



Res4net-CBAM: a deep cnn with convolution block attention module for tea leaf disease diagnosis

Parag Bhuyan¹ · Pranav Kumar Singh¹ · Sujit Kumar Das²

Received: 5 May 2023 / Revised: 17 July 2023 / Accepted: 3 October 2023 /
Published online: 3 November 2023

© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2023

Abstract

Early detection of tea leaf diseases is crucial for maintaining crop yield and agricultural production. However, manual inspection is a time-consuming and error-prone process, emphasizing the need for automated procedures. Deep learning methods have shown great potential in diagnosing plant leaf diseases. Convolutional Neural Networks (CNNs) outperform traditional deep learning models. However, the performance of these approaches is limited due to computational complexity, feature quality issues, and increasing feature dimensionality. In this study, we propose Res4net-CBAM, a deep convolutional neural network (CNN) specifically designed for tea leaf disease diagnosis, aiming to reduce the model's complexity and improve disease identification accuracy. The Res4net-CBAM model utilizes a residual block-based Res4net architecture with a network interactive convolutional block attention module (CBAM) to accurately extract complex features associated with different diseases. We conducted extensive experiments to compare the performance of our model with standard CNN models such as AlexNet, VGG16, ResNet50, DenseNet121, and InceptionV3, based on metrics such as accuracy, precision, recall, and F1-score. Our results demonstrate that the Res4net-CBAM model outperforms all other models, achieving an average recognition accuracy of 98.27% on self-acquired tea leaf disease data samples. Specifically, the Res4net-CBAM model achieved an average sensitivity of 98.39%, specificity of 98.26%, precision of 98.35%, and F1-score of 98.37%, while utilizing the Adagrad optimizer with a learning rate of 0.001. Moreover, our model surpasses some recent and existing works in this field, highlighting its effectiveness in diagnosing tea leaf diseases.

Keywords Tea leaf disease · Deep learning · Smart agriculture · Attention module

✉ Pranav Kumar Singh
sngpranav@gmail.com

Parag Bhuyan
paragbhuyan12@gmail.com

Sujit Kumar Das
dassujit88@gmail.com

¹ Central Institute of Technology, Kokrajhar, BTR, Assam 783370, India

² Siksha 'O' Anusandhan University, Bhubaneswar, Odisha, India

1 Introduction

Tea is one of the most popular non-alcoholic beverages in the world. It is also an essential part of India's identity and economy. India is the second-largest tea-producing country globally, accounting for 28% of the world's tea production [1]. Darjeeling Tea, a unique blend of tea from India, is famous for its distinctive flavor and aroma [2]. Around 80% of the tea produced in India is consumed domestically, making India one of the top tea-consuming nations in the world [3]. In the calendar year 2020, India produced 1257.52 million kg of tea, and in the fiscal year 2020-21, it produced 1283 million kg. In April 2022, the production was 91.77 million kg, and in May 2022, it increased to 127.11 million kg. As of January-September 2022, India's tea production reached 984.67 million kg [4].

The primary concern in tea production is the prevention and detection of tea leaf diseases. It is crucial to have an understanding of the frequency of tea leaf diseases and to be able to make accurate disease predictions, as they can significantly impact overall tea production. If disease outbreaks are not effectively managed, they can have a major detrimental effect on tea production. Tea production can face a wide range of obstacles, particularly pest and disease attacks, which can sometimes lead to mediocre yields. Tea leaf infections, caused by fungi, viruses, and bacteria, occur during the harvest period. Dealing with these infections in a large-scale manner poses significant challenges for farmers.

Despite living in an era of technological advancements, many farmers still rely on traditional methods of visually inspecting plants to identify diseases. However, this manual observation and inspection approach brings forth several issues. The manual approach to disease detection means that farmers rely mainly on books or their own experience to recognize diseases, which has limitations such as high laboratory testing costs, time-consuming, and difficult to perform in the field. The manual technique only allows farmers to identify specific plant infections and diseases on a small scale. Moreover, it does not guarantee early detection, and newly emerging infections may not be visible through visual inspection alone. Consequently, tea leaf diseases may go undetected, and inappropriate precautionary measures may be taken by farmers, resulting in a loss of crop yield.

There is a need for technological interventions in the form of smart agriculture technology to effectively prevent and control tea leaf diseases, ensuring the sustainability of tea production and providing a stable income for farmers. The utilization of an automated system that leverages image classification techniques can play a crucial role in this endeavor.

1.1 Problem definition

Although conventional image processing techniques are useful for detecting crop diseases, they have some drawbacks, such as scalability, efficiency, performance, and precision. Deep learning has emerged as a solution to these limitations of conventional image processing, especially in image classification. Deep learning models have demonstrated remarkable success and have gradually become the predominant approach in various domains [5]. In the medical sector, image processing techniques are commonly employed for disease detection, diagnosis, and monitoring of medical care [6, 7]. Furthermore, deep learning models have been utilized for automatic character recognition systems [8]. With the powerful ability of deep learning architectures to extract crucial features from images, researchers have increasingly focused on employing deep learning, computer vision, and image processing techniques for plant disease detection [9, 10]. Recent approaches have made disease and pest detection more robust, enabling early prevention and treatment [11]

1.2 Motivation

Despite previous efforts in the field, there is still significant room for improvement in automated tea leaf disease detection systems. The use of an Attention module in the primary deep learning model can reduce the aforementioned drawbacks of the existing approaches. By combining the Attention module with deep learning models, these inspiring image classification methods can provide encouraging results in agricultural disease detection [12]. Attention modules have recently been integrated into deep learning models in various fields [13]. Attention modules, such as CBAM, discard irrelevant information and retain important features only. This allows the model to assign adjusted and optimized weights to each feature map channel. As a result, the number of trainable parameters in the model is reduced, making it lightweight. Therefore, we are motivated to develop a new model that incorporates the CBAM with Res4net, which has the potential to surpass existing techniques and pave the way for more effective teal leaf disease detection. The motivation of this research is to investigate tea leaf diseases in their natural environment and provide a highly accurate solution for automated disease detection using deep learning. This study proposed a lightweight model for real-time tea leaf disease classification. The success of this study could empower farmers to make informed decisions about using pesticides, resulting in more efficient tea leaf disease prevention and control.

1.3 Contribution

In this paper, we strive to present a better solution to above-mentioned problems and provide the following contributions towards it:

- We propose a tea leaf disease detection model called Res4net-CBAM for the automatic detection and identification of tea leaf diseases in natural environments. Res4net integrated with CBAM helps to add adjusted optimized weights to each feature map channel in order to ensure that the network pays attention to relevant feature information and minimizes the interference of redundant information.
- We fine-tune the proposed architecture using multiple optimizers, including Adagrad, Adam, and Stochastic Gradient Descent with momentum, and using learning rates of 0.01 and 0.001 for the identification and detection of diseased tea leaf images.
- We use data augmentation techniques to increase the number of training images, addressing the issue of insufficient samples and improving the network's detection and identification capabilities.
- We conducted extensive experiments and compared the proposed model with five different standard CNN techniques: AlexNet, VGG16, ResNet50, DenseNet121, and InceptionV3. Additionally, we compared our work with recent and relevant research works in the field.

1.4 Organization of the Work

The rest of the paper is structured as follows:

- Section 2 comprehensively discusses the related works in this field and summarizes them in a tabular format.

