Journal of Applied Data Sciences
Vol. 5, No. 4, December 2024, pp. 2039-2053

Transfer Learning Boosts Ensembles for Precise Sugarcane Leaf Disease Detection

Bappaditya Das[1,], Chandan Das[2,], C S Raghuvanshi[3,*,]

[1,3]*Department of Computer Science and Engineering, Faculty of Engineering & Technology, Rama University, Kanpur, Uttar Pradesh 209217, India*

[2]*Department of Computer Science and Engineering, DR. B. C. Roy Engineering College, Durgapur 713206, India*

(Received: July 25, 2024; Revised: September 02, 2024; Accepted: October 08, 2024; Available online: November 7, 2024)

Abstract

The United Nations' Sustainable Development Goals (SDGs) are committed to ensuring that all individuals have access to sufficient, safe, and nutritious food by 2030, acknowledging that food security is a fundamental right of human survival. However, the exponential growth of the world population raises concerns about the threat of global food insecurity by 2050. An increase in agricultural output is inevitable to meet the growing demand for food. Maximizing agricultural output requires safeguarding crops against disease due to the scarcity of arable land. In the modern age of technology-driven agriculture, the traditional approach of visually detecting agricultural diseases, employed by skilled farmers, is susceptible to inaccuracies and can be a time-consuming process. Transfer learning achieves exceptional accuracy on a noise-free image dataset by using pre-trained CNN models for early crop disease detection. However, their performance significantly deteriorates on datasets with images with complex natural backgrounds. This paper describes an ensemble of transfer learning-based binary classifiers to detect multiple sugarcane leaf diseases using a binary classification tree. Our model successfully classified five distinct sugarcane leaf diseases, achieving an impressive overall validation accuracy of 98.12%, macro-average precision of 97.75%, Recall of 97.93% and F1-score of 97.84%. Moreover, a methodological approach derived from the empirical observations of experienced agricultural experts led to a significant reduction in the computational complexity of our model, transitioning from exponential to linear search space framework.

Keywords: Sugarcane Leaf Disease Detection, Transfer Learning, Ensemble Model, Computer Vision

1. Introduction

Sugarcane, a significant crop contributing to global sugar production and byproducts like syrups and bagasse, has the potential to achieve SDG 2 (Zero Hunger) and SDG 8 (Decent Work and Economic Growth) [1], [2], [3], [4], [5]. It is a cornerstone of modern civilization and crucial for providing necessary resources [1], [2]. The presence of infectious pathogens and extreme climatic conditions are the primary causes of plant diseases. Plant diseases can have a negative impact on the growth, function, and structure of crops, which can seriously affect individual's dependent on them. Reliance on traditional methods for identifying plant diseases can decrease agricultural productivity and increase losses due to their ineffectiveness in detecting diseases in their early stages. Agricultural productivity is a significant driver of economic growth. Agricultural industries rely on accurate identification and classification of plant diseases to improve productivity and economic outcomes [6], [7].

Conventional methods for detecting and categorizing plant diseases are time-consuming, error-prone, require experts, and negatively impact productivity [8]. Accurate classification of plant diseases can boost crop productivity and support various cultivation methods [9]. Researchers developed several image processing, machine learning (ML), and deep learning (DL) techniques to identify and classify plant diseases using plant leaf datasets. DL using Convolution Neural Network (CNN) for plant disease identification has gained attention [10], particularly after the release of the PlantVillage [11] dataset in 2015. Studies on leaf disease detection showed only 1% focusing on sugarcane, compared to 39% for tomatoes and 16% for rice [12]. This disparity highlights a notable gap in the existing academic research

***Corresponding author: C S Raghuvanshi (drcsraghuvanshi.fet@ramauniversity.ac.in)**
DOI: https://doi.org/10.47738/jads.v5i4.396

landscape. Deep transfer learning has improved object detection and image classification using previously acquired knowledge. The outcomes of hyperparameter optimization of multiple pre-trained CNN models for leaf disease detection, such as NASNetMobile, ConvNeXtSmall, DenseNet201, ResNet101, ResNet50, GoogleNet, AlexNet, ResNet18, and EfficientNetB7, are very encouraging [13]. The researchers assessed diverse methodologies for amalgamating multiple pretrained CNNs within ensemble learning techniques to identify various leaf diseases in sugarcane [13], [14]. However, the optimized weighted average ensemble technique showed a significant increase in accuracy [14]. Existing DL models trained on the PlantVillage dataset may face generalization problem to real-world settings due to the dataset's constraints [15]. Despite the economic importance of sugarcane, leaf disease detection in this crop remains relatively understudied. However, the availability of sugarcane leaf data collected under field conditions presents the potential for enhancing detection accuracy. This research can enhance the sustainability and efficiency of sugarcane production by addressing existing research gaps and limitations. The major contributions of this research are outlined below.

Binary classification approach: The multiclass classification problem was reformulated into a binary classification framework, effectively reducing the search space from exponential to linear and decreasing dependencies on labeled training datasets.

New activation function: We introduced a new activation function named "msswish" and investigated its effectiveness within a customized TL-based CNN architecture.

Efficient model selection: Through comprehensive empirical investigation involving transfer learning (TL) using various pre-trained CNN architectures in conjunction with diverse activation functions, we identified the optimal combination.

Enhanced model’s reliability and superiority: In order to enhance the model's relevance in practical situations and address class imbalance, the dataset of real-time field images was enriched with a variety of additional field images from multiple sources at every stage, while data augmentation techniques were also implemented. Empirical results indicate that the proposed model outperforms the majority of the TL and existing ensemble models in accurately classifying sugarcane leaf diseases. It harnesses the capabilities of top-performing CNNs at each level while circumventing the need for hyperparameter tuning.

Following the introduction, this research organizes the remaining portion into four sections. The “Related Literature” section provides insights into the existing research on TL-based plant disease classification and detection, including classification methods, crop types, and accuracy. The “Material and Methods” section describes the datasets, the leaf diseases under study, and the proposed methodology. The “Results and Discussion” section demonstrates the findings and presents related discussions. These illustrate the model’s performance evaluations and comparisons with existing models that address the same problem. Finally, Section “Conclusion” concisely summarizes our proposed approach and future research directions.

2. Related Literature

Conventional ML focuses on feature engineering and segmentation strategies, while DL techniques involve directly assimilating insights from raw data. In [16] three TL models, ResNet-18, ShuffleNet, and MobileNet, were utilized with six ML classifiers to identify ten types of tomato leaf diseases using the PlantVillage dataset. The experiment yielded an accuracy of 99.9%. The TL approach using ResNet50 and SVM achieved an f1 score of 0.9838 among 11 pre-trained CNN (AlexNet, VGG16, VGG19, GoogleNet, ResNet18, ResNet50, ResNet101, InceptionV3, InceptionResNetV2, DenseNet201 and XceptionNet) models to identify four rice leaf diseases on a dataset of 5932 on-field leaf images [17]. In [18], GoogLeNet and DenseNet emerged as the most proficient classifiers, while ResNet101 and AlexNet exhibited slightly lower accuracy in classifying five major leaf diseases on a private eggplant dataset without providing the total number of images within the dataset. The RGB images showed the highest classification accuracy compared to gray, HSV, and YCbCr color spaces. However, in the YCbCr color space, VGG16 with MSVM achieved a classification accuracy of 99.4%, which is very close to GoogLeNet in the RGB color space. State-of-the-art (SOTA) models like VGG19, ResNet152, DenseNet169, Inception-NetV3, and MobileNetV2 can accurately identify wheat rust diseases with upto 97.8% accuracy on the WheatRust21 dataset [19]. The dataset contains 6556 images of healthy leaves and three classes of rust-diseased leaves from natural field conditions. The fine-tuned EfficientNet B4 model achieved a testing accuracy of 99.35%, the highest among eight variants of EfficientNet architecture.

