce better segmentation results is the multilevel thresholding method.

Kapur's entropy approach is a nonparametric threshold strategy that divides an image into many classes based on the entropy of the histogram; a higher entropy value indicates more homogeneous groups. Numerous academics have expressed interest in the suggested approach, which is better than existing thresholding-based techniques. The following are the special benefits of Kapur's entropy method: the minimum amount of computations needed, simple implementation, robust constancy, rapid processing speed, and excellent segmentation correctness. An image's entropy provides information on how distinct classes are compact and isolated from one another. Image segmentation has been achieved through the widespread application of Kapur's entropy, which efficiently finds the ideal threshold values by maximizing the objective function. Assume that the image is divided into many classes using n threshold values from the ideal threshold values $[u_1, u_2, \dots, u_o]$. the probability q_j is defined as follows in Eq. (7):

$$q_j = \frac{i_j}{\sum_{j=0}^{M-1} i(j)} \quad (7)$$

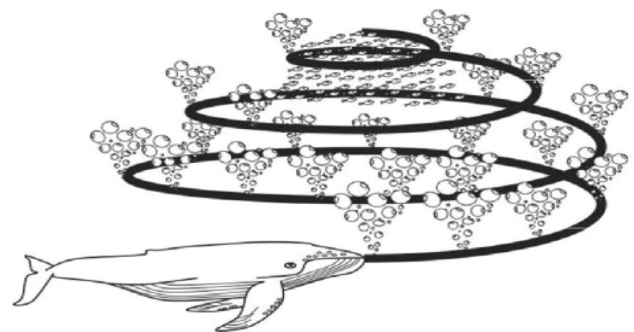
where i_j indicates the number of pixels, O the total number of pixels, M the number of levels, and j the grey level. Kapur's entropy is defined as shown in Eq. (8):

$$g(u_1, u_2, \dots, u_o) = H_0 + H_1 + H_2 + \dots + H_n \quad (8)$$

where,

$$H_0 = - \sum_{j=0}^{u_1-1} \frac{q_j}{\omega_0} \ln \frac{q_j}{\omega_0}, \omega_0 = \sum_{j=0}^{u_1-1} q_j \quad (9)$$

Fig. 5 Bubble-net feeding behavior of humpback whales



$$I_1 = - \sum_{j=u_1}^{u_2-1} \frac{q_j}{\omega_1} \ln \frac{q_j}{\omega_1}, \omega_1 = \sum_{j=u_1}^{u_2-1} q_j \quad (10)$$

$$I_2 = - \sum_{j=u_2}^{u_3-1} \frac{q_j}{\omega_2} \ln \frac{q_j}{\omega_2}, \omega_2 = \sum_{j=u_2}^{u_3-1} q_j \quad (11)$$

$$I_O = - \sum_{j=u_o}^{M-1} \frac{q_j}{\omega_O} \ln \frac{q_j}{\omega_O}, \omega_O = \sum_{j=u_o}^{M-1} q_j \quad (12)$$

The entropies of Kapur for each class are represented by I_1, I_2, \dots , and I_O is expressed in Eqs. (9)–(12), while the probabilities are represented by $\omega_0, \omega_1, \dots$, and ω_n .

3.4.2 WOA

To conduct a successful global search, the WOA is a state-of-the-art swarm intelligence optimization algorithm that essentially imitates the encirclement of the prey, the bubble-net assaulting strategy, and the hunt for prey. It is based on the way humpback whales assault nets with bubbles. Each humpback whale in the WOA is a possible remedy. The WOA search method is used to filter the candidate solutions to determine the global optimal solution. Figure 5 shows the bubble-net feeding behavior model.

A. Encircling prey

A humpback whale can locate its prey and rapidly close in on them. It is believed that the humpback whale's current location is either the intended prey or a less-than-ideal option because the location of the ideal solution is unknown. The other whales will adjust their positions based on the ideal position after the optimal humpback whale has been identified. One definition of the position update is expressed in Eqs. (13), (14),

$$\vec{E} = |\vec{D} \cdot \vec{Y}_{(u)}^* - \vec{Y}_{(u)}| \quad (13)$$

$$\vec{Y}_{(u+1)} = \vec{Y}_{(u)}^* - \vec{B} \cdot \vec{E} \quad (14)$$

where u is the iteration number now in progress, X^* is the position vector of the ideal solution, Y is the current position vector, $||$ is the absolute value, and \cdot is an elementwise multiplication. The definition of the coefficient vectors denoted by \vec{B} and \vec{E} can be found in Eqs. (15), (16):

$$\vec{B} = 2\vec{b} \cdot \vec{\bar{\zeta}} - \vec{b} \quad (15)$$

$$\vec{D} = 2 \cdot \vec{\bar{\zeta}} \quad (16)$$

where \vec{b} indicates a linear drop from 2 to 0 and $\vec{\bar{\zeta}}$ indicates a random vector in $[0, 1]$.

B. Bubble-net attacking technique (phase of exploitation)

The bubble-net assaulting method consists of the decreasing encircling technique and the spiral position updating. Using a random vector \vec{B} and a control variable b , the method used to reduce the gap between the current optimal location and the global optimal position is the shrinking encircling mechanism. The updated spiral position

guides the humpback whales as they swim in the direction of their prey, and they determine the distance between their location and the prey's position to successfully catch them. One definition of the position update is illustrated in Eqs. (17), (18).

$$\vec{E}^i = |\vec{Y}_{(u)}^* - \vec{Y}_{(u)}| \quad (17)$$

$$\vec{Y}_{(u+1)}^i = \vec{Y}_{(u)}^i \cdot f^{cm} \cos(2\pi m) + \vec{Y}_{(u)}^* \quad (18)$$

where m represents a random value in $[-1, 1]$, c indicates a constant for defining the shape of the logarithmic spiral, and \vec{E}^i indicates the distance between whale and prey.

where \vec{E}^i is the distance between the whale and its prey, m is a random value in $[-1, 1]$, and c is a constant that defines the shape of the logarithmic spiral. When humpback whales update their positions to intercept prey, there is a 50% chance that they will use the logarithmic spiral position updating method or the diminishing encircling mechanism. One definition of the position update is shown in Eq. (19):

$$\vec{Y}(y+1) = \begin{cases} \vec{Y}_{(u)}^* - \vec{B} \cdot \vec{D} & \text{if } q < 0.5 \\ \vec{E}^i \cdot f^{cm} \cdot \cos(2\pi m) + \vec{Y}_{(u)}^* & \text{if } q > 0.5 \end{cases} \quad (19)$$

wherein q represents a random number within the range $[0, 1]$.

C. Encircling prey

To avoid slipping into the local optimum, the WOA modifies the vector \vec{B} . In a random search strategy to find the prey. The WOA has a great exploration capacity to find the global optimal solution if $|\vec{B}| > 1$. The definition of the position update is expressed in Eq. (20) and (21):

$$\vec{E} = |\vec{D} \cdot \vec{Y}_{rand} - \vec{Y}| \quad (20)$$

$$\vec{Y}_{(u+1)} = \vec{Y}_{rand} - \vec{B} \cdot \vec{E} \quad (21)$$

where a random position vector (a random whale) chosen from the present population is indicated by the symbol \vec{Y}_{rand} .

In crop disease and pest detection, Kapur's entropy is employed for image segmentation, proving well-suited for delineating regions of interest in agricultural images where manifestations of pests or diseases may occur. However, utilizing Kapur's entropy alone in pest and disease detection may encounter challenges related to local optima and difficulties in handling the complex and varied visual characteristics of crops. To address these challenges, the WOA is integrated. Its global optimization capabilities efficiently explore the solution space, enhancing adaptability to diverse crop conditions. By combining Kapur's entropy with WOA, the segmentation process becomes more robust and adept at handling the intricate visual patterns associated with crop health. With its global optimization capabilities and automatic parameter adjustment, this integrated approach significantly improves agricultural pest and disease detection systems' accuracy and dependability. The threshold value of specific image segmentation is shown by the position of each whale or search agent. By adjusting its location, a whale sets the threshold level to achieve the best outcome. Figure 3 displays the WOA flowchart for Kapur's Entropy.