- Section 3 provides details about the materials used, including dataset acquisition and preprocessing. It also presents the details of the proposed method to address the given problem.
- Section 4 of our paper is dedicated to the experimental evaluation of our proposed method. It discusses the experimental setup details and presents the results obtained through these experiments.
- Section 5 provides a detailed discussion of the results obtained from the experiments.
- Section 6 presents the conclusion and overall summary of our work.

2 Related work

Computer vision and image processing techniques have increasingly been employed in the detection of crop diseases [14, 15]. Deep learning, a cutting-edge tool for image processing and data analysis [16], has demonstrated remarkable success in various agricultural applications. A survey on the use of deep learning in agricultural data highlights its high accuracy and superior performance compared to traditional image processing techniques [17]. CNNs are particularly intriguing within the realm of deep learning. They eliminate the need for complex image processing techniques and manual feature extraction, thereby facilitating disease detection in a more efficient and straightforward manner [18–20]. These CNNs have proven to be effective tools for crop disease identification and classification.

Currently, the most widely used CNN models for image processing include VGG, InceptionV4 [21], GoogLeNet [22], ResNet [23], and DenseNet [24]. The network architecture plays a crucial role in capturing and extracting relevant image information. It has been observed that wider and deeper networks have the capacity to capture a larger amount of image information and extract essential features. In 2018, Hu et al. [25] introduced Squeeze-and-Excitation networks (SE networks), which employ an adaptive recalibration of channel-wise feature responses by explicitly modeling interdependencies between channels. This recalibration enhances the network's ability to focus on important features.

The attention model CBAM was proposed by Woo et al. in 2018 [26]. CBAM is a compact and effective attention mechanism designed for feedforward CNNs. It sequentially calculates channel and spatial attention maps based on intermediate feature maps. These attention maps are then multiplied by the input features to refine and adapt the features. Due to its lightweight design, CBAM can be easily integrated into any CNN without causing computational overhead.

By incorporating attention mechanisms like CBAM or SE networks into deep neural networks, the models gain the ability to selectively focus on informative features, leading to improved performance in various image analysis tasks. These attention modules have demonstrated their effectiveness and versatility in enhancing the capabilities of CNNs.

In recent years, extensive research has been conducted to identify and diagnose plant leaf diseases. The integration of machine learning, computer vision, and high-performance computing solutions has demonstrated its power to solving a variety of agricultural problems. In this section, we examine various contributions to plant leaf disease detection research. Table 1 provides an overview of relevant studies, including their objectives, methodologies, accuracy rates, and other findings, which are discussed below.

Sun et al. [27] proposed a new approach that combines Simple Linear Iterative Clustering (SLIC) with Support Vector Machine (SVM) to detect diseases in tea leaves. They used 261 images of 5 types of common diseases in the complex background of tea leaf disease as their

Table 1 Summary of the Related Work

Year, Reference	Objective	Proposed Methodology	Result	Remarks
2019 [27]	Improving the extraction of tea plant leaf disease saliency map under complex backgrounds	SLIC (Simple Linear Iterative Clustering) + SVM (Support Vector Machine)	-Accuracy: 98.5%	-Reliable and effective method for extracting image saliency maps from complex background - Low shot learning methods - Low Accuracy
2019 [28]	Low shot learning method for identification of tea leaf's diseases	Support vector machine + Improved C-DCGAN + VGG16	-Accuracy: 90%	- Few parameters and fast identification speed - Low identification accuracy
2019 [29]	A model with low cost and high identification accuracy	Improved Cifar10-fast model	-Accuracy : 92.5%	- Low Accuracy
2021 [30]	To improve the performance and analysis of Tea Leaf Blight (TLB) disease detection	Fast RCNN + VGG16	-Accuracy: 84.5%	- Low Accuracy
2021 [31]	A new method for detection of five different tea leaf diseases automatically	Non-dominated sorting genetic algorithm (NSGA-II) + Principal Component Analysis (PCA) + Support Vector Machine (SVM)	-Accuracy: 83%	- Low Accuracy Generalization of the model is poor
2022 [32]	To automatically detect and diagnose tea leaf diseases in real field images	AX-RetinalNet	-F1-score: 0.954 -mAP value: 93.83%	- Lower predictive accuracy
Our Proposed Model	A lightweight model for Automatic detection and identification of tea leaf diseases with high accuracy and lower training time and low cost	Embedding CBAM module in a deep residual CNN Architecture Res4net	-Accuracy: 98.27%	-Reduced the model parameters as well as training convergence time -High accuracy - Fully Automatic and reliable -Low cost

research object and obtained an accuracy rate of 98.5%, a precision rate of 96.8%, a recall value of 98.6%, and an F1 score of 97.7%.

In recent years, a number of researchers have utilized few-shot learning (FSL) techniques for detecting and classifying plant leaf diseases. Hu et al. [28] presented a low-shot learning method for identifying diseases in tea leaves, which employed support vector machines and deep learning networks. Specifically, support vector machines were used for low-shot segmentation of disease spots in tea leaf images. The researchers then used enhanced conditional deep convolutional generative adversarial networks (C-DCGAN) for data augmentation. This involved using the segmented illness spot images as inputs to generate fresh training samples, which were then fed into the VGG16 deep-learning model for disease detection. The experimental results showed an average identification accuracy of 90%.

Hu et al. [29] presented an approach for detecting tea leaf diseases by utilizing a sophisticated deep CNN. They developed the CIFAR10-quick model, which utilizes two parallel convolution paths for multi-scale feature extraction instead of a series of connected convolution layers. To improve the model's computation and minimize the number of learnable parameters, they implemented depth separable convolution in the multi-scale feature extraction module rather than standard convolution. The enhanced Cifar10-fast model has the advantages of having fewer parameters, high detection accuracy, and fast detection speed. The proposed model achieves an average detection accuracy of 92.5% for healthy tea, tea bud blight, tea leaf blight, and tea red scab, with a loss of 0.002, surpassing the performance of the baseline Cifar10-quick model.

Hu et al. [30] introduced a deep learning-based approach for Tea Leaf Blight (TLB) disease detection, aimed at improving performance and analysis. The authors utilized the Retinex algorithm to improve image quality and reduce the impact of lighting variations and shadows on recognition performance. They employed an advanced Fast R-CNN model for TLB detection and a VGG16-based disease severity grading framework to assess overall disease severity and evaluate diseased leaves. The proposed method was designed to work with TLB images acquired in natural settings environments, and it addresses the effects of complex noise environments, adjusting illumination, leaf occlusion, and conflicting scale. The proposed method achieved an accuracy of 84.5%, which is 9% higher than the accuracy of the conventional machine learning method.

Mukhopadhyay et al. [31] presented a new method for the automatic detection of five different tea leaf diseases using image processing techniques. They utilized a non-dominated sorting genetic algorithm (NSGA-II) to detect disease patches in tea leaves and Principal Component Analysis (PCA) to determine the optimal set of features. For identifying disease from diseased leaves, a multiple-class SVM was employed, achieving an average accuracy of 83%.

