Journal of Intelligent Systems and Internet of Things
Vol. 16, No. 01, PP. 61-74, 2025
DOI: https://doi.org/10.54216/JISIoT.160106

An Enhancement of YOLOV3-Tiny Model for Turmeric Plant Disease Detection

Shylaja Santhosh[1,*], Revathi Thiyagarajan[2]
[1]Research Scholar, Department of Computer Science,PSG College of Arts & Science,
Coimbatore, Tamilnadu, India
[2]Associate professor & Dean, Department of Computer Science,PSG College of Arts & Science,
Coimbatore, Tamilnadu, India
Emails: shylusantophd@gmail.com; revathi@psgcas.ac.in

Abstract

Turmeric is a rhizomatous crop recognized for its medicinal effects which requires significant observation to ensure appropriate growth and progression. Turmeric plant diseases cause yield losses impacting food production systems and causing economic losses. Early prevention of these diseases is crucial for improving agricultural productivity. For this reason, The Improved YOLOV3-Tiny Model (IY3TM) was developed using Cycle-GAN and Convolutional Neural Network (CNN) with residual network for the early turmeric plant disease detection. However, this model leads to the omission of vital details along with the exact positioning of key attributes, thereby decreasing prediction accuracy. To resolve this, Convolutional and Vision Transformer model for Turmeric Diseases Detection (ConViT-TDD) is proposed for the prediction of turmeric plant diseases. ConViT-TDD is integrated into IY3TM with a self-attention mechanism and CNN-based global perspective to enhance the performance of the model A ConViT-TDD block involves the input channel transformation, the channel as well as spatial attention mechanism and global-minded transformers. The input channel transformation utilizes a convolutional layer to minimize the dimension of input channel and reduces the computational complexity. Global-minded transformers generate a feature vector based on the input channel transformation that is then transmitted to the encoder component. By collecting channel weights and spatial weights, respectively, the channel and spatial attention modules enhance the model's sensitivity to certain channel attributes and spatial locations, hence altering the feature representation of those channels and spatial locations. The attention module can adaptively change the weights of channel and spatial features for improved feature extraction and fusion. Once the initial attributes are reformed, the IY3TM detects and classifies the turmeric plant diseases. The test outcomes reveal that the ConViT-TDD model accomplishes an overall accuracy of 93.16% on the collected turmeric plant diseases images which is contrasted with the classical CNN models.

Received: November 17, 2024 Revised: January 07, 2025 Accepted: February 13, 2025

Keywords: Turmeric plant disease; Convolutional Neural Network; Vision Transformer; Spatial Attention Module; Channel Attention Module

1. Introduction

Turmeric is commonly known as curcuma longa, is a famous spice crop from the Zingiberaceae family [1]. It is a perennial herbaceous plant with oblong, sharply pointed leaves and funnel-shaped yellow blossoms. The plant can grow up to 1-meter-long and 3 feet tall. Its primary extracts include rhizome, curcumin, oleoresins, volatiles, and essential oil, used in both sustenance and non-sustenance sectors worldwide [2, 3]. Turmeric cultivation is prevalent in numerous nations, but its yield is primarily impacted by plant diseases. Turmeric production is primarily impacted by diseases such as leaf spot, leaf blotch, and rhizome rot [4]. Among these, rhizome rot is the most detrimental, resulting in significant economic losses and a decline in turmeric availability and distribution [5]. Early detection of turmeric plant diseases is crucial for improving food productivity, managing farm diseases, and enabling farmers to make informed decisions about disease types.

Traditional plant disease detection requires naked eye inspection by professionals that necessitates a large team and ongoing plant monitoring systems [6]. This manual detection process is slow, requires considerable effort, and depends largely on experts, making it impractical for precision farming. Additionally, these methods are susceptible to human bias and fatigue, leading to decreased accuracy. To overcome this issues, image processing models have been developed to evaluate the plant diseases using plant disease images [7]. This procedure involves capturing images, data preparation, segmentation, extraction of features, and categorization [8]. Pathologists can forecast plant diseases by analysing color, texture, spot, and size changes from captured images. This framework aids in high-precision plant analysis for repair and future disease prediction, benefiting farmers in various agricultural activities [9]. Nevertheless, drawbacks include reduced precision with larger datasets and the reliance on manual feature extraction, which is labour-intensive and prone to subjective bias based on varying prioritization criteria.