Algorithm 1 KEWO-based Image Segmentation

Start**Step 1.** Set the whale population to start.**Step 2.** Using the Kapur-based technique, determine each search agent's fitness using Equation (8).Find the finest search agent. Y^* **Step 3. While** ($u < u_{\max}$)**do****for** each search agentUpdate $b, B, D, m,$ and q **If1**($q < 0.5$)**If2**($|B| < 1$)

Update the search agent's position using Equation (14)

else if2 ($|B| \geq 1$)Select a random search agent (Y_{rand})

change the search agent's position using Equation (21)

end if2**else if1** ($q \geq 0.5$)

update the search agent's position using Equation (18)

end if1**end for**

Verify and adjust any search agents that cross the search space.

Using the Kapur-based technique, determine each search agent's fitness using Equation (8).

Update Y^* If an improved option exists

$$u = u + 1$$

end whileyield the best search agent Y^* , which represents the ideal segmentation threshold values.**End**

Fig. 6 GLCM directions for extracting texture features

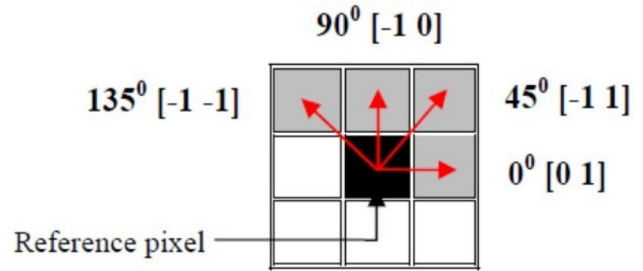
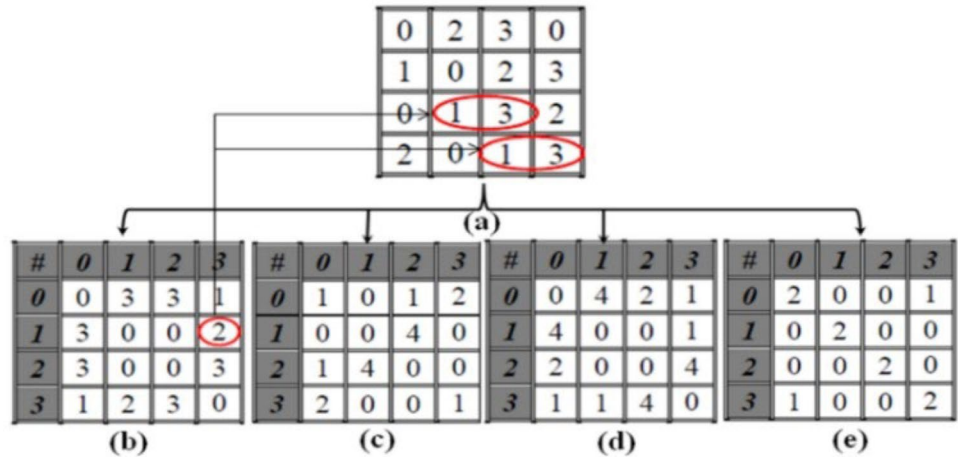


Fig. 7 GLCM [31] Construction based **a** test picture in four potential orientations **b** 0° **c** 45° **d** 90° and **e** 135° with a distance $f = 1$



3.5 Feature extraction via GLCM

To extract the best features from the feature-segmented data, a GLCM is utilized.

Gray-Level Co-occurrence Matrix (GLCM) a square matrix, contains some features related to a texture image’s gray-level distribution in space. The intensity value of m , also known as the neighbor pixel, and a pixel value with the intensity value of l , also known as the reference pixels, are shown about each other in terms of frequency. Each component (l, m) Of the matrix is the number of occurrences of a pair of pixels with values of l and m that are apart by a distance of d . The spatial relationship between two adjacent pixels can be specified in several ways, including different offsets and angles. A pixel is in default relationship with its immediate neighbor when it is to the right of it. Four likely spatial associations ($0^\circ; 45^\circ; 90^\circ$ and 135°) Were identified, described, and put into practice.

The elements of a $J \times J$ GLCM MCO for a displacement vector $d(= da, db)$ for an assumed image L of size $N \times N$ are distinct mathematically as shown in Eq. (22),

$$P_{ER} = \sum_{a=1}^N \sum_{a=1}^N \left\{ \begin{array}{l} 1, \text{if } L(a, b) = l \text{ and } L(a + g_a, b + g_b) = m \\ 0, \text{otherwise} \end{array} \right\} \tag{22}$$

Figure 6 shows how to create four symmetrical co-occurrence matrices from a 4×4 image that has four grayscale values ranging from 0 to 3. To do this, we took into account one nearby pixel ($f = 1$) along each of the four potential directions, as shown by Eq. (23),

$$\{[01] \text{ for } 0^\circ, [-11] \text{ for } 45^\circ, [-10] \text{ for } 90^\circ, \text{ and } [-1 - 1] \text{ for } 135^\circ\} \tag{23}$$

In the GLCM, each element denotes the frequency of two grayscale pixels, l and m , being adjacent in both directions (θ) and distance (f). For the 0° , Co-occurrence matrix, there are two instances of the pixel intensity values 1 and 3 adjacent to one another in the input image. In addition, there are two examples of adjacent pixel strengths of 3 and 1. Because of this, The symmetric nature of these vectors and the co-occurrence pairing that results from setting θ equal to 0° would resemble those that result from setting θ equal to 180° . This idea also applies to angles of $45^\circ, 90^\circ,$ and 135° . The GLCM is calculated for each of the four possible angles while accounting for all of these variables, and it is shown in Fig. 7 below.

stands for the number of repetitions [28]. The feature-extracted data moved on to the crop pest and disease detection stage to detect crop diseases and pests.

The advantage of employing the Gray-Level Co-occurrence Matrix (GLCM) over other methodologies is its increased capacity to collect detailed texture features, which is critical for diagnosing and classifying crop diseases and pests. GLCM provides crucial textural properties such as contrast, correlation, energy, and homogeneity, allowing for more accurate separation between healthy and infected regions.

3.6 MFRNN-based crop pest and diseases detection

From the feature-extracted data, the Crop Pests and Diseases are detected using MFRNN. The proposed MFRNN framework is a combination of the MFO and RNN. The RNN is optimized through MFO algorithm.

3.7 Optimizing RNN using MFO

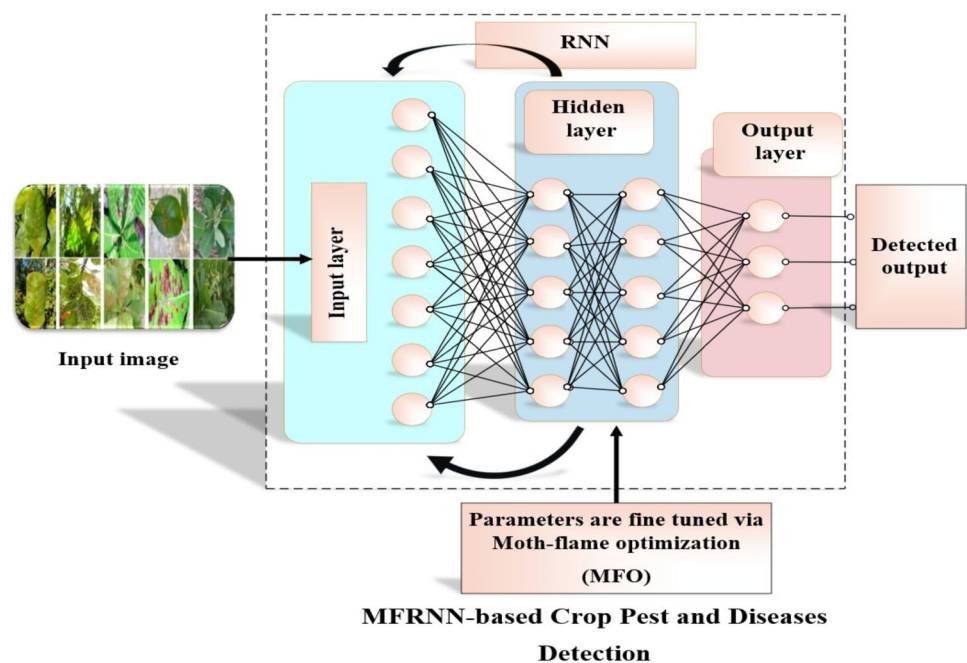
The structure of the suggested MFRNN model is illustrated in Fig. 8. MFO fine-tunes the RNN for enhanced performance. This algorithm optimizes RNN parameters, ensuring better accuracy and efficiency in processing sequential data. By utilizing MFO, the RNN is dynamically adjusted to improve its ability to capture patterns and nuances, resulting in a more effective and finely-tuned model for various applications, including crop pest and diseases detection.