Bao et al. [32] proposed AX-RetinaNet, an advanced RetinaNet for the automatic detection and diagnosis of tea leaf diseases in real field images. The proposed model achieves this by fusing multiscale features and creating feature maps with rich semantic information. To accomplish this, an enhanced multiscale feature fusion module of the X-module is utilized. Additionally, the channel attention module in AX-RetinaNet adds adaptively tuned weights to each feature map channel, enabling the network to select more useful features while minimizing interference from unused features. An image augmentation technique is also applied to increase the number of training photos and prevent overfitting issues. The F1-score value of AX-RetinaNet reached 0.954, and the mean average precision (mAP) value for recognition and detection reached 93.83%.

3 Materials and method

This section outlines the materials used in the research, including the dataset acquisition and necessary preprocessing techniques used, and also provides a comprehensive overview of the methodology employed in this work.

3.1 Materials

In this section, we describe our process of collecting tea plant leaf images, including both healthy and diseased leaves, and detail the preprocessing techniques applied in Sections 3.1.1 and 3.1.2, respectively.

3.1.1 Dataset acquisition

The dataset used in this study consisted of tea plant leaf disease images collected from various angles within a Tea garden environment. The images were captured using a hand-held digital camera (Canon EOS 3000D) and a mobile phone (iPhone 12) in a local tea garden located in Dibrugarh, Assam. The geographic coordinates of the tea garden are approximately 27.472108° North latitude, 94.963828° East longitude, with an altitude of 40 meters above sea level.

During the image acquisition process, the target samples included healthy tea leaves, as well as leaves exhibiting red leaf spots and brown blight. The camera settings employed auto exposure and autofocus features, ensuring optimal image capture. The resolution of the images was set to 4000 by 3000 pixels.

In total, the dataset comprised 2867 images, consisting of 867 brown blight images, 1000 red leaf spot images, and 1000 healthy images. These images form the foundation of the dataset utilized for training and evaluating the proposed model in this research.

3.1.2 Dataset preprocessing

The leaf images utilized in this study were carefully labeled by domain experts. The selection criteria for identifying infected tea leaves were based on a previously published detection scheme [33, 34]. In order to enhance the training process of our proposed model and standardize our data samples, we augmented our own dataset with images from a standard dataset [35]. Specifically, we combined 113 brown blight images, 143 red leaf spot images, and 74 healthy tea leaf images from the standard dataset with our own data samples.

The total dataset was divided into three subsets: 80% for training data, 20% for testing data, and a 20% validation set extracted from the training data to prevent overfitting. The experiments involved a total of 3197 tea leaf disease images, with 2045 images allocated for training, 512 for validation, and 640 for testing. To augment the training samples and increase the diversity of the dataset, we performed natural data augmentation techniques on the initial 2045 training samples, resulting in a total of 6135 training samples. In order to reduce computational complexity, the dimensions of the images were reduced to 128 x 128 pixels.

Sample images depicting healthy tea leaves, as well as two types of infected tea leaves, are shown in Fig. 1. These visual examples provide a visual representation of the tea leaf images used in the study, showcasing the differences between healthy leaves and those affected by brown blight and red leaf spots.



Fig. 1 Sample Images of Tea Leaves: Healthy and Infected

3.2 Methods

This section is organized into three subsections, each dedicated to a distinct aspect of the proposed model. In Section 3.2.1, we delve into the Res4Net network architecture, providing a comprehensive description of its layers, filters, and activation functions. Section 3.2.2 centers around the CBAM. Here, we elucidate the architecture and functionality of CBAM, emphasizing its advantageous role in improving the attention mechanism of the network. The discussion also encompasses the benefits of incorporating CBAM, such as its capability to enhance feature representation and focus on salient regions within the tea leaf images. Finally, Section 3.2.3 presents the proposed integrated Res4Net-CBAM network. This subsection highlights the fusion of the Res4Net architecture with the CBAM module, illustrating how their combination synergistically enhances the model's capabilities.

3.2.1 Res4net network

We developed a feature extraction module to enhance the performance of image recognition. The model architecture, as shown in Fig. 2, is described below layer by layer. The first layer of the model is a convolution layer with a kernel size of 7×7 and strides of 2×2 . A transition layer is created by adding Batch Normalization (BN) layers and LeakyReLU activation to regulate the input and expedite the training. Each convolutional block, also known as a residual block, consists of a sequence of 2D Convolution, Batch Normalization, LeakyReLU layers, and an identity or convolutional skip connection only after the transition layer. We developed the Res4net classification network, consisting of four optimized residual block modules. After the residual blocks, a Global Average Pooling operation generates a flattened and compacted representation of the data from the upper layers. The final prediction of classes is obtained using a fully connected layer with a dropout and an output layer (Softmax).

The relevant parameters of the Res4net network are shown in Table 2. Furthermore, we investigate the impact of increasing the number of residual blocks on the classification performance by including various numbers of residual blocks in the network model while keeping the same residual block architecture. Specifically, we examine five variations of the proposed architecture (with 3-7 residual blocks, keeping the underlying structure of the remaining blocks) empirically. However, we found that increasing the number of blocks beyond four did not improve the classification performance for this dataset. Therefore, we keep our model to four residual blocks.

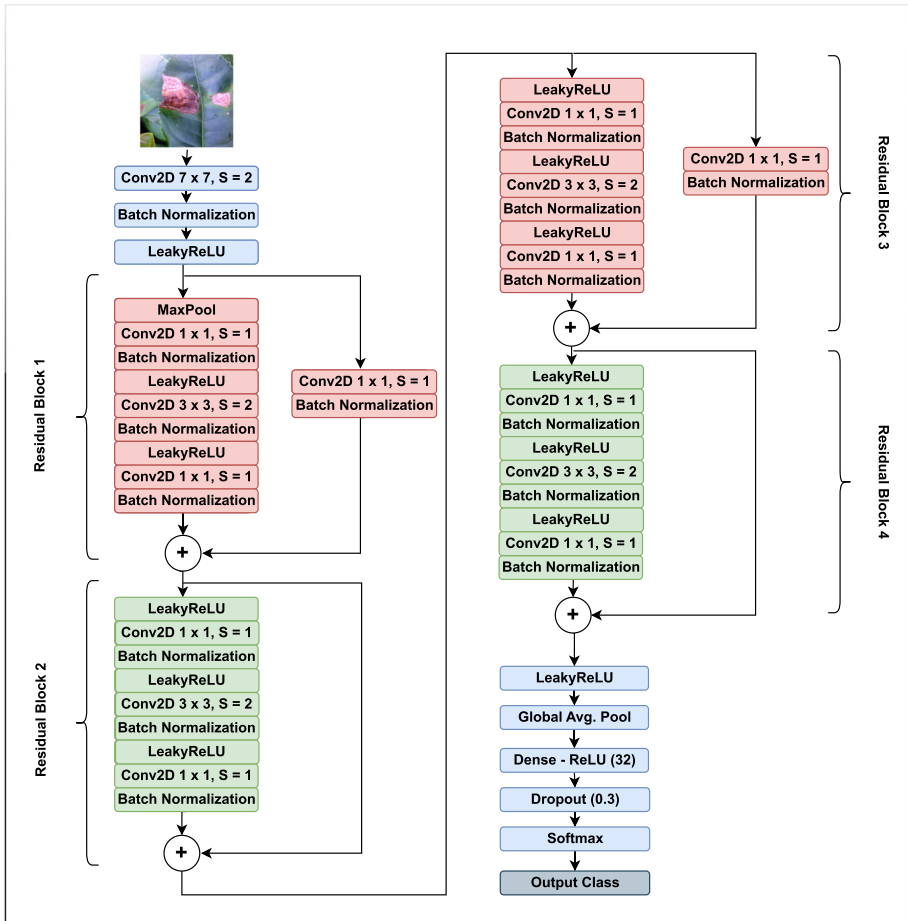


Fig. 2 Res4net Network Structure

3.2.2 Convolutional block attention module

Self-attention networks have gained popularity in recent years due to their ability to capture distinctive characteristics of regions of interest, and researchers from a wide range of fields have extensively explored them. For example, Zeng et al. [36] used self-attention in crop leaf disease recognition, Mi et al. [37] used it in wheat stripe rust grading, Xiao et al. [38] used it in brain disease detection, Liu et al. [39] used it in tablets defect detection, Chao et al. [40] used it in sign language recognition, and so on. The attention mechanism is a deep learning technique that automatically learns important features and focuses on them selectively. The attention model allocates resources to only important features [41]. Woo et al. [26] proposed CBAM to extract characteristics from both channel and spatial dimensions simultaneously.