A study [20] proposed a hybrid DL architecture that combines a CNN with a convolutional attention module (CBAM) and a SVM to detect and classify tomato plant leaf diseases early. This lightweight and efficient model facilitates straightforward installation on any farmer-operated smart device possessing a digital camera and the requisite processing capabilities. 99.6% accuracy achieved for tomato leaf disease classification using an ensemble model [21] of MobileNetV3Small, EfficientNetV2L, InceptionV3, and MobileNetV2. The hyperparameters were fine-tuned using PSO. Shovon, Md Sakib Hossain, et al. [22] have introduced a reliable DL model named PlantDet, which employs InceptionResNetV2, EfficientNetV2L, and Xception. On a complex dataset, it outperforms the previous SOTA model in classifying the five most common rice leaf diseases with higher accuracy, precision, recall, f1 score, and specificity. Applying a majority voting strategy, the proposed EC [23] used InceptionV3, MobileNetV2, and DenseNet121 to distinguish between rice leaf diseases on a laboratory-based dataset with 96.42% accuracy.

Upadhye et al. [24] designed a CNN that accurately distinguishes healthy and diseased sugarcane leaves with an accuracy of almost 98.7%. A DL model with multilayer perceptron architecture achieved over 99% accuracy in binary classification of healthy and red-rot infected sugarcane leaves [25]. TL models, particularly InceptionV4, AlexNet, ResnetV2-152, and VGG16, achieved impressive accuracies of 99.61%, 99.24%, 99.23%, and 98.88%, respectively, when classifying sugarcane diseases using a dataset of 24000 leaf images [26]. A stacked ensemble model comprising of two CNNs, one incorporating level-wise spatial attention, struggled to achieve an accuracy of 87% after 50 epochs when trained on a dataset of 2,569 leaf images divided into five classes [27]. The authors [14] developed the 'SugarcaneNet24' EC to identify sugarcane leaf diseases using seven pre-trained CNNs and a grid search technique to assign the optimal weight combination. Cuimin Sun et al. [28] introduced a hybrid neural network architecture termed SE-ViT, amalgamating Transformer and CNN with SE attention modules to diagnose sugarcane leaf diseases. Despite achieving an impressive accuracy of 97.26% on the PlantVillage dataset, the model's performance experienced a slight decline, dropping below 90% when evaluated on the private dataset. The EnC-SVMWEL model [29] integrates DenseNet201 with a novel SVMWEL classifier. The model achieved a classification accuracy of 97.45% in recognizing between five distinct sugarcane leaf classes.

Garg et al. [30] developed a hybrid CNN-LSTM that accurately differentiated healthy and diseased sugarcane leaf images with more than 98% accuracy in binary detection. However, the model's accuracy in determining the severity of brown-spot disease falls below 94%. Another hybrid CNNLSTM model [31] achieved slightly greater than 94% accuracy in predicting the severity of downy mildew disease. Researchers also investigated the effectiveness of hybrid CNNSVM models in classifying the severity of sugarcane leaf diseases. The models achieved promising results, with an accuracy of 81.53% for grassy-shoot disease [32] and reaching approximately 98% accuracy for smut disease [33]. Binary classification trees can address multi-class classification problems by creating a binary classifier for every unique pair of classes in the dataset, breaking down the multiclass problem into binary sub-problems [34]. This requires $\binom{k}{2} = k(k-1)/2$ binary classifiers for a k-class classification problem.

3. Material and Methods

3.1. Datasets

In this study, we investigated five different varieties of sugarcane leaves. We examined healthy leaves and four diseased leaves: rust, red rot, mosaic, and yellow (as shown in figure 1). Our primary objective is to identify diseases in sugarcane leaves collected from fields with complex backgrounds rather than controlled laboratory conditions to ensure the proposed model's applicability in real-world scenarios. The paucity of large-scale, real-world datasets continues to be a substantial impediment to advancements in various machine learning applications. Through an extensive search, we successfully integrated a comprehensive collection of field images into our study. We strategically used them at various stages to improve the model's reliability. We gathered 2521 sugarcane leaf images from [35], including 522 healthy, 514 rust, 518 red rot, 462 mosaic, and 505 yellow diseased. We have assembled an additional 1018 images of healthy sugarcane leaves: 430 from [36], 488 from [37], and 100 from [38]. We compiled a total of 314 leaf images

 ISSN 2723-6471

that are infected with rust from [36], along with an additional 75 images from [39]. Moreover, we have /collected 100 images of red-rot diseased leaves from [40] and 73 from [39].

The datasets [35], [36] of RGB sugarcane leaf field images captured in Maharashtra, India, are stored in the reliable Mendeley repository. Sugarcane Leaf Disease Dataset [35] encompasses 2521 images of varying sizes, reflecting the inherent diversity in smartphone camera configurations. The Sugarcane Leaf Dataset [36] includes 6748 high-resolution JPEG images of sugarcane leaves, each with dimensions of 768×1024 pixels. Image acquisition was conducted during daylight hours between April and June through field/farm visits using a Samsung Galaxy F 23 5G.

Figure 1. Sample image of sugarcane leaf from each class

Android mobile equipped with a 50-megapixel (f/1.8) Sony IMX 582 1/2″ sensor camera. Data collection involved capturing images of sugarcane leaves in their natural habitat and detached or severed leaves from a distance of 30–50 cm to ensure a diverse and representative dataset. The previous research [41], [42] established that the optimal split ratio for constructing machine-learning models is 70:30. For model development and evaluation, we pre-processed the dataset and then partitioned it into three mutually exclusive subsets: training, validation, and testing, in a ratio of 70:20:10.

3.2. SOTA CNN Model's Architecture

A schematic of the Transfer Learning Framework for our proposed model (figure 2) typically outlines how various components interact to facilitate this process.

Traditional CNNs with L layers have L connections, each connected only to its immediate successor. However, in the DenseNet [43] architecture, each layer receives input feature maps from all preceding layers in the feed-forward path. The output feature maps of each layer serve as inputs for all subsequent layers. As a result, the complex network structure of DenseNet leads to a total of L (L + 1) / 2 direct connections, as illustrated in figure 3. DenseNet121, DenseNet169, and DenseNet201 are three DenseNet variants based on the network's depth. Each variant consists of four dense blocks. DenseNet121 is the shallowest variant with 121 layers, DenseNet169 has a moderate depth of 169 layers, and DenseNet201 boasts the deepest architecture with 201 layers. DenseNet201 has nearly double the parameters of DenseNet121 as the number of layers increases.

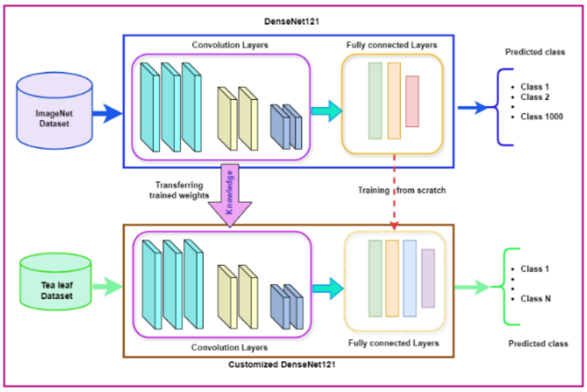


Figure 2. Schematic of the TL Framework for Proposed model

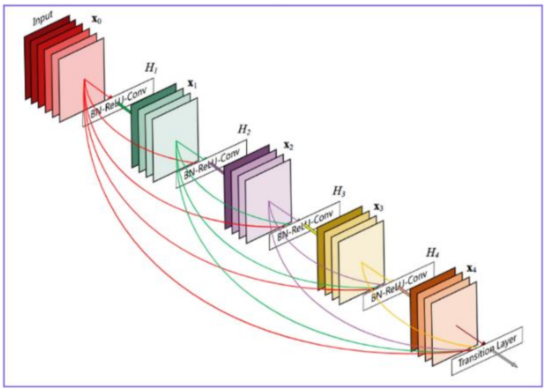


Figure 3. DenseNet architecture (Source: [43])

MobileNetV2 utilizes depthwise separable convolutions as a fundamental architectural element. It decomposes standard convolutions into depthwise convolutions for feature extraction and pointwise convolutions to combine features, reducing computational costs. Introducing low-dimensional intermediate layers between standard convolutional layers can effectively reduce the channel's dimension without compromising accuracy. Furthermore, the inclusion of inverted residual blocks inside bottleneck layers enables efficient processing, rendering it well-suited for mobile devices.