The choice of using RNN in crop disease and pest diagnosis is informed by the fact that RNN is capable of capturing temporal and sequential data which is paramount in this case. However, image data is not sequential; when used together with time series data such as disease progression or environmental conditions that affect crops, RNNs can adequately capture the correlation between consecutive observations.

RNNs are particularly useful in continuous monitoring and hence can be applied to real-time crop disease detection systems in agricultural environments where conditions vary. The integration of RNN with feature extraction techniques improves the system's general performance in diagnosing crop diseases and pests.

An artificial neural network type called an RNN uses feedback connections between nodes to process data sequentially. Three main layers make up RNNs, as opposed to conventional feedforward neural networks: input, hidden, and output. Recurrent connections found in the hidden layer are essential for preserving a hidden state that records details about previous inputs in a series. RNNs are well-suited for tasks involving time-series data, natural language processing, and other applications where temporal dependencies are important because of their capacity to maintain recollection of previous inputs.

Fig. 8 MFRNN Model



The study uses an RNN for real-time illness and pest detection in dynamic systems, which has fixed coefficients in the hidden neurons and self-feedback connections. The RNN's architecture, comprising input, hidden, and output layers, enhances its memorization capacity, leading to improved convergence precision and reduced learning time.

The RNN is used after extracting relevant characteristics from the images using the GLCM. Instead of directly processing raw pixel data, the RNN works with extracted feature vectors, which contain essential texture and spatial correlations in images. The RNN does not work directly on the images; instead, it processes the meaningful, lower-dimensional feature representations obtained from them, allowing it to contribute to accurate diagnosis without requiring direct image input. This hybrid technique ensures that RNN's capabilities are maximized for image-based crop disease diagnosis.

3.7.1 layer 1: Input layer

Equation (24) illustrates how the node input and output are represented in the input layer.

$$z_k^K(m) = h(\text{net}_k^K(m)) = \text{net}_k^K(m) = w(m)k = 1, 2, \dots, n \quad (24)$$

where w and z_k^K are the layer's input and output, respectively, and m denotes the m – th iteration.

3.7.2 layer 2: Hidden layer

The node input and output in the hidden layer are described by Eqs. (25) and (26).

$$z_l^J(m) = U(\text{net}_l^J(m)) \quad (25)$$

$$\text{net}_l^J = \sum_k Y_{lk}^{JK}(m) \times z_k^K(m) + \sum_t Y_{lt}^{JE}(m) \times z_t^E(m) \quad (26)$$

$$l = 1, 2, \dots, o, \quad o_t = 1, 2, \dots, o$$

where net_l^J and $z_k^K(m)$ are the input and output of the hidden layer respectively,

The context layer's output is represented by z_t^E , and the weight matrices connecting the input layer and the hidden layer are $Y^{JK} \in T^{o \times n}$ and $Y^{JE} \in T^{o \times o}$, respectively and the sigmoid function is $U(z)$ is expressed in Eq. (27).

$$U(z) = 1 / (1 + e^{-z}) \quad (27)$$

3.7.3 Context layer

The node input and output are represented in the context layer as indicated by Eq. (28).

$$z_t^E(m) = \alpha z_t^E(m-1) + z_l^J(m-1) \quad (28)$$

where $0 \leq \alpha < 1$ is the expression for the self-feedback connection coefficient. When the coefficient is zero, the modified Elman network is identical to the original Elman network.

3.7.4 layer 3: Output layer

Equation (29) and (30) illustrate how the node input and output are represented in the output layer.

$$a_s(m) = i(\text{net}_s^Q(m)) \quad s = 1, 2, \dots, p \quad (29)$$

$$\text{net}_s^Q(m) = \sum_l Z_{sl}^{QJ}(m) \times z_l^J(m) \quad (30)$$

where net_s^O and $a_s(m)$ are the input and the output of the output layer, respectively, Y^{OJ} is the weight matrix between the hidden layer and the output layer, and $i(z)$ is often taken as a linear function.

Choosing the network's parameters is crucial while building an RNN. The MFO technique is utilized in this research to adjust many parameters of the RNN, including Learning Rates, Weight Matrices (Y^{JK} , Y^{JE} , and Y^{OJ}) and Self-feedback Connecting Coefficient (α) of the RNN.

3.7.5 Hyperparameter tuning

Lastly, the MFRNN method's hyperparameters are adjusted using the MFO methodology. The population-based metaheuristic method known as MFO was inspired by the sloping migrations of moths. The MFO technique functions similarly to flames and moths, with the problem variable being indicated by the moth's position in the search space and the moth representing the answer. In this instance, the moth designates search mediators that make an effort to locate a timetable within the search space. The moth group notifies the position when searching various locations inside the search range. In the meantime, the frames show each moth's ideal location. The following moths and fames are simulated by the MFO approach using four arrays.

The goal of the 2D array N is to save the solution. The first dimension of the array can be represented by the number of moths (o), and the second dimension can be represented by the number of variables in the problem (e).

The goal of a 1D array called PN is to preserve each moth's unique fitness value. The goal of the 2D array G is to preserve the fames that resemble the N array. The goal of the 1D array PG is to maintain the matched fitness rate for every optimal location. Three key elements comprise the MFO method: initiation (J), search process (Q), and termination (U). Initially, a random trial of the work was formed by the random series of jobs, and the population of the solution was randomly generated using Eq. (31).

$$N(j) = \text{Random Sequence of Jobs}(0, e - 1) \quad (31)$$

where the moth objective function denoted by PN is equivalent to the fitness function (PN = fitness function (N)). After d, the number of jobs, the makespan is calculated using the moth's fitness function.

Iteratively looking for the moth solution continues in the local search (also known as the search procedure) stage until the stopping condition (U) or predetermined termination is satisfied. The least valuable solution was given back at the termination point.

However, the moth's position was improved with the application of $N_j = T(N_j, G_k)$, wherein M_i depicts the moth, j and k stand for indices, T signifies the spiral process, and G_k stands for notoriety. The mathematical expression for this is shown in Eq. (32),

$$T(N_j, G_k) = E_j * f^{cu} * \cos(2\pi u) + G_k \quad (32)$$

where c displays a logarithmic spiral in its constant form. E_j explained how far the moth was from stardom, $E_j = |G_k - N_j|$, and u denotes the value that was chosen at random between s and 1. The s value decreased linearly from -1 to -2 .

The MFO was designed to handle ongoing issues. However, it went further by proposing employment relocations about the discrete optimization problem. The moth is prompted to approach the famous by the name T . The jobs that could migrate are denoted by the letter T . In some cases, its worth may beyond the boundaries of the issue. The shift in this instance may be directed toward the arbitrary job. In contrast, the following Eq. (33) was used to determine the Flame_no, or fame number:

$$\text{Flame}_{no} = \text{round}\left(O - m * \frac{O - 1}{U}\right) \quad (33)$$

where O represents the greatest fame number, U indicates the number of iterations, and m indicates the number of iterations that already exist.