CBAM has two sub-modules, namely the Spatial Attention Module (SAM) and the Channel Attention Module (CAM). The CBAM module reduces the computational power and

Table 2 Summary of the layer-wise parameters for the proposed Model

Name of the Layers	Filter size (FS) and Strides (S)	Activations	Number of Trained Parameters
Input Layer	-	128 x 128 x 3	0
Conv2D_0, BatchNorma_0, LeakyReLU_0	FS = 7 x 7, S = 2	64 x 64 x 64	9472 + 256 + 0
MaxPool2D_0	-	32 x 32 x 64	0
Conv2D_1, BatchNorm_1, LeakyReLU_1	FS = 1 x 1, S = 1	32 x 32 x 128	8320 + 512
Conv2D_2, BatchNorm_2, LeakyReLU_2	FS = 3 x 3, S = 2	32 x 32 x 128	147584 + 512 + 0
Conv2D_3	FS = 1 x 1, S = 1	32 x 32 x 256	16640
Conv2D_4	FS = 1 x 1, S = 1	32 x 32 x 256	33024
BatchNorm_3, BatchNorm_4	-	32 x 32 x 256	1024 + 1024
Add_0	-	32 x 32 x 256	0
GlobalAvgPool2D_0, GlobalMaxPool2D_1	-	1 x 1 x 256	0
Dense_0	-	1 x 1 x 32	8192
Dense_1	-	1 x 1 x 256	8192
Add_1	-	1 x 1 x 256	0
Activation_0	-	1 x 1 x 256	0
Multiply_0	-	32 x 32 x 256	0
GlobalAvgPool2D_2, GlobalMaxPool2D_3	-	32 x 32 x 1	0
Concatenate_0	-	32 x 32 x 2	0
Conv2D_5	FS = 7 x 7, S = 2	32 x 32 x 1	99
Activation_1	-	1 x 1 x 256	0
Multiply_1	-	32 x 32 x 256	0
LeakyReLU_3	-	32 x 32 x 256	0
Conv2D_6, BatchNorm_5, LeakyReLU_4	-	32 x 32 x 128	32896 + 512 + 0
Conv2D_7, BatchNorm_6, LeakyReLU_5	-	32 x 32 x 128	147584 + 512 + 0
Conv2D_8, BatchNorm_7	-	32 x 32 x 256	33024 + 1024
Add_2	-	32 x 32 x 256	0
GlobalAvgPool2D_4, GlobalMaxPool2D_5	-	1 x 1 x 256	0

Table 2 continued

Name of the Layers	Filter size (FS) and Strides (S)	Activations	Number of Trained Parameters
Dense_2	-	1 x 1 x 32	8192
Dense_3	-	1 x 1 x 256	8192
Add_3	-	1 x 1 x 256	0
Activation_2	-	1 x 1 x 256	0
Multiply_2	-	32 x 32 x 256	0
GlobalAvgPool2D_6, GlobalMaxPool2D_7	-	32 x 32 x 1	0
Concatenate_1	-	32 x 32 x 2	0
Conv2D_9	FS = 7 x 7, S = 2	32 x 32 x 1	99
Activation_3	-	1 x 1 x 256	0
Multiply_3	-	32 x 32 x 256	0
LeakyReLU_6	-	16 x 16 x 256	0
Conv2D_10, BatchNorm_8, LeakyReLU_7	FS = 1 x 1, S = 1	32 x 32 x 256	65792 + 1024 + 0
Conv2D_11, BatchNorm_9, LeakyReLU_8	FS = 3 x 3, S = 2	16 x 16 x 256	590080 + 1024 + 0
Conv2D_12	FS = 1 x 1, S = 1	16 x 16 x 512	131584
Conv2D_13	FS = 1 x 1, S = 1	16 x 16 x 512	131584
BatchNorm_10, BatchNorm_11	-	16 x 16 x 512	2048 + 2048
Add_4	-	16 x 16 x 512	0
GlobalAvgPool2D_8, GlobalMaxPool2D_9	-	1 x 1 x 512	0
Dense_4	-	1 x 1 x 64	32768
Dense_5	-	1 x 1 x 512	32768
Add_5	-	1 x 1 x 512	0
Activation_4	-	1 x 1 x 512	0
Multiply_4	-	16 x 16 x 512	0
GlobalAvgPool2D_10, GlobalMaxPool2D_11	-	16 x 16 x 1	0

Table 2 continued

Name of the Layers	Filter size (FS) and Strides (S)	Activations	Number of Trained Parameters
Concatenate_2	-	16 x 16 x 2	0
Conv2D_14	FS = 7 x 7, S = 2	16 x 16 x 1	99
Activation_5	-	1 x 1 x 512	0
Multiply_5	-	32 x 32 x 512	0
LeakyReLU_9	-	16 x 16 x 512	0
Conv2D_15, BatchNorm_13, LeakyReLU_10,	-	16 x 16 x 256	131328 + 1024 + 0
Conv2D_16, BatchNorm_14, LeakyReLU_11	-	16 x 16 x 256	590080 + 1024 + 0
Conv2D_17, BatchNorm_15	-	16 x 16 x 512	131584 + 2048
Add_6	-	16 x 16 x 512	0
GlobalAvgPool2D_12, GlobalMaxPool2D_13	-	1 x 1 x 512	0
Dense_6	-	1 x 1 x 64	32768
Dense_7	-	1 x 1 x 512	32768
Add_7	-	1 x 1 x 512	0
Activation_6	-	1 x 1 x 512	0
Multiply_6	-	16 x 16 x 512	0
GlobalAvgPool2D_14, GlobalMaxPool2D_15	-	16 x 16 x 1	0
Concatenate_3	-	16 x 16 x 2	0
Conv2D_18	FS = 7 x 7, S = 2	16 x 16 x 1	99
Activation_7	-	1 x 1 x 512	0
Multiply_7	-	32 x 32 x 512	0
LeakyReLU_12	-	16 x 16 x 512	0
GlobalAvgPool2D_16	-	1 x 1 x 512	0
Dense-ReLU(32)	-	32(FC)	16416
Dropout(0.3)	-	Dropout(30%)	0
Dense-Softmax	-	3	99

parameters and is compatible with all existing models. Figure 3(a) depicts the details of CBAM. The inputs and outputs of the CBAM structure are sequentially coupled to the channel attention module and the spatial attention module. To aggregate spatial data, CBAM uses both global maximum pooling and global average pooling. Combining both pools can eliminate redundant data and ensure extensive extraction of high-level features, allowing accurate learning of inter-dependencies between different channels. The channel attention module highlights the more important parts of the images while ignoring irrelevant information. First, average-pooling and max-pooling are used to process the input features in parallel. The multilayer perceptron (MLP) subsequently transfers these two types of information through a single hidden layer. Finally, the output features are aggregated using component-based aggregation.