NASNetMobile uses a unique cell-based architecture developed through Neural Architecture Search (NAS). The search space covers various convolutional operations, including standard 3x3 convolutions, depthwise separable convolutions, 1x1 pointwise convolutions, and average pooling. The NAS algorithm assesses various operation combinations within a cell, serving as a fundamental building block of the network. NASNetMobile creates a highly efficient and adaptable architecture by stacking these optimized cells. The inception module serves as the fundamental building block of the InceptionV3 model. Each module uses parallel processing of filters with diverse kernel sizes to extract features at different granularities. For example, 1×1 convolutions reduce dimensions by capturing channel-wise relationships, while 3×3 and 5×5 convolutions capture local spatial information at increasing scales.

InceptionResNet is a hybrid CNN model that amalgamates the concepts of residual connections from ResNet with the Inception architecture to address the vanishing gradient problem and facilitate the training of deeper networks. Inspired by the success of Inception, the Xception model replaces the standard inception modules with depthwise separable convolutions.

3.3. TL-based Proposed Model

Let f_P is a CNN model that has been trained on a specific task denoted as P, with the purpose of classifying images from the "ImageNet" dataset. $\mathbb{X}_P$ represents the input space of task P, which is a high-dimensional vector space containing images. Each dimension of $\mathbb{X}_P$ represents a specific feature or pixel intensity of the image. $\mathbb{Y}_P$ is the output space and contains 1000 image labels. (X_P, Y_P) represents a pair of $\mathbb{X}_P \times \mathbb{Y}_P$-valued random variables conforming to the target images' empirical distribution and their respective labels. We can represent the learning objective for task P mathematically as:

$$\min\{\mathcal{L}_P(f_P) | f_P \in G_P\} = \min\{\mathcal{E}[L_P(Y_P, f_P(X_P; \theta_P))] | f_P \in G_P\} \quad (1)$$

where $\mathcal{L}_P(f_P)$ is the loss function associated with the model $f_P : \mathbb{X}_P \to \mathbb{Y}_P$ for the task P and GP is the set of pre-trained models such that:

$$\forall f_P \left(f_P \in G_P \to \left(\forall X\,(X \in \mathbb{X}_P \to f_P(X) \in \mathbb{Y}_P)\right)\right) \quad (2)$$

Training a DL model from scratch typically requires extensive data and computational power, increasing the risk of overfitting. Transfer learning addresses this challenge by leveraging knowledge from a pre-trained model on a related source task (P) for a new target task (Q). Successfully addressing the domain shift phenomenon is paramount for achieving optimal outcomes when utilizing f_P with target inputs $X_Q \in \mathbb{X}_Q$. We employ the pre-trained model f_P (X_P ; θ_P) as a feature extractor for the input data XQ by replicating the architecture of the initial layers of model f_P. The optimization can be formulated as:

$$\min_{\theta_{PQ}} = \{L_Q(Y_Q, f_{PQ}(\psi(X_Q); \theta_{PQ})) | Y_Q \in \mathbb{Y}_Q, X_Q \in \mathbb{X}_Q\} \quad (3)$$

Where $\psi()$ represents the feature extraction function, $f_{PQ}()$ is the adaptive model parameterized by θ_{PQ}, and L denotes the loss function. In this study, we modified eight pre-trained models, including DenseNet121, DenseNet169, DenseNet201, MobileNetV2, InceptionV3, InceptionResNetV2, Xception, and NASNetMobile. We replaced the classification layers of these models with our custom architecture. As illustrated in the "Customized Pre-trained Models" section of Figure 3, the new architecture consisted of a sequential stack of three dense layers interspersed with two dropout layers. A dense layer, synonymous with a fully connected layer, establishes a comprehensive network of connections by linking each neuron in the current layer to every neuron in the preceding layer. The equation:

$$\hat{y} = \varphi(W.x + b) \quad (4)$$

represents the output of the dense layer where φ denotes the activation function, W denotes the weight matrix, x represents the input vector, and b is the bias vector.

ISSN 2723-6471

Rectified Linear Unit (ReLU) activation functions are prevalent in the hidden layers of SOTA CNNs designed for classification tasks. But ReLU is not differentiable at zero. Recent scholarly investigations have demonstrated the superiority of the Swish activation function over ReLU within the context of image classification tasks grounded in deep learning methodologies [44]. Swish has been effectively utilized in various architectures, including ResNet-18 for diabetic retinopathy detection [45], DenseNet121 and MobileNetV2 for plant disease identification [46], [47]. In our customized pre-trained CNN, we employed a hybrid activation strategy. The first dense layer utilized the sigmoid function to capture non-linear relationships effectively. For the second dense layer, we devised and executed the msswish (modified scaled swish) activation function, based on the principles of Swish. This decision was motivated by the necessity to enhance model performance in complex classification tasks. The msswish function is defined by equation 5:

$$\xi(x) = \frac{(1+\sigma(x)) \cdot x}{2} \tag{5}$$

Where:

$$\sigma(x) = \frac{1}{1+e^{-x}} \tag{6}$$

and the derivative of msswish is defined as:

$$\xi'(x) = \frac{\sigma(x) \cdot (1-\sigma(x)) \cdot x}{2} + \frac{1+\sigma(x)}{2} \tag{7}$$

In our customized pre-trained CNN, we used the sigmoid activation function in the first dense layer and employed msswish in the second.

$$\hat{y}(x) = \xi(W_2 \cdot (\sigma(W_1 \cdot x + b_1)) + b_2) \tag{8}$$

The left part of figure 4. represents the graph for msswish and the right for the first derivative of msswish function, respectively.

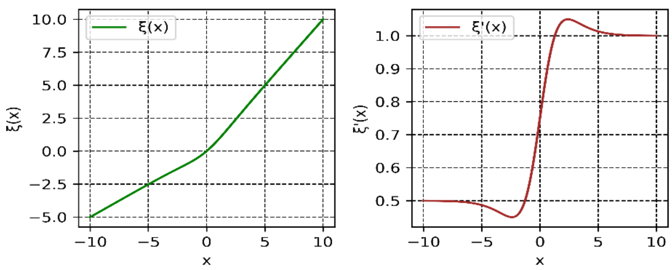


Figure 4. Proposed msswish activation function and its derivative

Based on the literature survey conducted in Section 2, binary classifiers outperform multiclass classifiers in terms of accuracy. Figure 5a examines the selection of the best CNN model from a set of customized CNN models for a binary image classification task. We chose a dataset containing images belonging to two target classes. The initial evaluation focused on the dataset's size to ensure it provided a reasonable amount of data for training a neural network.

We collected additional images for the target classes from reliable sources to support deep learning's data-intensive nature. This approach aimed to create a more evenly distributed dataset, ensuring adequate representation of both groups. All images were then scaled to a uniform size of $3 \times 128 \times 128$ pixels to standardize them for the CNN architecture. The ImageDataGenerator class provided by Keras was utilized to apply a range of transformations on the training images, including rotation, shearing, zooming, flipping, and shifting. These augmentations helped to reduce overfitting and improve the model's ability to handle variations in real-world data. We finally partitioned the dataset into three mutually exclusive subsets for model evaluation.