A fitness function (FF) that achieves better classifier results was created by the MFO method. It resolves a positive integer, indicating that the pest and disease detection is good. Here, Eq. (34) provides the minimal classifier rate of errors, which is FF.

$$fitness(y_j) = ClassificationErrorRate(y_j) = \frac{Number\ of\ misclassified\ samples}{Total\ number\ of\ samples} * 100 \quad (34)$$

In this paper, the MFO algorithm is strategically employed to fine-tune crucial parameters essential for optimizing the performance of RNNs in the realm of crop pest and disease detection. The parameters subject to fine-tuning include Learning Rates, which significantly impact the convergence speed and optimization efficiency of the neural network. Additionally, Weight Matrices, responsible for determining the strength of connections between different layers, are dynamically adjusted to enhance the network's ability to capture intricate patterns. Furthermore, the Self-feedback Connecting Coefficient in the context layer is optimized through MFO, influencing the network's capacity to retain valuable information about past inputs. This comprehensive approach ensures that the RNN's configuration is finely tuned, optimizing its learning process and overall efficiency in recognizing patterns associated with crop health. Ultimately, this contributes to improved accuracy in pest and disease detection. After fine-tuning the parameters of the RNN, the outcome is acquired from the output layer of the network. Algorithm 2 details the MFO Optimization for RNN hyperparameter tuning.

Algorithm 2 MFO Optimization for RNN hyperparameter tuning

```

Initialize the parameters for moth flame
Initialize Moth position  $N_j$  randomly
For  $j=1$  to  $o$  do
  Calculate the fitness function  $y_j$ 
end for
While  $iteration \leq max\_iterations$  do
  Update the position of  $N_j$ 
  Evaluate the fitness function  $y_j$ 
  If  $iteration == 1$  then
     $F = sort(N)$  and  $PG = sort(N_{u-1}, N_u)$ 
  end if
  for  $j=1$  to  $o$  do
    for  $k=1$  to  $e$  do
      Update the value of  $s$  and  $u$ 
      Calculate the value of  $E$  respect to its corresponding moth
      Update  $M(j, k)$  respect to its corresponding moth
    end for
  end for
end while

```

4 Result and discussion

In this subsection, the results and discussion of the suggested framework are presented. This paper introduces a MFRNN model, consisting of MFO and RNN components. Various metrics such as sensitivity, accuracy, precision, F- Measure, MCC and Confusion matrix have been considered to assess the performance effectiveness of the recommended technique.

4.1 Dataset description and experimental setup

In general, the CCMT Plant Disease Dataset is a database of high-resolution images of various plants that are suffering from diseases. Every image captures the visual symptoms on different parts of the plant such as leaves, stems, or fruits in respect to various diseases they suffered from. Images are labeled along with disease types that include various subcategories of the disease. In most cases, information will include the full name of the disease and the plant species affected. Ancillary metadata accompanying the dataset would include information on the particular species of the plant, the stage of the disease, the geographic location of the sample origin, and the conditions under which the sample environment exists. The dataset encompasses a variety of plant species, including crops: tomatoes, potatoes, corn, and ornamental plants. Diseases fall into several classes, including bacterial, fungal, viral, and parasitic diseases. Specific diseases are then further subdivided by classes bearing their distinct symptoms.

The recommended framework has been implemented in MATLAB. The suggested framework has been evaluated utilizing the CCMT Plant Disease Dataset (<https://www.kaggle.com/datasets/akshajsinghbisht/ccmt-augmented-split>) [Accessed Date: 2024-01-31]. 70% of the collected data has been utilized for training and 30% for testing. This assessment considered various metrics like sensitivity, accuracy, precision, F-Measure MCC and Confusion matrix.

4.2 Parameters of performance analysis

The suggested model's performance was assessed utilizing the following metrics as accuracy, precision, F- Measure, sensitivity, MCC and Confusion matrix.

(i) Accuracy

The degree to which measurements of a quantity are closest to that quantity's actual (true) value is measured by accuracy. Equation (35) is the mathematical equivalent of the accuracy formula.

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN} \quad (35)$$

(ii) Precision

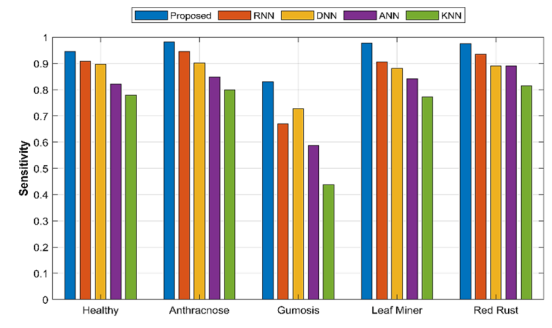
Precision lies on its ability to distinguish genuine positives from anticipated positives. It calculates the percentage of all positive forecasts that were accurately forecasted. We may see the precision formula in Eq. (36).

$$Precision = \frac{TP}{TP + FP} \quad (36)$$

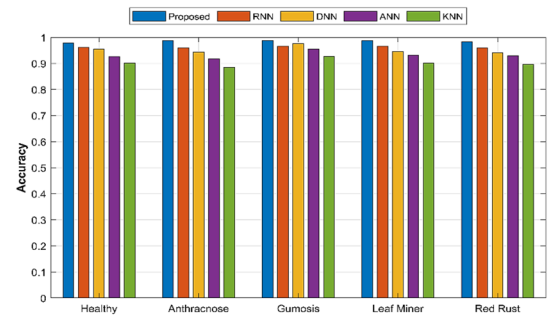
Table 2 Comparative analysis of performance metrics with existing models

Cashew					
Classification Models	Sensitivity	Accuracy	Precision	F-Measure	MCC
Proposed	0.9611	0.9844	0.9611	0.9611	0.9513
RNN [27]	0.9061	0.9624	0.9061	0.9061	0.8826
DNN [28]	0.8802	0.9521	0.8802	0.8802	0.8502
ANN [29]	0.8298	0.9319	0.8298	0.8298	0.7872
KNN [30]	0.7557	0.9023	0.7557	0.7557	0.6947

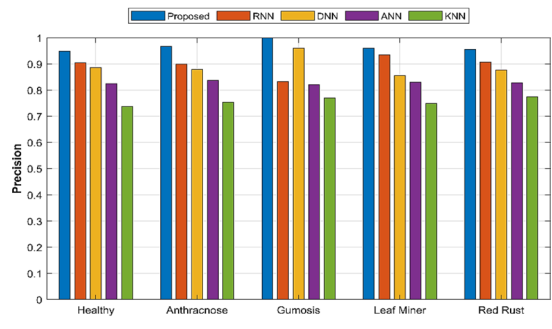
Fig. 9 a, b, c, d, e: graphical representation of the performance analysis: for cashew



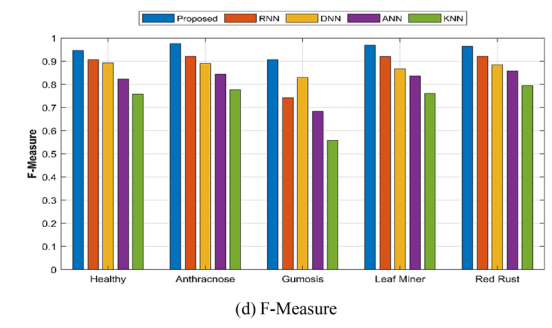
(a) Sensitivity



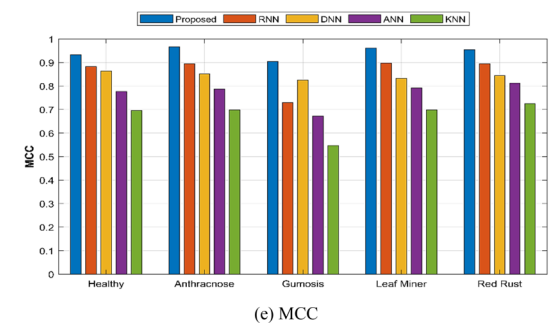
(b) Accuracy



(c) Precision



(d) F-Measure



(e) MCC

Table 3 Comparative analysis of performance metrics with existing models for cassava