CAM is expressed as follows:

$$M_c(F) = \sigma(MLP(AvgPool(F)) + MLP(MaxPool(F))) \tag{1}$$

$$= \sigma(W_1(W_0(F_{avg}^c)) + W_1(W_0(F_{max}^c)))$$

where M_c is the channel attention module, F represent input features, σ indicates sigmoid function, and W_1 and W_0 are weights.

SAM aims to identify the most important region of the input features, after they have been processed by the channel attention module. The input features are first processed in parallel by average-pooling and max-pooling, and the resulting features are then passed through a convolutional layer. In a nutshell, SAM is expressed as follows.

$$M_s(F) = \sigma(f[AvgPool(F); MaxPool(F)]) \tag{2}$$

where M_s is the SAM, f represent convolution operation.

3.2.3 Res4net-CBAM network

Plant leaf images often have complex background effects, and plant leaf disease spots exhibit small inter-class differences and large intra-class differences, which can make it difficult for

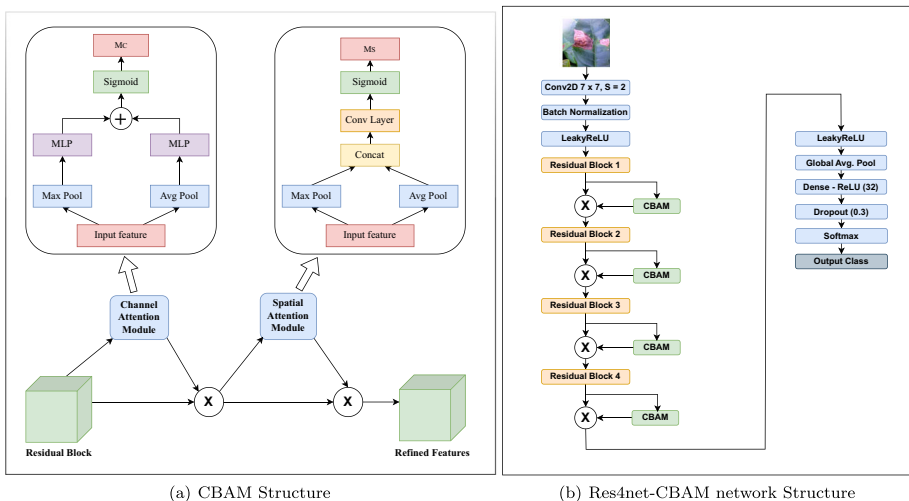


Fig. 3 Detailed Structure of CBAM and Res4net-CBAM

Res4net to detect disease spots due to redundant information. To address this issue, this study proposes integrating CBAM into the Res4net network, which can more accurately identify the location of disease spots in the image and increase the model's detection accuracy by reducing the influence of redundant information in the input image. The attention mechanism in CBAM magnifies the disease spots in the original image while suppressing non-disease features. The proposed Res4net-CBAM model consists of four residual blocks with a similar architecture to the Res4net network, and attention modules are embedded in each residual block.

Figure 3(b) depicts the detailed structure of the Res4net-CBAM network. The input layer takes images of size 128 x 128 with a depth of three color channels, R, G, and B. The first layer of Res4net-CBAM is a 7 x 7 convolution layer with 2 x 2 strides to process the input image. The following layers are Batch Normalization (BN) and LeakyReLU activation, which act as a transition layer to control the inputs and speed up training. The middle layer uses residual block modules, and the output of each module is passed through the CBAM module to extract image features more accurately. The output from the CBAM module is multiplied by the output of the original residual block, and this multiplied output is then added to the next residual block and so on. Finally, the network has one FC layer with 32 neurons, and a dropout layer of 30% is applied before entering the FC layer. The final layer is a softmax activation function for multi-class classification. The equation below defines the softmax function.

$$P(y = i|X; \theta) = \frac{\exp(\theta_i^T X)}{\sum_{k=1}^k \exp(\theta_k^T X)} \quad (3)$$

where x is the input vector to the model, θ^i is the dimension of feature vector x , y_i is the output values, The output probability of each class for each input x is given by $P(y = i|X; \theta)$.

4 Experimental evaluation

In this section, we evaluate the effectiveness of the proposed model using accuracy, F1-score, precision, sensitivity, and specificity and compare it with state-of-the-art models and several popular existing mechanisms. We provide experimental setup details and present obtained results in Sections 4.1 and 4.2, respectively.

4.1 Experimental setup

The experimental setup for predicting tea leaf disease involved implementing and analyzing a proposed model using Python 3.7.12, which was then compared to existing state-of-the-art networks. All experiments were conducted using the TensorFlow [42] Python package and the freely available Kaggle GPU environment. The dataset was initially split into an 80-20 random split, with 80% of the data used for training, 20% used for testing, and 20% of the training data used as a validation set to address overfitting. To increase the number of images and enhance model learning, the training samples were flipped horizontally and vertically, resulting in 6135 training images, 512 validation images, and 640 testing images. The proposed model was trained using different types of optimizers such as Adagrad, SGD (with momentum), and ADAM, with learning rate settings of 0.01 and 0.001, resulting in a total of six sets being tested. To evaluate each network's general behavior, the results were pooled for 10 different random splits in each model using an initial stopping condition. Each

instance of model training lasted up to 100 epochs, with an early termination threshold of 50 epochs. The training process was stopped if no improvement in validity loss was observed for 50 consecutive epochs, and the best set of network weights found so far was retrieved. Finally, the proposed model's classifier was tested and validated 60 times with six sets of experiments and 10 runs in each set. The proposed model has been evaluated using several important evaluation metrics, which are defined in Table 3.

4.2 Results

The main objective of this study is to investigate the classification performance of a deep learning network based on a residual structure for tea leaf diseases and compare it with AlexNet, ResNet50, VGG16, InceptionV3, and DenseNet121. All tests were performed on the same dataset. The proposed architecture, which includes a deep residual block composed of 2D convolution, batch normalization, and LeakyReLU, was found to be an innovative solution for tea leaf disease detection, showing promising results. We tested all networks from Res4Net to Res10Net and found that the performance of classifying tea leaf diseases was not significantly affected by the number of residual blocks. However, the performance of the network decreased when we reduced the number of residual blocks below four. Therefore, we conclude from the experimental results that Res4Net is one of the most effective in tea leaf disease detection.

All evaluation metrics taken into account show that the proposed model has good classification performance. The confusion matrix is used to test how well the proposed method responds. The confusion matrix used for testing the tea leaf test dataset is shown in Fig. 4. Table 3 shows the equation used to calculate the performance metric from the confusion matrix. The results of this study suggest that Res4Net-CBAM is an effective method for the detection and classification of tea leaf diseases.