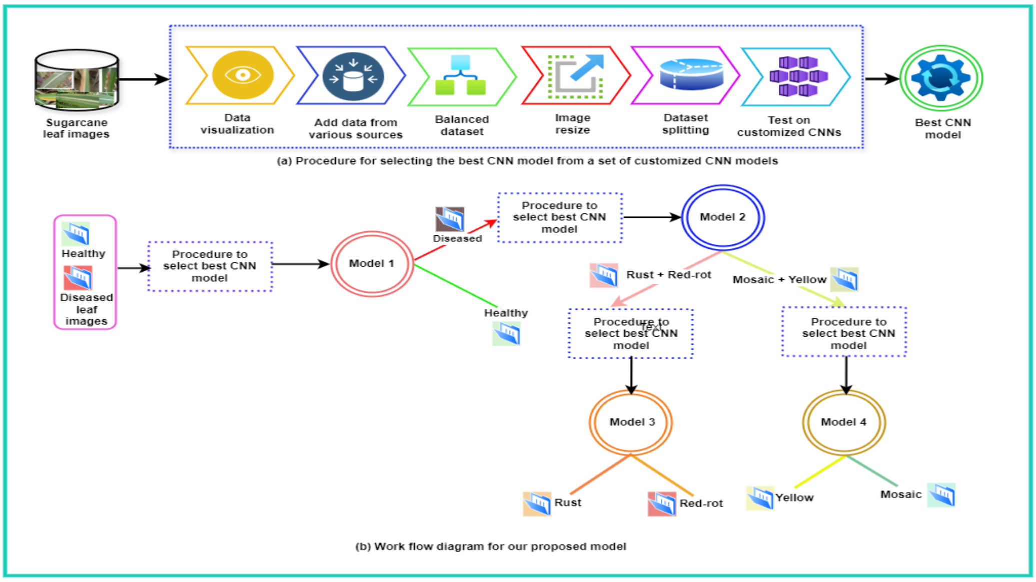


Figure 5. Process flow diagram of our proposed model

Figure 5 illustrates the complete workflow for our proposed model to classify sugarcane leaf disease. Motivated by human psychology, we initially applied CNN model as a binary classifier to ascertain whether the input leaf image diseased or appeared healthy. Diseased leaves were divided into two categories based on different symptomatic characteristics. The first one encompasses leaves exhibiting symptoms of rust or red rot. Rust is identified by raised, pustule-like structures on the leaf surface, often appearing orange or brown. Red rot displays sunken lesions that may appear reddish-brown or purplish. The second class included leaves displaying manifestations consistent with mosaic or yellow disease. Mosaic diseases are characterized by heterogeneously distributed discolorations across the leaf surface, often manifesting as mottled or patterned variations in pigmentation. Conversely, yellow diseases typically induce a uniform yellowing of the leaf tissue, encompassing a generalized loss of chlorosis. The established classification scheme was subsequently employed to differentiate between diseases within each significant class. Leaves initially classified as Class 1 were further analyzed to differentiate between those showing symptoms of rust and those displaying signs of red rot. Likewise, Class 2 leaves were examined in more detail to separate those affected by mosaic disease from those with symptoms of yellow leaf disease. The pseudocode of our proposed algorithm for sugarcane leaf disease classification is given below.

Pseudocode for Sugarcane Leaf Disease Classification

Input	:	Sugarcane leaf image		
Output	:	The predicted disease class		
Stage 1	:	Healthy vs. Diseased Classification		
		predicted_class = model1.predict(image)		
		If (predicted_class == "Healthy")	then	Return "Healthy"
			else	Proceed to Stage 2.
Stage 2	:	Diseased Classification		
		predicted_class = model2.predict(image)		
		If (predicted_class == "Rust-Redrot")	then	Proceed to Stage 3.
			else	Proceed to Stage 4.
Stage 3	:	Rust vs. Redrot Classification		
		predicted_class = model3.predict(image)	return	Predicted_class

ISSN 2723-6471

Stage 4	:	Mosaic vs Yellow Classification		
		predicted_class = model4.predict(image)	return	Predicted_class

We then trained the customized pre-trained CNN models on the prepared dataset. The model with the highest validation accuracy was chosen as optimal. If multiple models achieved the same accuracy, the one with the lowest validation loss was preferred. This approach highlights the importance of models prioritizing accurate classification and showing improved confidence in their predictive capabilities.

3.4. Experimental Setup

This research utilized a computational infrastructure consisting of a laptop equipped with an AMD Ryzen 5 5600H 1600X six-core processor, an NVIDIA GeForce GTX 1650 GPU, and a 64-bit Windows 11 operating system. The deep learning framework employed was TensorFlow with Keras, running on Python 3.9.12 and utilizing CUDA 11.6. All experiments were conducted on the Kaggle Accelerator's GPU P100 configuration. The model was trained using 50 epochs, a batch size of 16, a learning rate of 0.001, the Adam optimizer, and sparse categorical cross entropy as loss function. Early stopping was implemented to prevent overfitting, monitoring validation accuracy for termination.

4. Results and Discussion

4.1. Performance of SOTA CNN Models

We evaluated several SOTA-CNNs, namely DenseNet121, DenseNet169, DenseNet201, MobileNetV2, InceptionV3, InceptionResNetV2, Xception, and NASNetMobile, on the 'Sugarcane Leaf Disease Dataset.' These models were paired with various activation functions, including sigmoid followed by ReLU, sigmoid followed by swish, and sigmoid followed by msswish.

Table 1 summarizes the model's performance on both [35] and the merged datasets, presenting average classification accuracy and per-class identification accuracy. Within table 1, shaded rows correspond to results from dataset [35], while unshaded rows for the merged dataset. We combined 3437 images from external sources (detailed in Section 3.1) with 2521 images [35] belonging to the same five target classes. A subset of 4171 was designated exclusively for training SOTA CNN models.

Table 1. Performance of customized TL models with different activation function on two datasets

CNN model	Activation function	Average accuracy (%)	Healthy (%)	Mosaic (%)	Red rot (%)	Rust (%)	Yellow (%)
DenseNet121	ReLU	96.6	100	97	96	94	96
	ReLU	94.64	96	96	89	99	92
	Swish	96.35	100	97	98	91	96
	Swish	94.86	96	95	91	93	96
	msswish	94	97	90	96	91	96
	msswish	95.98	98	95	98	94	95
DenseNet169	ReLU	93.6	95	87	98	97	91
	ReLU	93.97	97	90	85	97	96
	Swish	96.4	97	93	96	100	96
	Swish	96.42	97	95	100	93	97
	msswish	95.8	100	93	98	97	91
	msswish	94.86	97	92	98	93	95
DenseNet201	ReLU	94.8	100	93	93	91	96
	ReLU	95.53	97	95	100	93	94
	Swish	95.4	100	97	96	91	93
	Swish	95.53	96	94	98	93	97
	msswish	95.4	100	97	96	91	93
	msswish	95.31	98	96	98	93	92
MobileNetV2	ReLU	87.4	89	97	89	80	82

	ReLU	92.18	97	92	83	89	93
	Swish	87.8	92	90	91	86	80
	Swish	93.08	96	94	89	94	94
	msswish	86.2	97	93	91	77	73
	msswish	93.75	95	93	89	94	95
InceptionV3	ReLU	88.54	87	86	88	92	89
	ReLU	93.75	98	94	96	90	91
	Swish	92.18	92	93	91	89	96
	Swish	94.19	97	93	100	91	92
	msswish	87.4	95	87	91	77	87
	msswish	93.08	97	92	87	96	91
InceptionResNetV2	ReLU	89.34	95	97	87	86	82
	ReLU	94.64	97	94	91	93	95
	Swish	91.74	95	100	91	86	87
	Swish	94.41	97	91	94	94	95
	msswish	92.4	97	97	91	86	91
	msswish	95.08	97	90	94	96	97
Xception	ReLU	80.2	89	81	79	82	68
	ReLU	87.72	91	93	89	81	83
	Swish	81.25	89	81	74	89	70
	Swish	87.94	92	90	85	80	88
	msswish	83.85	89	92	76	89	70
	msswish	87.5	91	92	85	84	83
NASNetMobile	ReLU	75	64	75	97	63	81
	ReLU	87.5	93	94	70	77	90
	Swish	83.85	98	72	85	71	89
	Swish	83.03	98	55	66	91	94
	msswish	82.81	94	53	82	84	97
	msswish	93.75	99	89	94	87	97

Table 2 presents the classification accuracy of eight custom pre-trained convolutional neural networks (CNNs) using various activation functions. The best-performing model from each stage was selected as a binary classifier to distinguish between specific leaf conditions, including healthy vs. diseased, Rust-Redrot vs. Mosaic-Yellow, Rust vs. Redrot, and Mosaic vs. Yellow.