AI techniques such as Machine Learning (ML) and Deep Learning (DL) are progressively being employed to autonomously identify plant diseases, offering a promising solution for streamlined agricultural management [10]. ML methods like Support Vector Machines (SVM), K-Nearest Neighbors (KNN), Random Forest (RF), and Artificial Neural Networks (ANN) are commonly applied for plant disease prediction through image analysis [11]. However, extracting useful data from large images is challenging and ML approaches result in computational complexity and require distinct feature selection before training. In comparison to ML frameworks, DL models offer superior performance in predicting and classifying plant diseases [12]. These models can extract features without human intervention, providing valuable information for farmers, reducing crop treatment costs and increasing productivity [13]. DL algorithms like Convolutional Neural Network (CNN), Recurrent Neural Network (RNN) and Deep Belief Network (DBN) assists the pathologists detect plant diseases based on unique features from images, enhancing crop productivity, disease management, and recognition accuracy [14].

Various DL models have been implemented to predict and classify the turmeric plant diseases. Foe instances, an Improved YOLOV3-Tiny model (IY3TM) was developed [15] for the early detection of turmeric plant diseases. In this method, Cycle-GAN (CG) model was used to augment the collected data to improve the recognition accuracy. This model utilized CNN with residual network structure to extract the features form the augmented images. Finally, IY3TM model was utilized for the turmeric plant diseases detection. In order to prevent the dimensionality, The IY3TM models employ pooling layers, leading to the reduction of essential details, such as the exact positioning of key features.

Hence, in this paper, ConViT-TDD is designed to detect and classify the Turmeric plant diseases in turmeric plants. The ConViT-TDD model utilizes CNN and ViT structures merged into IY3TM, improving performance using the CNN's global vision and self-attention mechanism. The proposed model constitutes comprises fully linked layers, bottleneck, ConViT-TDD block adaptive pooling, and regular convolution. Remaining connections are used by the IY3TM structure to improve the network's training efficiency and solve the gradient-vanishing problem. ConViT-TDD composed of four stages, which is illustrated below.

- Initially, the input channel transformation lowers the size of the input channel by utilizing the 1×1 convolutional layer. This phase can minimise the amount of parameters and computational cost while maintaining consistent feature dimensionality throughout the transformer encoder input process. Second, the transformers with global views. Consider the feature map after input channel transformation $(Height\ (h), Weight\ (w), Dimension\ (d))$ can be unfolded to produce a word vector of (h, w, d) on the h and w axis surface. Which is then added with positional encoding and transmitted to the encoder part.
- The channel and spatial attention modules enhance the model's learning of channel weights to focus on various channel attributes and spatial positions and spatial weights respectively, thereby adjusting the feature representation of different channels and spatial positions.
- Finally, Convolutional Block Attention Module (CBAM) is used to improve feature integration and extraction, change the weights of the channel and spatial features.

Once the initial features are reconstructed, IY3TM classifier assist to detect and classify the turmeric plant diseases. This model minimizes the depletion of crucial data and identifies the precise position of significant features for predicting rhizome rot in turmeric plants.

The structure of this manuscript is organized as follows: Section II reviews various studies related to the identification and classification models of turmeric plant diseases. Section III provides an explanation of the ConViT-TDD model, while Section IV demonstrates its effectiveness. Section V summarizes the study and explores potential future improvements.

2. Related Work

Rajasekaran et al. [16] developed an artificial intelligence (AI) model for the identification and classification of diseases affecting turmeric plants. VGG-16 was used in this method for the detection and classification of miserable images. The model was trained to distinguish between infected and non-infected leaf images with an accuracy level of early detection using the hidden layer image processing layers. But due to inadequate optimisation, the VGG-16 parameters had a significant misclassification rate.