Cassava					
Classification models	Sensitivity	Accuracy	Precision	F-measure	MCC
Proposed	0.9594	0.9838	0.9594	0.9594	0.9492
RNN	0.9261	0.9704	0.9261	0.9261	0.9076
DNN	0.9008	0.9603	0.9008	0.9008	0.876
ANN	0.8489	0.9395	0.8489	0.8489	0.8111
KNN	0.7929	0.9172	0.7929	0.7929	0.7412

v) F-Measure

The F-Measure number is a tradeoff between fully identifying every data bit and guaranteeing that each definition correctly recognizes only one type of information item. The F-Measure's mathematical formula is given in Eq. (37).

$$F_Score = \frac{Precision \cdot Recall}{Precision + Recall} \quad (37)$$

ix) MCC

MCC is a trustworthy statistic for evaluating the effectiveness of binary classifiers since it takes into account TP, TN, FN, and FP. In actuality, MCC quantifies the degree of correlation between the predictor and the labels. A mathematical representation of the MCC formula can be found in Eq. (38).

$$MCC = \frac{(TP * TN) - (FP * FN)}{\sqrt{(TP + FP)(TP + FN)(FP + TN)(TN + FN)}} \quad (38)$$

x) Sensitivity

The sensitivity value is calculated by dividing the total positives by the percentage of true positive predictions. A mathematical representation is shown in Eq. (39).

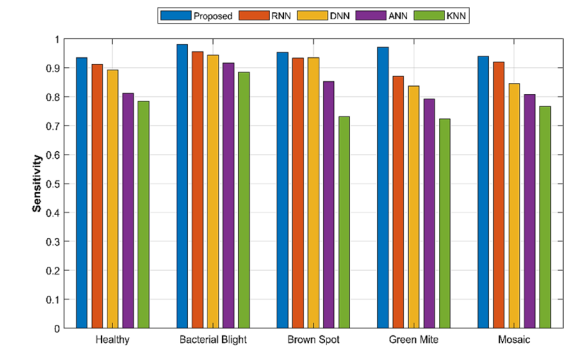
$$Sensitivity = \frac{TP}{TP + FN} \quad (39)$$

4.3 Overall performance analysis: for Cashew

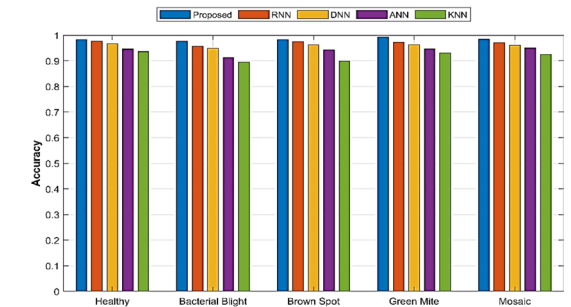
In this section, overall performance analysis is presented for cashew, considering both proposed model and other existing models. Table 2 displays a comparative analysis of the presented framework and existing models based on performance metrics.

In the context of cashew agriculture, the proposed model exhibits exceptional performance compared to existing models. The proposed model achieves a sensitivity of 96.11%, outperforming all other models, including the RNN, which has the second-highest sensitivity at 90.61%. This high sensitivity underscores the proposed model's effectiveness in accurately identifying true positive instances of cashew crop issues. The overall accuracy of the proposed model stands at 98.44%, indicating its ability to correctly classify instances across all classes. This surpasses the accuracy of the RNN [27], DNN [28], ANN [29], and KNN [30] models, highlighting the proposed model's robustness and reliability in the context of cashew agriculture. With a precision of 96.11%, the suggested model has an exceptionally high level of false positive avoidance. This suggests that the model is more likely to be accurate than any other model when it forecasts a positive outcome, outperforming all other models in terms of precision. The F-Measure, a metric that balances precision and recall, is consistently high for the proposed model at 96.11%. This signifies a balanced performance in terms of both precision and sensitivity, showcasing the model's efficacy in cashew crop issue diagnosis. The Matthews Correlation Coefficient (MCC), a comprehensive metric considering true and false positives and negatives, attests to the proposed model's overall excellence with an MCC of 95.13%. This score surpasses all other models, reinforcing the proposed model's suitability for reliable and efficient cashew crop issue management. In summary, the proposed Optimized Recurrent Neural Network-based model excels across all key metrics, including sensitivity, accuracy, precision, F-Measure, sensitivity MCC, and confusion matrix demonstrating its superiority in early diagnosis and management of cashew crop issues. Its heightened performance positions it as a valuable tool for precision agriculture, enabling timely interventions and

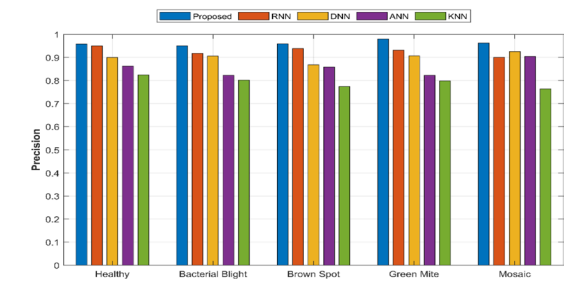
Fig. 10 a, b, c, d, e: shows the graphical representation of the performance analysis: for Cassava



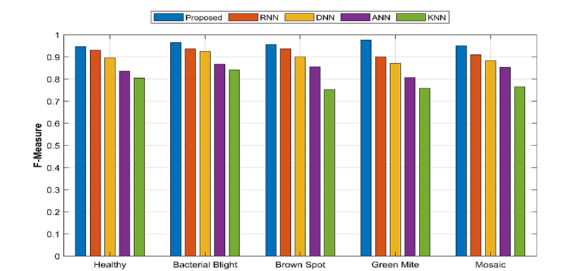
(a) Sensitivity



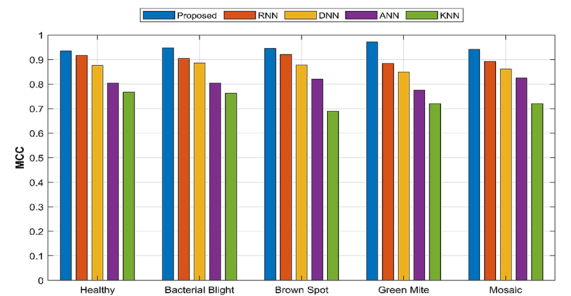
(b) Accuracy



(c) Precision



(d) F-Measure



(e) MCC

Table 4 Comparative analysis of performance metrics with existing models for maize

Maize					
Classification models	Sensitivity	Accuracy	Precision	F-measure	MCC
Proposed	0.9474	0.985	0.9474	0.9474	0.9387
RNN	0.9033	0.9724	0.9033	0.9033	0.8872
DNN	0.8563	0.959	0.8563	0.8563	0.8324
ANN	0.7906	0.9402	0.7906	0.7906	0.7557
KNN	0.6995	0.9142	0.6995	0.6995	0.6495

improved overall crop health in the cashew farming sector. The graphical depiction of the cashew performance analysis is displayed in Fig. 9(a), (b), (c), (d), (e).

4.4 Overall performance analysis: for Cassava

In this section, we present a comprehensive performance analysis for cashew, comparing the proposed model with existing ones. Table 3 illustrates a comparative assessment based on various performance metrics for cassava.

The proposed model for early diagnosis of crop pest and diseases in cassava agriculture demonstrates superior performance compared to existing models. The evaluation metrics, including Sensitivity, Accuracy, Precision, F-Measure, and MCC, provide a comprehensive insight into the model's effectiveness. The sensitivity of the proposed model stands out at 95.94%, indicating its high ability to correctly identify true positive instances of crop pest and diseases in cassava. This surpasses all other models, including the RNN, which has the second-highest sensitivity at 92.61%. The higher sensitivity of the proposed model implies a reduced likelihood of false negatives, which is crucial in agricultural applications where early detection of pests and diseases is essential for timely intervention. In terms of overall accuracy, the proposed model achieves an impressive 98.38%, showcasing its capability to correctly classify instances across all classes. This significantly outperforms the RNN, DNN, ANN, and KNN models, reinforcing the proposed model's robustness and reliability. With a precision of 95.94%, the suggested model also has a very good capacity to prevent false positives. This implies that the model has a high probability of being right when it predicts a positive case. The proposed framework surpasses all other models in this regard, emphasizing its precision in early diagnosis. The F-Measure, which balances precision and recall, is consistently high for the proposed model at 95.94%. This indicates a harmonious blend of precision and sensitivity, further solidifying its effectiveness in crop pest and disease diagnosis. The suggested model's overall performance is reflected in the MCC, a statistic that accounts for true and false positives as well as negatives. With an MCC of 94.92%, it outshines all other models, including the RNN, which has the second-highest MCC at 90.76%. Figure 10a, b, c, d, e shows the graphical representation of the performance analysis: for cassava.