We evaluated the effectiveness of our proposed model using several metrics including accuracy, F1-score, precision, sensitivity, and specificity, and utilized the cross-entropy loss function. For multi-classification problems, achieving accuracy is straightforward, and we only need to focus on correctly predicting the probability of each class. To train our model, we employed various optimizers such as ADAM, Stochastic Gradient Descent (SGD), and Adagrad, setting the learning rates to 0.01 and 0.001 to enhance network convergence. Among these, we found that Adagrad with a learning rate of 0.001 produced relatively good results for each evaluation metric. The obtained results are presented in Table 4. Experimental results showed that the highest accuracy (98.27%), sensitivity (98.39%), specificity (98.26%),

Table 3 Performance Metrics for Tea Leaf Disease Detection

Performance measure	Equation
Accuracy	$\frac{TP + TN}{TP + TN + FP + FN}$
Sensitivity or recall	$\frac{TP}{TP + FN}$
Specificity	$\frac{TN}{TN + FP}$
Precision	$\frac{TP}{TP + FP}$
F1-score	$2 \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$

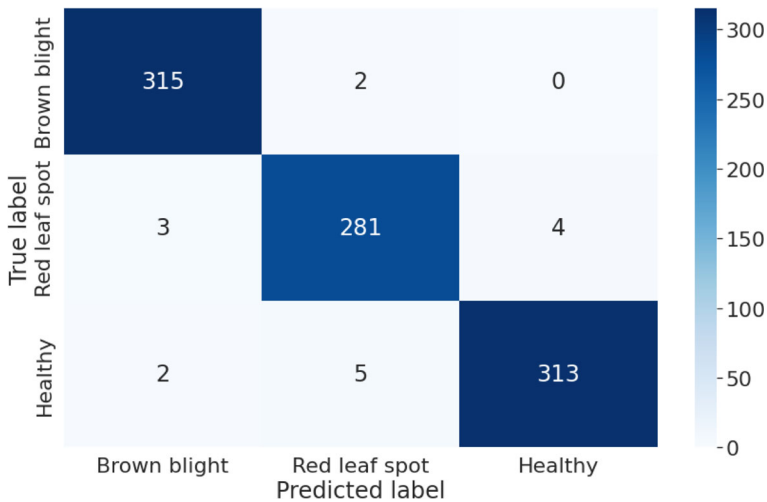


Fig. 4 Confusion matrix showing the classification accuracy of the proposed Res4Net-CBAM model

accuracy (98.35%), and F1-score (98.37%) for tea leaf disease classification was achieved by Res4Net embedding CBAM after each block in the Res4Net network.

The sensitivity and specificity metrics are critical in measuring the model's ability to predict classes accurately. Sensitivity, also known as the true positive rate, is the percentage of diseased leaf images correctly predicted by the classifier, while specificity, also known as the true negative rate, is the percentage of disease-free leaf images correctly predicted by the classifier. These metrics are vital for an automated disease prediction system. Figure 5(a) shows a bar graph illustrating the model's performance in terms of sensitivity and specificity.

The results of our study demonstrate that the proposed Res4net-CBAM model achieved excellent performance, with the highest sensitivity of 98.39% and specificity of 98.26%. We compared our model with five standard CNN techniques (AlexNet, VGG16, ResNet50, DenseNet121, and InceptionV3) on the same dataset, and our proposed model was found to be significantly more accurate and faster. The proposed Res4net-CBAM model achieved the best accuracy of 98.27% due to the use of four deep residual block image features, which enabled more accurate identification of tea leaf disease. Additionally, the CBAM block helped detect the disease location more accurately and improve the model detection accuracy. We also evaluated the performance of our proposed method against other proposed models by the researchers, and the results showed that our model significantly outperformed those models. We used Grad-CAM for visualization, which displayed the areas where the model concentrated its attention. Figure 5(b) shows some of the heat maps generated using Grad-CAM. Table 5 provides a detailed comparative analysis of our model against standard CNN models. On the other hand, Table 6 presents the performance evaluation of our model against existing methods.

5 Discussion

The proposed image feature extraction architecture, which integrates the attention module into residential blocks, is presented in this paper. This architecture is utilized to construct

Table 4 Average Accuracy, Sensitivity, Specificity, Precision, Recall and F1-Score of the proposed Res4net-CBAM model

Optimizer	Learning Rate	Accuracy(%)	Sensitivity(%)	Specificity(%)	Precision(%)	F1-Score(%)
Adam	0.01	95.63	95.56	95.60	95.64	95.62
	0.001	94.25	94.32	94.41	94.38	94.27
SGD	0.01	96.04	96.09	96.15	96.26	96.06
	0.001	96.88	96.87	96.95	96.91	96.88
Adagrad	0.01	97.09	97.11	97.16	97.13	97.06
	0.001	98.27	98.39	98.26	98.35	98.37

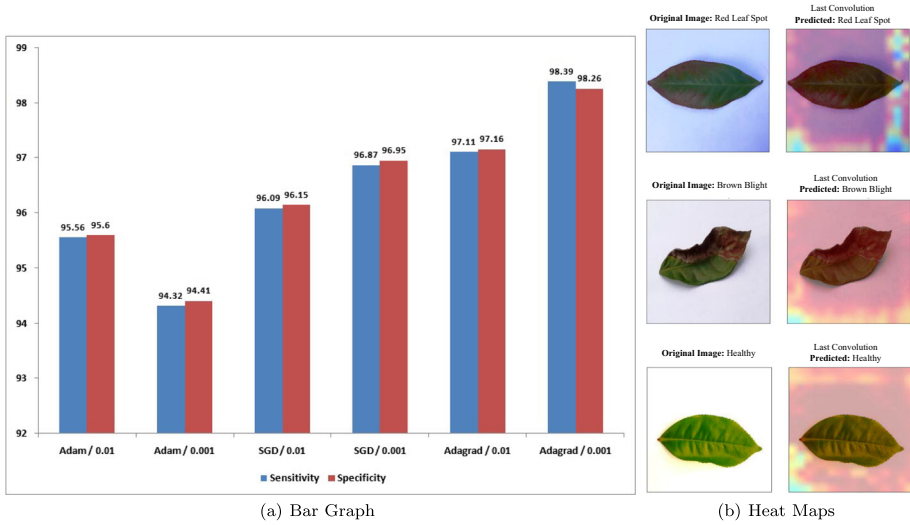


Fig. 5 Res4net-CBAM: (a) Bar Graph in terms of Sensitivity and Specificity (b) Heat Maps

a tea leaf disease classification network. We proposed a new deep residual block-based network architecture called Res4net, and embed CBAM modules in residual blocks in four different regions of Res4Net. The CBAM module in each residual block of Res4net-CBAM helps to reduce feature redundancy and improve the performance of tea leaf disease detection.