Table 2. TL-based binary classifier's accuracy for identifying sugarcane leaf disease

	DenseNet-121	**DenseNet-169**	**DenseNet-201**	**MobileNet-V2**	**Inception-V3**	**Inception-ResNetV2**	**Xcepti-on**	**NASNet-Mobile**
Healthy vs. Diseased leaf (ReLU)	97.65%	99.21%	98.04%	94.92%	96.10%	97.26%	94.53%	95.31%
Healthy vs. Diseased leaf (Swish)	98.04%	98.04%	98.82%	94.14%	95.70%	98.04%	95.70%	96.87%
Healthy vs. Diseased leaf (ms-swish)	97.65%	98.43%	99.60%	94.14%	98.04%	96.87%	95.31%	97.26%
Rust-Redrot vs. Mosaic-Yellow (ReLU)	97.39%	96.57%	96.35%	95.31%	96.87%	97.39%	91.65%	93.75%
Rust-Redrot vs. Mosaic-Yellow (Swish)	97.39%	97.91%	97.91%	96.35%	96.35%	96.35%	92.60%	94.27%
Rust-Redrot vs. Mosaic-Yellow (msswish)	97.91%	98.43%	97.91%	94.27%	96.35%	96.35%	92.70%	94.27%
Rust vs. Redrot (ReLU)	99.20%	98.41%	99.20%	97.61%	97.61%	96.82%	96.03%	95.23%
Rust vs. Redrot (Swish)	96.82%	98.41%	98.41%	96.82%	96.03%	97.61%	97.61%	83.24%
Rust vs. Redrot (msswish)	98.50%	99.20%	98.41%	96.82%	95.23%	98.41%	92.85%	96.82%
Mosaic vs. Yellow (ReLU)	98.55%	98.55%	98.55%	93.23%	98.06%	97.58%	96.13%	91.88%
Mosaic vs. Yellow (Swish)	99.03%	99.03%	99.51%	96.61%	97.58%	98.55%	96.13%	94.90%
Mosaic vs. Yellow (msswish)	99.51%	98.55%	99.51%	97.58%	96.61%	98.06%	96.61%	95.12%

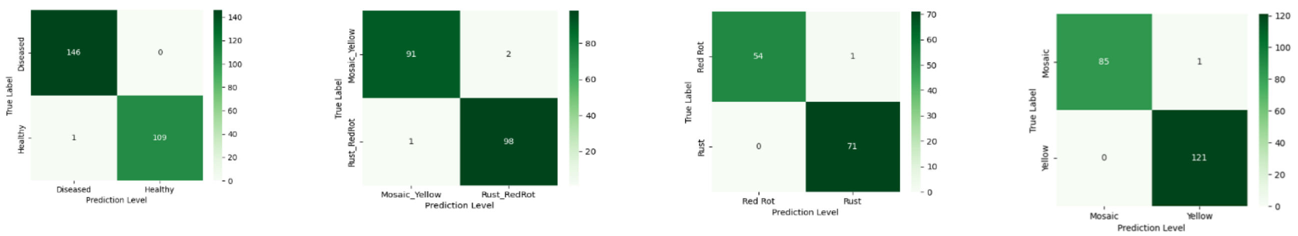
Journal of Applied Data Sciences
Vol. 5, No. 4, December 2024, pp. 2039-2053
ISSN 2723-6471

We conducted paired-sample t-tests on datasets with varied image distributions to understand how activation functions (ReLU, swish, and msswish) impact the model's performance across different classes. This approach allowed us to assess if the choice of activation function significantly affects the accuracy of class-specific classification tasks while also considering potential biases from diverse data distributions. Table 3 shows the t-test results for the dataset [35] and the merged dataset.

Table 3. Statistical Comparison of Activation Functions

Activation Function	T-statistic	P-value
ReLU	2.3877	0.0219
swish	1.2625	0.2143
msswish	2.9522	0.0053

Figure 6 presents a series of confusion matrices generated by the model achieving the highest classification accuracy. These confusion matrices presented provide insights into the models' ability to distinguish between healthy and diseased leaves (figure 6a) and differentiate between specific disease categories (figure 6b, figure 6c, figure 6d). Figure 6(a): DenseNet201with msswish activation function effectively classifies healthy versus diseased leaves. Figures 6 (figure 6b, figure 6c, figure 6d) focus on differentiating between specific disease classes (Rust-Redrot vs. Mosaic-Yellow, Rust vs. Redrot, Mosaic vs. Yellow).



a. b. c. d.

Figure 6. Confusion matrix generated by best CNN

Figure 6, figure 7, figure 8, figure 9 and figure 10 present the results of our investigation, which employed a customized DenseNet121 architecture to classify sugarcane leaves as either healthy or infected with red rot. Based on the validation loss curves shown in figure 7, figure 8, figure 9, the DenseNet121 model using the msswish activation function achieved the lowest validation loss compared to models utilizing ReLU and Swish activations. In scenarios where models exhibit similar accuracy, the DenseNet121 with msswish activation is thus preferred due to its superior loss performance, suggesting enhanced generalization and robustness. Figure 10 presents the confusion matrix generated by our proposed model to classify healthy and Red rot-infected leaves. This matrix comprehensively assesses the model's performance, including true positive, true negative, false positive, and false negative rates.

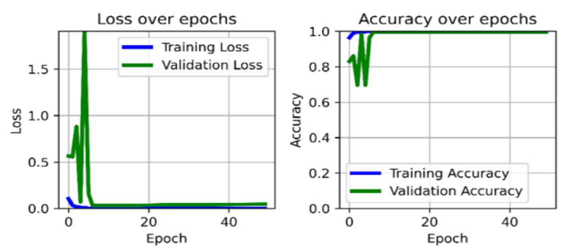


Figure 7. DenseNet121 with Swish

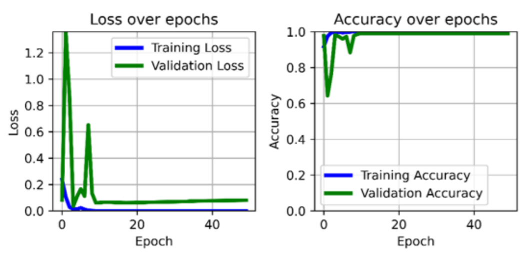


Figure 8. DenseNet121 with ReLU

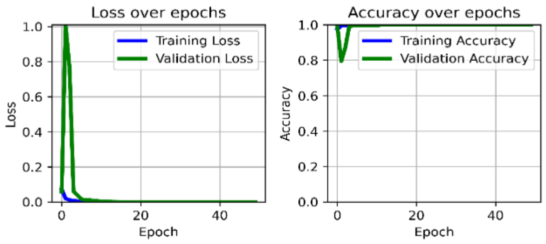


Figure 9. DenseNet121 with msswish

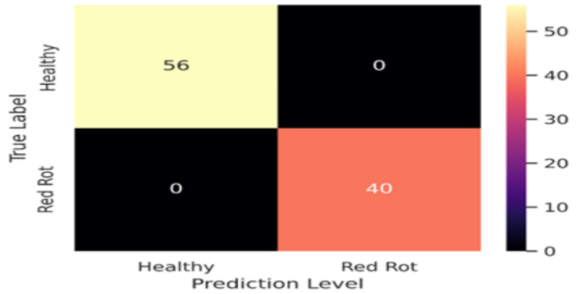


Figure 10. CM: Healthy vs Red Rot

Figure 11 demonstrates that our proposed method outperforms customized TL using ReLU, swish, and msswish activation functions consistently.