4.5 Overall performance analysis: for maize

This section provides an in-depth performance analysis for cashews, juxtaposing the proposed model with existing ones. Table 4 delineates a comparative evaluation using various performance metrics for maize.

The proposed classification model for maize outperforms existing models across various evaluation metrics. With a sensitivity of 0.9474, the proposed model excels in correctly identifying positive instances (maize), surpassing the performance of the RNN, DNN, ANN, and KNN models. Its accuracy, at 0.985, demonstrates its overall effectiveness in maize classification, outshining all other models. Precision, a measure of minimizing false positives, is highest in the proposed model at 0.9474. The F-Measure, which balances precision and recall, also attains the highest value of 0.9474 in the proposed model. Furthermore, the MCC stands at 0.9387, indicating a robust correlation between predicted and actual classifications. In contrast, the existing models, including RNN, DNN, ANN, and KNN, exhibit slightly lower values across these metrics. While they perform reasonably well, there is a noticeable trade-off between precision and recall, leading to lower F-measure values. The proposed model's consistent superiority across sensitivity, precision, accuracy, F-measure, and MCC positions it as the most promising choice for maize classification compared to the alternatives. Overall, the proposed model demonstrates a compelling advantage in accuracy and reliability for pest and disease identification in maize. Figure 11a, b, c, d, e shows the graphical representation of the performance analysis: for maize.

Fig. 11 a, b, c, d, e: shows the graphical representation of the performance analysis: for maize

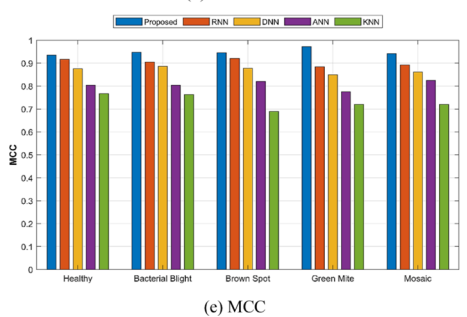
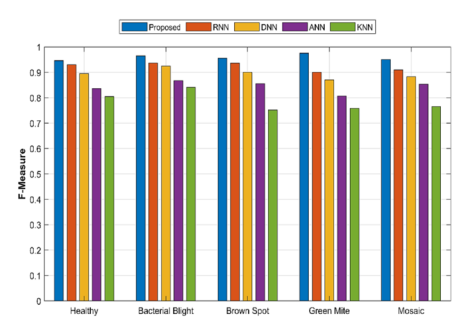
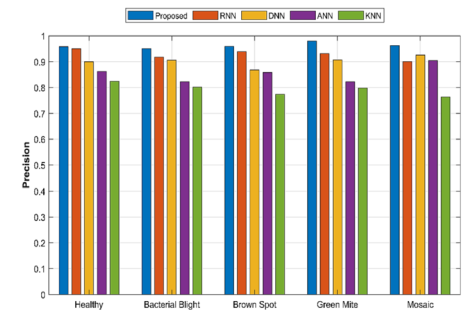
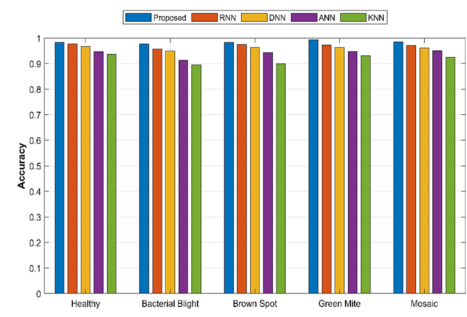
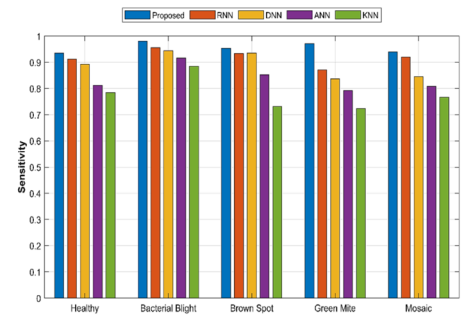


Table 5 Comparative analysis of performance metrics with existing models for tomato

Tomato					
Classification models	Sensitivity	Accuracy	Precision	F-measure	MCC
Proposed	0.9706	0.9883	0.9706	0.9706	0.9633
RNN	0.9283	0.9713	0.9283	0.9283	0.9104
DNN	0.8964	0.9585	0.8964	0.8964	0.8705
ANN	0.8333	0.9333	0.8333	0.8333	0.7917
KNN	0.7651	0.906	0.7651	0.7651	0.7064

4.6 Overall performance analysis: for tomato

Within this section, we conduct a thorough performance analysis for cashews, juxtaposing the proposed model with existing counterparts. Table 5 outlines a comparative evaluation utilizing various performance metrics for tomatoes.

The proposed Optimized Recurrent Neural Network (RNN)-based model for early diagnosis of crop pests and diseases in tomato plants outperforms existing models across various metrics. With a sensitivity of 97.06%, accuracy of 98.83%, precision of 97.06%, F-Measure of 97.06%, and MCC of 96.33%, the proposed model demonstrates superior performance compared to RNN, DNN, ANN, and KNN. While RNN is a close competitor, the proposed model excels in sensitivity and overall accuracy. The consistent outperformance across metrics suggests the effectiveness and potential significance of the proposed model in enhancing agricultural diagnostics and addressing challenges related to tomato crop health. The graphical depiction of the performance analysis for Tomato is displayed in Fig. 12(a), (b), (c), (d), (e).

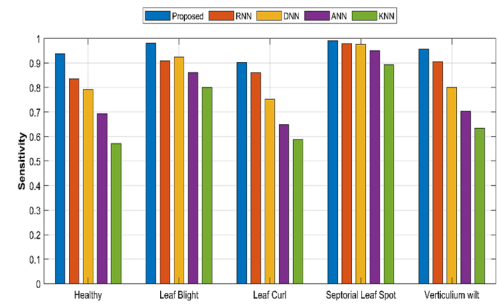
4.6.1 Confusion matrix

The confusion matrix, sometimes referred to as an error matrix, is a particular table arrangement used in the field of deep learning, most especially in the statistical classification problem, which enables the visualization of algorithm performance. It takes into account many measures, such as false positives (FP), false negatives (FN), true positives (TP), and true negatives (TN). The confusion matrix for cashew, cassava, maize, and tomato are illustrated below in Fig. (13).