Table 5 Comparison analysis with Standard CNNs

Network	Classes	Precision	Recall	F1-Score	Accuracy
AlexNet	Brown Blight	0.95	0.83	0.88	0.938
	Red Leaf Spot	0.98	0.98	0.99	
	Healthy	0.81	0.89	0.85	
VGG16	Brown Blight	0.76	0.92	0.83	0.879
	Red Leaf Spot	0.98	0.99	0.99	
	Healthy	0.85	0.59	0.57	
ResNet50	Brown Blight	0.90	0.85	0.87	0.932
	Red Leaf Spot	0.92	0.99	0.97	
	Healthy	0.99	0.82	0.92	
DenseNet121	Brown Blight	0.91	0.81	0.85	0.915
	Red Leaf Spot	0.90	0.99	0.96	
	Healthy	0.99	0.79	0.91	
InceptionV3	Brown Blight	0.83	0.89	0.86	0.903
	Red Leaf Spot	0.95	0.98	0.97	
	Healthy	0.91	0.66	0.77	
Proposed Res4net-CBAM	Brown Blight	0.98	0.99	0.99	0.9827
	Red Leaf Spot	0.98	0.98	0.98	
	Healthy	0.99	0.98	0.98	

Table 6 Comparison to other Existing Methods

Author and Year	Method	Dataset	Result
Hossain et al. [43], 2018	SVM classifier	self acquired	93% (Accuracy)
Chen et al. [44], 2019	LeafNet algorithm	3810 self acquired field images	90.16%(Accuracy)
Bhowmik et al. [45], 2020	convolutional neural network	2341 self acquired field images	95.93%(Accuracy)
Mukhopadhyay et al. [31], 2021	NSGA-II + Multi-class SVM and PCA	self acquired field images	83%(Accuracy)
Bao et al. [32], 2022	AX-RetinaNet	700 self acquired field images	93.83 (mAP)%
Datta et al. [46], 2023	Deep CNN	5867 self acquired field images	96.56% (Accuracy)
Our Work	Res4net-CBAM	2867 self acquired field images + 330 images from Tea Sickness dataset [35]	98.27%(Accuracy)

The experimental results demonstrate that Res4net-CBAM significantly enhances the disease recognition performance of tea leaf diseases, with an accuracy improvement of 5% compared to Res4net. The study also shows that a shallow network like Res4Net is suitable for final feature extraction in tea leaf disease detection, and adding more residual blocks to the network does not necessarily improve identification performance. It is critical to identify the ideal depth of a deep network for a particular activity. The findings suggest that CBAM modules can improve advantageous features and reduce unwanted features, making it beneficial for improving test accuracy. We recommend future studies to investigate the applicability of CBAM modules in other deep learning models.

In this study, the dataset was classified into three classes: brown blight, red leaf spot, and healthy. The proposed Res4net-CBAM model achieved an impressive average detection accuracy of 98.27%. The model was trained with three different learning optimizers: ADAM, Stochastic Gradient Descent (SGD), and Adagrad, with learning rates set at 0.01 and 0.001. According to Table 4, Adagrad with a learning rate of 0.001 provided the best results for each evaluation metric considered. Additionally, the model's detection speed is fast, meeting the requirements for real-time operation. The study's results were compared to other works discussed in Section 2, and all of these research results fell short of the proposed model's performance. The proposed Res4net-CBAM model performed well overall and achieved a decent rate of detection accuracy for tea leaf diseases.

Despite achieving impressive results, it is important to understand the limitations and possible future directions for tea leaf disease detection. Firstly, the majority of the images used in this research were captured in a supervised condition, which means the trained model may not perform well under complex field situations. Secondly, using multiple image-capturing devices with varying resolutions and aspects is common in real-time applications, and it is necessary to continue developing basic technologies for data collection. Lastly, due to the shortage of computational resources, all images in this study were downsized to 128 x 128, which could result in data loss. However, this could be prevented with additional computational resources. On the other hand, the dataset used in the experiment is specific, and each image contains only one disease. Our proposed model has several advantages over existing techniques, such as improved detection accuracy, lightweight design with fewer parameters, and reduced training and estimation time. Therefore, this study will be useful for researchers in designing efficient networks for portable devices.

6 Conclusion

The present study proposed Res4net-CBAM, a new CNN architecture for automatic recognition of tea leaf diseases. Res4net-CBAM combines deep residual blocks with skip connections and a lightweight CBAM attention module. The proposed model achieved high accuracy rates of 98.27%, 98.39%, 98.35%, and 98.37% for accuracy, recall, precision, and F1-score, respectively, in the three-class tea leaf disease test. With only 2.39 million parameters, the Res4net-CBAM model successfully balanced high precision and a small number of parameters. Additionally, the proposed model outperformed AlexNet and conventional CNN models, making it suitable for use in embedded and mobile devices to help farmers accurately diagnose diseases of the tea leaf. However, this study has some limitations, such as using summertime images of tea leaf disease and a specific dataset containing images with only one disease. Future work will focus on obtaining additional images of tea leaf disease in different climatic and geographic settings, broadening the categories for crop diseases to improve the gener-

alizability of the model, and deploying the proposed model on handheld devices for use in real scenarios.

Funding Funding information is not applicable. No funding was received.

Data Availability Statement The data that support the findings of this study are available on request from the corresponding author. The data are not publicly available due to privacy or ethical restrictions.

Declarations

Conflicts of interest The authors declare that they have no conflict of interest.

References

1. Kaur A, Kaur M, Kaur P, Kaur H, Kaur S, Kaur K (2015) Estimation and comparison of total phenolic and total antioxidants in green tea and black tea. *Global J Bio-Sci Biotechnol* 4(1):116–120
2. Das K (2006) International protection of India's geographical indications with special reference to "darjeeling" tea. *J World Intellect Prop* 9(5):459–495. <https://doi.org/10.1111/j.1422-2213.2006.00300.x>
3. Shah SK, Pate VA et al (2016) Tea production in India: challenges and opportunities. *J Tea Sci Res* 6. <https://doi.org/10.5376/jtsr.2016.06.0005>
4. Tea Borad of India (2022). <https://www.ibef.org/exports/indian-tea-industry>
5. Alzubaidi L, Zhang J, Humaidi AJ, Al-Dujaili A, Duan Y, Al-Shamma O, Santamaría J, Fadhel MA, Al-Amidie M, Farhan L (2021) Review of deep learning: Concepts, cnn architectures, challenges, applications, future directions. *J Big Data* 8:1–74. <https://doi.org/10.1186/s40537-021-00444-8>
6. Ahmed ST, Guptha NS, Lavanya N, Basha SM, Fathima AS et al (2022) Improving medical image pixel quality using micq unsupervised machine learning technique. *Malaysian Journal of Computer Science* pp 53–64
7. Thouheed Ahmed SS, Thanuja K, Guptha NS, Narasimha S (2016) In: 2016 International conference on computing technologies and intelligent data engineering (ICCTIDE'16). pp 1–4. <https://doi.org/10.1109/ICCTIDE.2016.7725324>
8. Guptha NS, Balamurugan V, Megharaj G, Sattar KNA, Rose JD (2022) Cross lingual handwritten character recognition using long short term memory network with aid of elephant herding optimization algorithm. *Pattern Recogn Lett* 159:16–22
9. Vishnoi VK, Kumar K, Kumar B (2021) Plant disease detection using computational intelligence and image processing. *J Plant Dis Protect* 128:19–53
10. Jackulin C, Murugavalli S (2022) A comprehensive review on detection of plant disease using machine learning and deep learning approaches. *Measurement: Sensors* p 100441
11. Ishengoma FS, Rai IA, Ngoga SR (2022) Hybrid convolution neural network model for a quicker detection of infested maize plants with fall armyworms using uav-based images. *Ecol Inform* 67:101502. <https://doi.org/10.1016/j.ecoinf.2021.101502>
12. Bhuyan P, Singh PK, Das SK, Kalla A (2023) Se_spnet: Rice leaf disease prediction using stacked parallel convolutional neural network with squeeze-and-excitation. *Expert Systems* p e13304. <https://doi.org/10.1111/exsy.13304>
13. Lee H, Park J, Hwang JY (2020) Channel attention module with multiscale grid average pooling for breast cancer segmentation in an ultrasound image. *IEEE Trans Ultrason Ferroelectr Freq Control* 67(7):1344–1353. <https://doi.org/10.1109/TUFFC.2020.2972573>
14. Pooja V, Das R, Kanchana V (2017) In: 2017 IEEE technological innovations in ICT for agriculture and rural development (TIAR) (IEEE). pp 130–133
15. Liu B, Zhang Y, He D, Li Y (2017) Identification of apple leaf diseases based on deep convolutional neural networks. *Symmetry* 10(1):11. <https://doi.org/10.3390/sym10010011>
16. Kamrul MH, Rahman M, Robin MRI, Hossain MS, Hasan MH, Paul P (2020) In: Proceedings of the international conference on computing advancements. pp 1–8. <https://doi.org/10.1145/3377049.3377122>
17. Kamilaris A, Prenafeta-Boldú FX (2018) Deep learning in agriculture: A survey. *Comput Electron Agric* 147:70–90. <https://doi.org/10.1016/j.compag.2018.02.016>