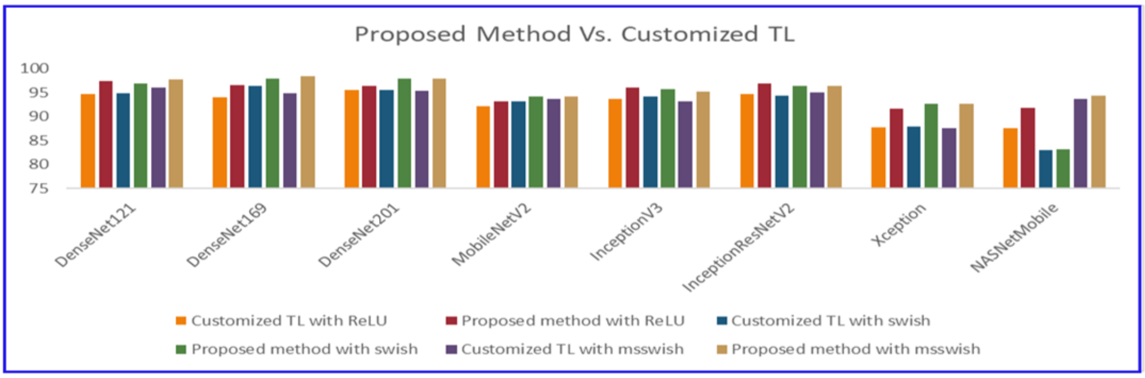


Figure 11. Performance of proposed method with customized TL using various activation function

4.2. Discussions

An analysis of table 2 shows that models using the msswish activation function outperform others in classifying healthy leaves in most cases on merged dataset. The statistical results (table 3) show that the msswish activation function has the highest absolute t-statistic value (2.9522), indicating the most significant performance difference compared to other activation functions. The observation is supported by the p-value (0.0053), which is below the significance level of 0.05. This statistically significant result ($p < 0.05$) demonstrates that msswish performs differently. ReLU also displays a statistically significant difference based on the p-value (0.0219). However, its t-statistic (2.3877) is lower in magnitude than msswish, indicating a potentially weaker effect on performance. Conversely, swish exhibits the lowest evidence for a significant difference. Its t-statistic (1.2625) and relatively high p-value (0.2143) suggest that Swish's performance might be similar to ReLU based on these metrics.

The results presented in table 2 suggest that utilizing DenseNet architectures with various activation functions holds promise for accurately classifying sugarcane leaf diseases. DenseNet201 with msswish activation function (figure 6(a)) achieved an average validation accuracy of 99.60% and demonstrated precise classification performance. It reached 100% accuracy in identifying diseased leaves and 99% in identifying healthy leaves. This surpasses the benchmark established by the authors' own CNN model, which attained an accuracy of 98.69% [24]. DenseNet169 with msswish activation function attained an average validation accuracy of 98.43% in distinguishing between Mosaic-Yellow and Rust-Redrot diseased leaves, with a 98% accuracy in identifying Mosaic-Yellow and 99% accuracy in identifying Rust-Redrot diseased leaves. The DenseNet169 architecture, combined with the msswish activation function, performed incredibly well in distinguishing between Rust and Redrot diseased leaves, achieving an average accuracy of 99.20%. It showed flawless accuracy (100%) in identifying rust disease and achieved a high accuracy (98%) in recognizing redrot disease.

Based on validation loss, DenseNet121 and DenseNet201 architectures employing the ReLU activation function were not pursued further. The DenseNet169 architecture with the msswish activation function exhibited superior performance in this metric. The evaluation results revealed that the DenseNet121 architecture and the msswish activation function achieved a noteworthy average validation accuracy of 99.51% in differentiating between Mosaic and Yellow diseased leaves. Furthermore, it exhibited perfect accuracy (100%) in classifying Yellow disease while demonstrating very high accuracy (99%) in identifying Mosaic disease. Our investigation identified DenseNet121 as the optimal model architecture because of its lower validation loss and fewer parameters compared to DenseNet201 for classifying mosaic and yellow diseased leaves. The initial stage demonstrated exceptional performance, achieving a classification accuracy of 99.6% for healthy instances. The cumulative probabilities of disease progression to subsequent stages were calculated by multiplying the stage-specific accuracies. For instance, the likelihood of progressing to Stage 2 (Diseased) was ascertained by multiplying the accuracy of Stage 1 (0.996) with the accuracy of Stage 2 (0.9843). Similarly, the probabilities of advancing to Stage 3 (Rust-Redrot) and Stage 4 (Mosaic-Yellow) were derived by multiplying the accuracies of all prior stages. The macro-average accuracy of the model, calculated by finding the mean of these cumulative probabilities, was 98.12%. The model exhibited outstanding performance, attaining classification accuracies of 99.2%, 98%, 96.43%, 98%, and 99% for the classes Healthy, Rust, Red rot, Mosaic, and Yellow respectively.

Significantly, our model exceeded the accuracy of 87% [27] and 97.45% [29] achieved by previous research. It attained a validation accuracy of 97.87% in classifying three separate groups of sugarcane leaves, surpassing the 97.78% accuracy recorded by DenseNet201 with SVM [48]. Furthermore, proposed model outperformed a previously published custom-built model [49]. Our investigation yielded a noteworthy achievement with a custom CNN model for classifying healthy leaves from those infected with red rot. The model demonstrated exceptional performance, achieving classification accuracy of 100% across various activation functions, including ReLU, Swish, and msswish. This performance exceeds the accuracy reported in the reference study [25].

5. Conclusion

This research investigates the development of a multiclass DNN Model for sugarcane leaf disease detection by integrating multiple TL-based binary CNN classifiers. Given the constraint of a limited publicly available dataset, the work employs a data augmentation strategy that incorporates images with natural backgrounds. This approach aims to enhance the model's robustness and generalizability to real-world scenarios by increasing the diversity of the training data and reducing the impact of dataset bias. This study investigates the performance of the msswish activation function in the final dense layer of various pre-trained CNN architectures, including DenseNet121, DenseNet169, DenseNet201, MobileNetV2, InceptionV3, InceptionResNetV2, Xception, and NASNetMobile. The results demonstrate that the incorporation of msswish outperforms the commonly used ReLU activation function and even the recently introduced swish function. Moreover, based on the p-values, msswish emerged as the activation function with the most robust statistical evidence of a difference in performance compared to the baseline. The proposed ensemble DNN model achieved a remarkable feat, attaining a macro-average validation accuracy of 98.12% in classifying five sugarcane leaf diseases. This accomplishment is noteworthy as it bypasses the typically time-consuming hyperparameter tuning process. Our model demonstrated notable training efficiency, with each epoch typically completing in two seconds within Kaggle's GPU 100 environment. The maximum training duration across all four CNNs was approximately 400 seconds. The model features a compact structure with fewer than 57 million parameters and approximately 225MB in size, making it suitable for deployment on resource-constrained platforms like mobile or edge computing devices. We will also carry out a comprehensive field image acquisition campaign to enrich the dataset. We anticipate that the enlarged dataset, covering a wider range of sugarcane diseases and environmental conditions, will improve the model's resilience and adaptability, making it easier to deploy in various real world agricultural settings.