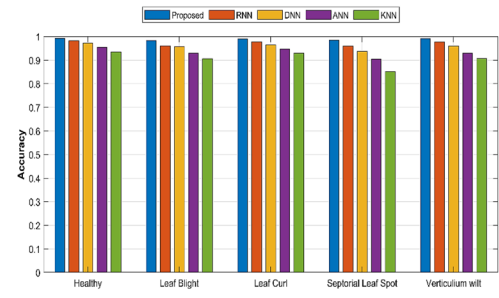
5 Conclusion

This paper proposed a ML-based Early Diagnosis of crop disease and pests. The initial phase in the training and testing procedure was to create a database with images of every component of the affected crop. The image dataset was augmented using rotation, flipping, and zooming during the image augmentation stage. The input image dataset was then sent to the preprocessing unit. During the preprocessing phase, the Adaptive Bilateral Filter was employed to lower noise and improve the image quality. The pre-processed pictures were resized using Lanczos interpolation and normalized to scale pixel values between 0 and 1. KEWO was proposed to obtain efficient image segmentation which is the combination of the Kapur's Entropy and Whale Optimization Algorithm. In this step, the sick portion was grouped into segments, and the characteristics were then retrieved. To extract features and assess the presence of different grey-level combinations in a picture, an efficient GLCM was utilized. It assessed the spatial separation between the pixels and created a feature-based gray level matrix for the color image. Finally, the MFRNN framework was proposed to detect crop diseases and pests. The proposed model is the combination of the MFO and RNN. The implementation was performed using the MATLAB software. The proposed model was evaluated in terms of accuracy, precision, F-Measure, MCC, and confusion matrix. The proposed model demonstrated superior performance across four types of crops, namely cashew, cassava, maize, and tomato, achieving remarkable accuracy rates of 98.4%, 98.3%, 98.5%, and 96.8%, respectively. It also exhibited impressive precision rates of 96.1%, 95.5%, 94.7%, and 97%, outperforming previous models in this field. Additionally, the model showcased strong F1 measures of 96.1%, 95.9%, 94.7%, and 97%, and achieved high sensitivities of 96.1%, 95.9%, 94.7%, and 97%. Notably, it attained MCC values of 95.1%, 94.9%, 93.8%, and 96.3% for cashew, cassava, maize, and tomato, respectively. Furthermore, the confusion matrix for each different crop was also presented.

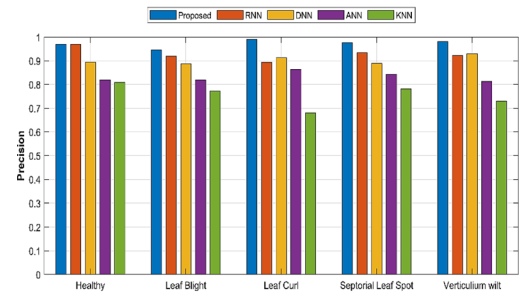
Fig. 12 a, b, c, d, e: shows the graphical representation of the performance analysis: for Tomato



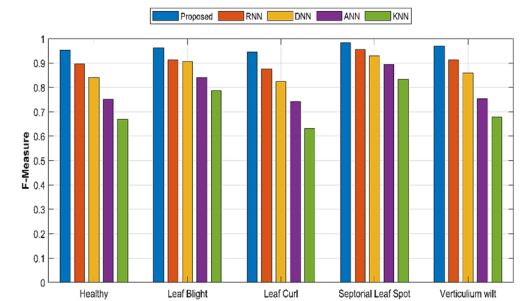
(a) Sensitivity



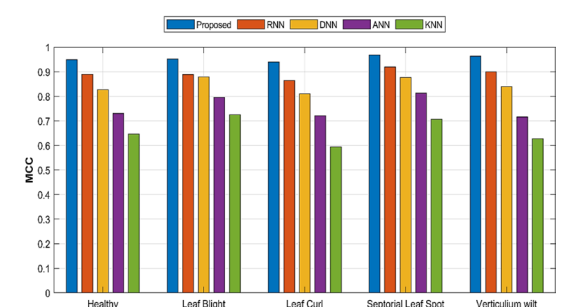
(b) Accuracy



(c) Precision



(d) F-Measure



(e) MCC

Fig. 13 Confusion matrix for cashew, cassava, maize, and tomato

Confusion Matrix

	Healthy	Anthraxnose	Gumosis	Leaf Miner	Red Rust	
Healthy	266 20.3%	5 0.4%	1 0.1%	2 0.2%	2 0.2%	96.4% 3.6%
Anthraxnose	2 0.2%	328 25.0%	2 0.2%	1 0.1%	4 0.3%	97.3% 2.7%
Gumosis	4 0.3%	5 0.4%	72 5.5%	1 0.1%	3 0.2%	84.7% 15.3%
Leaf Miner	0 0.0%	4 0.3%	3 0.2%	270 20.6%	4 0.3%	96.1% 3.9%
Red Rust	2 0.2%	4 0.3%	0 0.0%	2 0.2%	323 24.7%	97.6% 2.4%
	97.1% 2.9%	94.8% 5.2%	92.3% 7.7%	97.8% 2.2%	96.1% 3.9%	96.1% 3.9%
	Healthy	Anthraxnose	Gumosis	Leaf Miner	Red Rust	

i. Cashew

Confusion Matrix

	Healthy	Bacterial Blight	Brown Spot	Green Mite	Mosaic	
Healthy	226 15.0%	6 0.4%	2 0.1%	1 0.1%	3 0.2%	95.0% 5.0%
Bacterial Blight	2 0.1%	504 33.6%	1 0.1%	2 0.1%	3 0.2%	98.4% 1.6%
Brown Spot	3 0.2%	8 0.5%	290 19.3%	4 0.3%	5 0.3%	93.5% 6.5%
Green Mite	5 0.3%	4 0.3%	1 0.1%	195 13.0%	4 0.3%	93.3% 6.7%
Mosaic	3 0.2%	1 0.1%	2 0.1%	1 0.1%	226 15.0%	97.0% 3.0%
	94.6% 5.4%	96.4% 3.6%	98.0% 2.0%	95.1% 4.9%	93.8% 6.2%	96.9% 3.1%
	Healthy	Bacterial Blight	Brown Spot	Green Mite	Mosaic	

ii. Cassava

Confusion Matrix

	Healthy	Fall Armyworm	Grasshoper	Leaf Beetle	Leaf Blight	Leaf Spot	Streak Virus	
Healthy	39 3.7%	0 0.0%	1 0.1%	4 0.4%	3 0.3%	1 0.1%	2 0.2%	78.0% 22.0%
Fall Armyworm	0 0.0%	56 5.3%	0 0.0%	0 0.0%	2 0.2%	1 0.1%	3 0.3%	90.3% 9.7%
Grasshoper	1 0.1%	0 0.0%	132 12.4%	4 0.4%	2 0.2%	0 0.0%	4 0.4%	92.3% 7.7%
Leaf Beetle	0 0.0%	0 0.0%	0 0.0%	176 16.5%	2 0.2%	2 0.2%	3 0.3%	96.2% 3.8%
Leaf Blight	1 0.1%	0 0.0%	0 0.0%	1 0.1%	189 17.7%	3 0.3%	1 0.1%	96.9% 3.1%
Leaf Spot	0 0.0%	0 0.0%	0 0.0%	2 0.2%	1 0.1%	240 22.5%	4 0.4%	97.2% 2.8%
Streak Virus	0 0.0%	1 0.1%	2 0.2%	1 0.1%	1 0.1%	3 0.3%	177 16.0%	95.7% 4.3%
	95.1% 4.9%	98.2% 1.8%	97.8% 2.2%	93.6% 6.4%	94.5% 5.5%	96.0% 4.0%	91.2% 8.8%	94.7% 5.3%
	Healthy	Fall Armyworm	Grasshoper	Leaf Beetle	Leaf Blight	Leaf Spot	Streak Virus	

iii. Maize

Confusion Matrix

	Healthy	Leaf Blight	Leaf Curl	Septorial Leaf Spot	Verticillium wilt	
Healthy	91 7.9%	2 0.2%	1 0.1%	4 0.3%	1 0.1%	91.9% 8.1%
Leaf Blight	1 0.1%	253 21.8%	1 0.1%	5 0.4%	1 0.1%	96.9% 3.1%
Leaf Curl	2 0.2%	1 0.1%	101 8.7%	5 0.4%	2 0.2%	91.0% 9.0%
Septorial Leaf Spot	0 0.0%	0 0.0%	0 0.0%	530 45.8%	2 0.2%	99.6% 0.4%
Verticillium wilt	0 0.0%	3 0.3%	0 0.0%	3 0.3%	149 12.9%	96.1% 3.9%
	96.8% 3.2%	97.7% 2.3%	98.1% 1.9%	96.9% 3.1%	96.1% 3.9%	97.1% 2.9%
	Healthy	Leaf Blight	Leaf Curl	Septorial Leaf Spot	Verticillium wilt	

iv. Tomato

6 Future work

By focusing on expanding the dataset to encompass diverse environmental conditions and crop varieties, enhancing the model's robustness. Additionally, exploring real-time implementation in the field using edge computing or on-device processing for quicker diagnosis is crucial. Integration of sensor data, such as weather and soil information, could further improve the model's predictive capabilities. Lastly, collaborative efforts to develop a user-friendly interface for farmers to easily interpret and act upon the model's findings would contribute to practical and widespread adoption in agriculture.