18. Boulent J, Foucher S, Théau J, St-Charles PL (2019) Convolutional neural networks for the automatic identification of plant diseases. *Front Plant Sci* 10:941. <https://doi.org/10.3389/fpls.2019.00941>
19. Ma J, Zheng F, Zhang L, Sun Z et al (2018) Disease recognition system for greenhouse cucumbers based on deep convolutional neural network. *Trans Chinese Soc Agric Eng* 34(12):186–192. <https://doi.org/10.3389/fpls.2019.00941>
20. Chao X, Sun G, Zhao H, Li M, He D (2020) Identification of apple tree leaf diseases based on deep learning models. *Symmetry* 12(7):1065. <https://doi.org/10.3390/sym12071065>
21. Szegegy C, Ioffe S, Vanhoucke V, Alemi AA (2017) In: Thirty-first AAAI conference on artificial intelligence. <https://doi.org/10.1609/aaai.v31i1.11231>
22. Szegegy C, Liu W, Jia Y, Sermanet P, Reed S, Anguelov D, Erhan D, Vanhoucke V, Rabinovich A (2015) In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp 1–9. <https://doi.org/10.48550/arXiv.1409.4842>
23. He K, Zhang X, Ren S, Sun J (2016) In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp 770–778. <https://doi.org/10.48550/arXiv.1512.03385>
24. Huang G, Liu Z, Van Der Maaten L, Weinberger KQ (2017) In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp 4700–4708. <https://doi.org/10.48550/arXiv.1608.06993>
25. Hu J, Shen L, Sun G (2018) In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp 7132–7141. <https://doi.org/10.48550/arXiv.1709.01507>
26. Woo S, Park J, Lee JY, Kweon IS (2018) In: Proceedings of the European conference on computer vision (ECCV). pp 3–19. <https://doi.org/10.48550/arXiv.1807.06521>
27. Sun Y, Jiang Z, Zhang L, Dong W, Rao Y (2019) Slic_svm based leaf diseases saliency map extraction of tea plant. *Comput Electron Agric* 157:102–109. <https://doi.org/10.1016/j.compag.2018.12.042>
28. Hu G, Wu H, Zhang Y, Wan M (2019) A low shot learning method for tea leaf's disease identification. *Comput Electron Agric* 163:104852. <https://doi.org/10.1016/j.compag.2019.104852>
29. Hu G, Yang X, Zhang Y, Wan M (2019) Identification of tea leaf diseases by using an improved deep convolutional neural network. *Sustain Comput Inform Syst* 24:100353. <https://doi.org/10.1016/j.suscom.2019.100353>
30. Hu G, Wang H, Zhang Y, Wan M (2021) Detection and severity analysis of tea leaf blight based on deep learning. *Comput Electr Eng* 90:107023. <https://doi.org/10.1016/j.compeleceng.2021.107023>
31. Mukhopadhyay S, Paul M, Pal R, De D (2021) Tea leaf disease detection using multi-objective image segmentation. *Multim Tools App* 80(1):753–771. <https://doi.org/10.1007/s11042-020-09567-1>
32. Bao W, Fan T, Hu G, Liang D, Li H (2022) Detection and identification of tea leaf diseases based on ax-retinanet. *Sci Rep* 12(1):1–16. <https://doi.org/10.1038/s41598-022-06181-z>
33. Lehmann-Danzinger H (2000) Diseases and pests of tea: Overview and possibilities of integrated pest and disease management. *J Agric Tropics Subtropics* 101(1):13–38
34. Keith L, Ko WH, Sato DM (2006) Identification guide for diseases of tea (*camellia sinensis*)
35. Gibson Kimutai AF (2022) Tea sickness dataset
36. Zeng W, Li M (2020) Crop leaf disease recognition based on self-attention convolutional neural network. *Comput Electron Agric* 172:105341. <https://doi.org/10.1016/j.compag.2020.105341>
37. Mi Z, Zhang X, Su J, Han D, Su B (2020) Wheat stripe rust grading by deep learning with attention mechanism and images from mobile devices. *Front Plant Sci* 11:558126. <https://doi.org/10.3389/fpls.2020.558126>
38. Xiao Y, Yin H, Wang SH, Zhang YD (2021) Trec: Transferred resnet and cbam for detecting brain diseases. *Front Neuroinformatics* 15. <https://doi.org/10.3389/fninf.2021.781551>
39. Yanan L, Maofan C, Hongbin M, Ying J (2021) Tablets defect detection based on improved resnet-cbam
40. Chao H, Fenhua W, Ran Z (2019) In: Proceedings of the 2019 international conference on artificial intelligence and advanced manufacturing. pp 1–6. <https://doi.org/10.1145/3358331.3358379>
41. Chen Y, Zhang X, Chen W, Li Y, Wang J (2020) Research on recognition of fly species based on improved retinanet and cbam. *IEEE Access* 8:102907–102919. <https://doi.org/10.1109/ACCESS.2020.2997466>
42. Abadi M, Agarwal A, Barham P, Brevdo E, Chen Z, Citro C, Corrado GS, Davis A, Dean J, Devin M et al (2016) Tensorflow: Large-scale machine learning on heterogeneous distributed systems. [arXiv:1603.04467](https://arxiv.org/abs/1603.04467) <https://doi.org/10.48550/arXiv.1603.04467>
43. Hossain S, Mou RM, Hasan MM, Chakraborty S, Razzak MA (2018) In: 2018 IEEE 14th international colloquium on signal processing & its applications (CSPA). IEEE, pp 150–154. <https://doi.org/10.1109/CSPA.2018.8368703>
44. Chen J, Liu Q, Gao L (2019) Visual tea leaf disease recognition using a convolutional neural network model. *Symmetry* 11(3):343. <https://doi.org/10.3390/sym11030343>
45. Bhowmik S, Talukdar AK, Sarma KK (2020) In: 2020 Advanced communication technologies and signal processing (ACTS). IEEE, pp 1–6. <https://doi.org/10.1109/ACTS49415.2020.9350413>

46. Datta S, Gupta N (2023) A novel approach for the detection of tea leaf disease using deep neural network. *Procedia Comput Sci* 218:2273–2286. <https://doi.org/10.1016/j.procs.2023.01.203>

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

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.