6. Declarations

6.1. Author Contributions

Conceptualization: B.D., C.D., and C.S.R.; Methodology: C.D.; Software: B.D.; Validation: B.D., C.D., and C.S.R.; Formal Analysis: B.D., C.D., and C.S.R.; Investigation: B.D.; Resources: C.S.R.; Data Curation: C.S.R.; Writing Original Draft Preparation: B.D., C.D., and C.S.R.; Writing Review and Editing: C.S.R., B.D., and C.D.; Visualization: B.D.; All authors have read and agreed to the published version of the manuscript.

6.2. Data Availability Statement

The data presented in this study are publicly available and can be accessed at https://www.kaggle.com/, https://data.mendeley.com/datasets/9424skmnrk/1 and https://data.mendeley.com/datasets/355y629ynj/1.

6.3. Funding

The authors received no financial support for the research, authorship, and/or publication of this article.

6.4. Institutional Review Board Statement

Not applicable.

6.5. Informed Consent Statement

Not applicable.

6.6. Declaration of Competing Interest

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

References

[1] Tirkey, K. K. Singh, and S. Tripathi, "Performance Analysis of Ai based Solutions for Crop Disease Identification, Detection, and Classification," *Smart Agricultural Technology,* vol. 5, no. 1, pp. 100238-100249, 2023.

[2] Ramanjot, U. Mittal, A. Wadhawan, J. Singla, N. Jhanjhi, R. M. Ghoniem, S. K. Ray, and A. Abdelmaboud, "Plant Disease Detection and Classification: A Systematic Literature Review," *Sensors,* vol. 23, no. 10, p. 4769, 2023.

[3] S. E. Saqib, N. Lakapunrat, M. Yaseen, S. Visetnoi, K. Boonchit, and S. Ali, "Assessing the Impacts of Transitioning from Rice Farming to Sugarcane Cultivation on Food Security: Case of the Dietary Energy Intake in the Northeast of Thailand," *NJAS: Impact in Agricultural and Life Sciences,* vol. 95, no. 1, pp. 1-12, 2023.

[4] A. Ali, M. Khan, R. Sharif, M. Mujtaba, and S.-J. Gao, "Sugarcane Omics: An Update on The Current Status of Research and Crop Improvement," *Plants,* vol. 8, no. 9, p. 344, 2019.

[5] M. Dotaniya, S. Datta, D. Biswas, C. Dotaniya, B. Meena, S. Rajendiran, K. Regar, and M. Lata, "Use of sugarcane industrial by-products for improving sugarcane productivity and soil health," *International Journal of Recycling of Organic Waste in Agriculture,* vol. 5, no. 1, pp. 185–194, 2016.

[6] V. G. Krishnan, J. Deepa, P. V. Rao, V. Divya, and S. Kaviarasan, "An Automated Segmentation and Classification Model for Banana Leaf Disease Detection," *Journal of Applied Biology and Biotechnology,* vol. 10, no. 1, pp. 213–220, 2022.

[7] M. Sharma, C. J. Kumar, and A. Deka, "Early Diagnosis of Rice Plant Disease using Machine Learning Techniques," *Archives of Phytopathology and Plant Protection,* vol. 55, no. 3, pp. 259–283, 2022.

[8] E. Dhinesh, A. Jagan, "Detection of Leaf Disease Using Principal Component Analysis and Linear Support Vector Machine," *in 2019 11th International Conference on Advanced Computing (ICoAC),* vol. 11, no. 1, pp. 350– 355, IEEE, 2019.

[9] V. Pooja, R. Das, and V. Kanchana, "Identification of Plant Leaf Diseases Using Image Processing Techniques," *in 2017 IEEE Technological Innovations in ICT for Agriculture and Rural Development (TIAR),* vol. 2017, no. 1, pp. 130–133, IEEE, 2017.

[10] B. Liu, Y. Zhang, D. He, and Y. Li, "*Identification of Apple Leaf Diseases Based on Deep Convolutional Neural Networks*," *Symmetry,* vol. 10, no. 1, pp. 11-23, 2017.

[11] D. Hughes, M. Salath´e, "An Open Access Repository of Images on Plant Health to Enable the Development of Mobile Disease Diagnostics," arXiv preprint arXiv:1511.08060, 2015.

[12] W. B. Demilie, "Plant Disease Detection and Classification Techniques: A Comparative Study of The Performances," *Journal of Big Data,* vol. 11, no. 1, pp. 5-18, 2024.

[13] W. Shafik, A. Tufail, C. De Silva Liyanage, and R. A. A. H. M. Apong, "Using Transfer Learning-Based Plant Disease Classification and Detection for Sustainable Agriculture," *BMC Plant Biology,* vol. 24, no. 1, pp. 136-148, 2024.

[14] S. Sarkar and G. Kumar, "Identification of Sugarcane Leaf Diseases and Deficiency Disorders Using Transformers," *2024 First International Conference on Pioneering Developments in Computer Science and Digital Technologies (IC2SDT),* vol. 11, no. 1, pp. 493–498, Aug. 2024. doi:10.1109/ic2sdt62152.2024.10696467

[15] K. P. Ferentinos, “Deep Learning Models for Plant Disease Detection and Diagnosis,” *Computers and electronics in agriculture*, vol. 145, no. Feb., pp. 311–318, 2018.

[16] O. Attallah, “Tomato Leaf Disease Classification Via Compact Convolutional Neural Networks with Transfer Learning and Feature Selection,” *Horticulturae*, vol. 9, no. 2, pp. 149-161, 2023.

[17] P. K. Sethy, N. K. Barpanda, A. K. Rath, and S. K. Behera, “Deep Feature Based Rice Leaf Disease Identification Using Support Vector Machine,” *Computers and Electronics in Agriculture*, vol. 175, no. Aug., pp. 1-9, 2020.

[18] A. Krishnaswamy Rangarajan and R. Purushothaman, “Disease Classification in Eggplant Using Pre-Trained Vgg16 and MSVM,” *Scientific reports*, vol. 10, no. 1, pp. 2322-2334, 2020.

[19] S. Nigam, R. Jain, S. Marwaha, A. Arora, M. A. Haque, A. Dheeraj, and V. K. Singh, “Deep transfer learning model for disease identification in wheat crop,” *Ecological Informatics*, vol. 75, no. Jul., pp. 1-12, 2023.

[20] M. Altalak, M. A. Uddin, A. Alajmi, and A. Rizg, “A Hybrid Approach for The Detection and Classification of Tomato Leaf Diseases,” *Applied Sciences*, vol. 12, no. 16, pp. 8182-8194, 2022.

[21] H. Ulutas, and V. Aslanta, “Design of Efficient Methods for The Detection of Tomato Leaf Disease Utilizing Proposed Ensemble CNN Model,” *Electronics*, vol. 12, no. 4, pp. 827-837, 2023.

[22] M. S. H. Shovon, S. J. Mozumder, O. K. Pal, M. F. Mridha, N. Asai and J. Shin, "PlantDet: A Robust Multi-Model Ensemble Method Based on Deep Learning for Plant Disease Detection," *in IEEE Access*, vol. 11, no. Apr., pp. 34846-34859, 2023.

[23] P. Kanuparthi, V. Bejgam, and V. M. Viswanatham, “A Novel Approach of Ensembling the Transfer Learning Methods for Rice Plant Disease Detection and Classification.,” *Webology*, vol. 18, no. 2, pp. 439-448, 2021.

[24] S. A. Upadhye, M. R. Dhanvijay, and S. M. Patil, “Sugarcane Disease Detection Using CNN-Deep Learning Method,” *in 2022 Fourth International Conference on Emerging Research in Electronics, Computer Science and Technology (ICERECT)*, vol. 2022, no. Mar., pp. 1–8, IEEE, 2022.