Author contributions Mr. Vijesh Kumar Patel devised the overall concept and structure of the research. Directed the composition of the text, including the introductory, methodological, and concluding parts. Kumar Abhishek contributed to the development, testing, supervision and verification of the final model Shitharth Selvarajan oversaw and gave advice throughout the whole research. Oversaw the submission process and acted as the corresponding author, managing all correspondence with the journal.

Data availability The datasets used and/or analysed during the current study available from the corresponding author on reasonable request.

Declarations

Competing interests The authors declare no competing interests.

Open Access This article is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License, which permits any non-commercial use, sharing, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if you modified the licensed material. You do not have permission under this licence to share adapted material derived from this article or parts of it. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by-nc-nd/4.0/>.

References

1. Bhattacharya S, Mukherjee A, Phadikar S. A deep learning approach for the classification of rice leaf diseases. *Intell Enabled Res DoSIER*. 2020;2019:61–9.
2. Das A, Mallick C, Dutta S. Deep learning-based automated feature engineering for rice leaf disease prediction *Computational Intelligence in Pattern Recognition proceedings: of CIPR 2020*. Singapore: Springer; 2020.
3. Brinthakumari S, Sivaraja PM. MCNN: an approach for plant disease detection using modified convolutional neural network intelligent systems and human machine collaboration: select of ICISHMC 2022. Singapore: Springer Nature Singapore; 2023.
4. Jiang H, Li X, Safara F. IoT-based agriculture: deep learning in detecting apple fruit diseases. *Microprocess Microsyst*. 2021. <https://doi.org/10.1016/j.micpro.2021.104321>.
5. Anami BS, Malvade NN, Palaiah S. Deep learning approach for recognition and classification of yield affecting paddy crop stresses using field images. *Artif intell Agric*. 2020;4:12–20.
6. Prasath B, Akila M. IoT-based pest detection and classification using deep features with enhanced deep learning strategies. *Eng Appl Artif Intell*. 2023;121: 105985.
7. Rani APAS, Singh NS. Protecting the environment from pollution through early detection of infections on crops using the deep belief network in paddy. *Total Environ Res Themes*. 2022;3: 100020.
8. Sujaritha M, Kavitha M, Roobini S. Pest detection using improvised YOLO architecture. In *computer vision and machine intelligence paradigms for SDGs: select proceedings of ICRTAC-CVMIP*. Singapore: Springer Nature; 2023.
9. Kathole AB, Katti J, Lonare S, Dharmale G. Identify and classify pests in the agricultural sector using metaheuristics deep learning approach. *Franklin Open*. 2023;3:100024.
10. Ahad MT, Li Y, Song B, Bhuiyan T. Comparison of CNN-based deep learning architectures for rice diseases classification. *Artif Intell Agric*. 2023;9:22–35.
11. Wani JA, Sharma S, Muzamil M, Ahmed S, Sharma S, Singh S. Machine learning and deep learning based computational techniques in automatic agricultural diseases detection: methodologies, applications, and challenges. *Archiv Comput Methods Eng*. 2022;29(1):641–77.
12. Chen CJ, Huang YY, Li YS, Chang CY, Huang YM. An AIoT based smart agricultural system for pests' detection. *IEEE Access*. 2020;8:180750–61.
13. Abd Algani YM, Caro OJM, Bravo LMR, Kaur C, Al Ansari MS, Bala BK. Leaf disease identification and classification using optimized deep learning. *Measure Sens*. 2023;25:100643.
14. Kaur P, Harnal S, Gautam V, Singh MP, Singh SP. A novel transfer deep learning method for detection and classification of plant leaf disease. *J Ambient Intell Humaniz Comput*. 2023;14(9):12407–24.
15. Nayagam MG, Vijayalakshmi B, Somasundaram K, Mukunthan MA, Yogaraja CA, Partheeban P. Control of pests and diseases in plants using IOT Technology. *Measurement Sensors*. 2023;26: 100713.

16. Wang Y, Wang H, Peng Z. Rice diseases detection and classification using attention based neural network and bayesian optimization. *Expert Syst Appl.* 2021;178: 114770.
17. Rahman CR, Arko PS, Ali ME, Khan MAI, Apon SH, Nowrin F, Wasif A. Identification and recognition of rice diseases and pests using convolutional neural networks. *Biosys Eng.* 2020;194:112–20.
18. Ahmed I, Yadav PK. A systematic analysis of machine learning and deep learning-based approaches for identifying and diagnosing plant diseases. *Sustain Operat Comput.* 2023;4:96–104.
19. Islam MM, Talukder MA, Sarker MRA, Uddin MA, Akhter A, Sharmin S, Debnath SK. A deep learning model for cotton disease prediction using fine-tuning with smart web application in agriculture. *Intell Syst Appl.* 2023;20:200278.
20. Jesie RS, Premi MG, Jarin T. Comparative analysis of paddy leaf diseases sensing with a hybrid convolutional neural network model. *Measurement Sens.* 2024;31:100966.
21. Sourav MSU, Wang H. Intelligent identification of jute pests based on transfer learning and deep convolutional neural networks. *Neural Process Lett.* 2023;55(3):2193–210.
22. Islam MM, Adil MAA, Talukder MA, Ahamed MKU, Uddin MA, Hasan MK, Debnath SK. DeepCrop: deep learning-based crop disease prediction with web application. *J Agric Food Res.* 2023;14:100764.
23. Liu Z, Bashir RN, Iqbal S, Shahid MMA, Tausif M, Umer Q. Internet of Things (IoT) and machine learning model of plant disease prediction—blister blight for tea plant. *IEEE Access.* 2022;10:44934–44.
24. Mensah PK, Akoto-Adjepong V, Adu K, Ayidzoe MA, Bediako EA, Nyarko-Boateng O, Boateng S, Donkor EF, Bawah FU, Awarayi NS, Nimbe P. CCMT: dataset for crop pest and disease detection. *Data Brief.* 2023. <https://doi.org/10.1016/j.dib.2023.109306>.
25. Shorten C, Khoshgoftaar TM. A survey on image data augmentation for deep learning. *Journal of big data.* 2019;6(1):1–48.
26. Parsania P, Virparia PV. A review: Image interpolation techniques for image scaling. *Int J Innov Res Computer Commun Eng.* 2014;2(12):7409–14.
27. Jayakumar, D., Elakkiya, A., Rajmohan, R., & Ramkumar, M. O. (2020, July). Automatic prediction and classification of diseases in melons using stacked RNN based deep learning model. In 2020 international conference on system, computation, automation and networking (ICSCAN) (pp. 1–5). IEEE.
28. Albanese A, Nardello M, Brunelli D. Automated pest detection with DNN on the edge for precision agriculture. *IEEE J Select Topics Circuits Syst.* 2021;11(3):458–67.
29. Waleed, M., Abdullah, A. S., & Ahmed, S. R. (2020, September). Classification of Vegetative pests for cucumber plants using artificial neural networks. In 2020 3rd International Conference on Engineering Technology and its Applications (IICETA) (pp. 47–51). IEEE.
30. Resti Y, Irsan C, Putri MT, Yani I, Ansyori A, Suprihatin B. Identification of corn plant diseases and pests based on digital images using multinomial naïve bayes and k-nearest neighbor. *Sci Technol Indonesia.* 2022;7(1):29–35.
31. Haryanto T, Pratama A, Suhartanto H, Murni A, Kusmardi K, Pidanič J. Multipatch-GLCM for texture feature extraction on classification of the colon histopathology images using deep neural network with GPU acceleration. *J Computer Sci.* 2020;16(3):210.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.