[25] R. Sharma and V. Kukreja, “Segmentation and Multi-Layer Perceptron: An Intelligent Multi-Classification Model for Sugarcane Disease Detection,” *in 2022 International Conference on Decision Aid Sciences and Applications (DASA)*, vol. 2022, no. May, pp. 1265–1269, IEEE, 2022.

[26] A. A. Hernandez, J. L. Bombasi, and A. C. Lagman, “Classification of Sugarcane Leaf Disease Using Deep Learning Algorithms,” *in 2022 IEEE 13th Control and System Graduate Research Colloquium (ICSGRC)*, vol. 13, no. Aug., pp. 47–50, IEEE, 2022.

[27] S. D. Daphal and S. M. Koli, “Enhancing Sugarcane Disease Classification with Ensemble Deep Learning: A Comparative Study with Transfer Learning Techniques,” *Heliyon*, vol. 9, no. 8, pp. 1-19, 2023.

[28] C. Sun, X. Zhou, M. Zhang, and A. Qin, “Se-VISIONTRANSFORMER: Hybrid Network for Diagnosing Sugarcane Leaf Diseases Based on Attention Mechanism,” *Sensors*, vol. 23, no. 20, pp. 1-22, 2023.

[29] U. Vignesh and B. S. Chokkalingam, “Enc-SVMWEL: Ensemble Approach Using CNN and SVM Weighted Average Ensemble Learning for Sugarcane Leaf Disease Detection,” *in 2023 International Conference on Sustainable Computing and Data Communication Systems (ICSCDS)*, vol. 2023, no. Apr., pp. 1663–1668, IEEE, 2023.

[30] S. Garg, V. Kukreja, R. Sharma, V. Sharma, and M. Aeri, “Unveiling the Hidden Patterns: Deep Learning Approach for Early Detection and Severity Classification Of Sugarcane Brown Spot,” *in 2023 14th International Conference on Computing Communication and Networking Technologies (ICCCNT)*, vol. 14, no. Nov., pp. 1–5, IEEE, 2023.

[31] N. Dhawan, V. Kukreja, R. Sharma, S. Vats, and A. Verma, “Deep Learning Based Sugarcane Downy Mildew Disease Detection Using CNNLSTM Ensemble Model for Severity Level Classification,” *in 2023 14th InternationalConference on Computing Communication and Networking Technologies (ICCCNT)*, vol. 14, no. Nov., pp. 1–5, IEEE, 2023.

[32] D. Banerjee, V. Kukreja, S. Hariharan, V. Jain, and S. Dutta, “An Intelligent Framework for Grassy Shoot Disease Severity Detection and Classification In Sugarcane Crop,” *in 2023 2nd International Conference on Applied Artificial Intelligence and Computing (ICAAIC)*, vol. 2, no. Jun., pp. 849–854, IEEE, 2023.

[33] V. Tanwar, S. Lamba, and B. Sharma, “Deep Learning-Based Hybrid Model for Severity Prediction of Leaf Smut Sugarcane Infection,” *in 2023 Third International Conference on Artificial Intelligence and Smart Energy (ICAIS)*, vol. 3, no. Mar., pp. 1004–1009, IEEE, 2023.

[34] J.-S. Lee and I.-S. Oh, “Binary Classification Trees for Multi-Class Classification Problems.,” *in ICDAR*, vol. 3, no. Sep., pp. 1-5, Citeseer, 2003.

[35] S. Daphal and S. Koli, “Sugarcane Leaf Disease Dataset,” *Mendeley Data,* vol. 1, 2022, doi: 10.17632/9424skmnrk. [Accessed: Aug. 5, 2024].

[36] S. Thite, Y. Suryawanshi, K. Patil, and P. Chumchu, “Sugarcane Leaf Dataset: A Dataset for Disease Detection and Classification for Machine Learning Applications,” *Data in Brief,* vol. 2024, no. Feb., pp. 1-9, 2024.

[37] A. Hussain Khan, “Red rot sugarcane disease leaf dataset,” Kaggle, 2022. [Online]. Available: https://www.kaggle.com/datasets/alihussainkhan24/red-rot-sugarcanedisease-leaf-dataset. [Accessed: Aug. 5, 2024].

[38] S. Shubham, “Five crop diseases dataset,” Kaggle, 2022. [Online]. Available: https://www.kaggle.com/datasets/shubham2703/five-crop-diseasesdataset. [Accessed: Aug. 5, 2024].

[39] R. A. Shitab, “Sugarcane leaf disease dataset,” Kaggle, 2022. [Online]. Available: https://www.kaggle.com/datasets/roshitab/sugarcane-leaf-diseasedataset. [Accessed: Aug. 5, 2024].

[40] P. Soundar, “Sugarcane disease dataset,” Kaggle, 2022. [Online]. Available: https://www.kaggle.com/datasets/prabhakaransoundar/sugarcanedisease-dataset. [Accessed: Aug. 5, 2024].

[41] Q. H. Nguyen, H.-B. Ly, L. S. Ho, N. Al-Ansari, H. V. Le, V. Q. Tran, I. Prakash, and B. T. Pham, “Influence of Data Splitting on Performance of Machine Learning Models in Prediction of Shear Strength of Soil,” *Mathematical Problems in Engineering,* vol. 2021, no. Feb., pp. 1–15, 2021.

[42] J. Seidu, A. Ewusi, J. S. Y. Kuma, Y. Y. Ziggah, and H.-J. Voigt, “Impact of Data Partitioning in Groundwater Level Prediction Using Artificial Neural Network for Multiple Wells,” *International Journal of River Basin Management,* vol. 21, no. 4, pp. 639–650, 2023.

[43] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, “Densely Connected Convolutional Networks,” *In Proceedings of The IEEE Conference on Computer Vision and Pattern Recognition,* vol. 88, no. Nov., pp. 4700–4708, 2017.

[44] A. Ramcharan, K. Baranowski, P. McCloskey, B. Ahmed, J. Legg, and D. P. Hughes, “Deep Learning for Image-Based Cassava Disease Detection,” *Frontiers in plant science,* vol. 8, no. 1, pp. 1-7, 2017.

[45] S. Sunkari, A. Sangam, M. Suchetha, R. Raman, R. Rajalakshmi, S. Tamilselvi, “A Refined Resnet18 Architecture with Swish Activation Function for Diabetic Retinopathy Classification,” *Biomedical Signal Processing and Control,* vol. 88, no. Feb., pp. 1-8, 2024.

[46] R. Bhuvanya, T. Kujani, R. Padmavathy, P. Matheswaran, and P. Punitha, “Beyond Relu: Unlocking Superior Plant Disease Recognition with Swish,” *International Journal of Innovation and Technology Management,* vol 2024, no. 1, pp. 1-12, 2024.

[47] M. K. A. Mazumder, M. Mridha, S. Alfarhood, M. Safran, M. Abdullah-Al-Jubair, and D. Che, “A Robust and Light-Weight Transfer Learning-Based Architecture for Accurate Detection of Leaf Diseases Across Multiple Plants Using Less Amount of Images,” *Frontiers in Plant Science,* vol. 14, no. Jan., pp. 1-22, 2024.

[48] C. Bajpai, R. Sahu, and K.J. Naik, "Deep Learning Model for Plant-Leaf Disease Detection in Precision Agriculture," *International Journal of Intelligent Systems Technologies and Applications,* vol. 21, no. 1, pp. 72-91, 2023.

[49] A. K. Bagchi, M. A. Haider Chowdhury and S. A. Fattah, "Sugarcane Disease Classification using Advanced Deep Learning Algorithms: A Comparative Study," *2024 6th International Conference on Electrical Engineering and Information & Communication Technology (ICEEICT), Dhaka, Bangladesh,* vol. 6, no. May, pp. 887-892, 2024.