Revathy et al. [17] presented an efficient approach for turmeric growth and plant disease prediction using Multilevel Linear Regression (MLR) method. The collected dataset was pre-processed and normalized to eliminate the noises and irrelevant data form the raw dataset. Then, MLR was employed to predict the turmeric growth level based on the growth level time period was calculated to identify the diseases on turmeric plant. But, this model was not suitable to deal with large datasets

Patil & Patil [18] developed an automated system for identifying turmeric rhizomes. The system uses images from fields or agriculture research centers, pre-processed using a median filter, and segmented using hue channel images, Otsu thresholding, and a canny operator. The rhizome images' color and texture are extracted using a GLCM model and Linear Discriminant Analysis (LDA) and SVM classifier are used to predict and categorize the types of turmeric rhizomes. SVM provides higher result than LDA in prediction task. But, maximum training time and results with high localization errors.

Bouni et al. [19] suggested an impact of pre-trained CNN models for tomato disease prediction. Initially, the plant diseases dataset was collected, pre-processed and feature extraction models like Principal Component Analysis (PCA) filters were used to extract the features form the pre-processed data. The DenseNet model was performed to predict the tomato diseases prediction and classification. But, the model suffers from overfitting due to the significant loss of valuable data.

Islam et al. [20] created a web-based DeepCrop model for crop disease prediction. This model utilizes pre-processed raw images from a dataset which have been rotated, resized and sheared before being integrated into the learning module. The model involves training Transfer Learning (TL) models with a training image dataset and validating the model with reserved test images for performance evaluation. Finally, 1D-CNN model was performed to effectively extract and detect crop diseases. But, this model was trained on a limited dataset, focusing solely on viral diseases.

Ullah et al. [21] constructed a DeepPlantNet for plant leaf diseases detection and classification. The plant diseases dataset was collected and pre-processed to remove the irrelevant information from the images. Then, the DeepPlantNet was employed to automatically extract the most discriminative features for accurate plant disease detection and classification. However, the training time of this model was high.

Gong [22] suggested an apple leaf disease recognition and prediction using Faster-CNN and YOLOv3 model. The dataset was pre-processed to enhance color channels, normalize pixel values, and reduce noise to enhance image quality. Then, Faster-CNN was utilized to extract high-quality regions and retrieve Region of Interests (ROIs) for similar shape features, enhancing performance accuracy. Finally, YOLOv3 was utilized for the feature extraction and fully connected layers of YOLOv3 was used for final classification. But, this model results with high memory consumption and training time.

Khan et al. [23] devised a YOLO models for maize plant leaf disease classification. The collected dataset was pre-processed using resizing model to remove noises. Then, data augmentation was applied for flipping, rotating, scaling and cropping the pre-processed images. Finally, YOLO models (YOLOv3-Tiny to YOLOv8) was employed for the feature extraction and classification of maize plant disease. However, this model results in lower accuracy due to the high requirement of training data.

Kini et al. [24] developed a ConvNets for early stage black pepper leaf disease prediction based on TL. The distinct plant diseases images were collected and pre-processed using Gaussian Mixture Model (GMM). Region based technique was employed for segmenting the pre-processed images. Atlast, the pre-trained CNN models were employed for plant diseases prediction and categorization. But, the learning rate and optimizer significantly impact the high variance of detection types.

Sutiaji et al. [25] devised a Weighted Ensemble CNN (WECNN) models to detect plant leaf diseases. In this model, two or three pre-trained CNN model were integrated and performed TL on individual CNN models to prioritize the high-dimensional features and using grid search to find each model's optimum weights. Finally, the softmax was employed for the plant disease classification. But, this model has suffered with overfitting issues and was not suitable for large-scale datasets.

3. Proposed Methodology

In this part, the proposed ConViT-TDD approach is developed for the prediction of rhizome rot (turmeric) plant diseases. In this module, the collected plant diseases images will be processed and normalized. Then, the ConViT-TDD is performed to preserve the information loss and accurately locates key features for predicting disease in turmeric plants. Finally, the YOLOV3 model from IY3TM [15] model is employed for turmeric plant diseases classification. Figure 1 illustrates the workflow of the entire proposed model. Table 1 presents the notation table for the proposed approach.

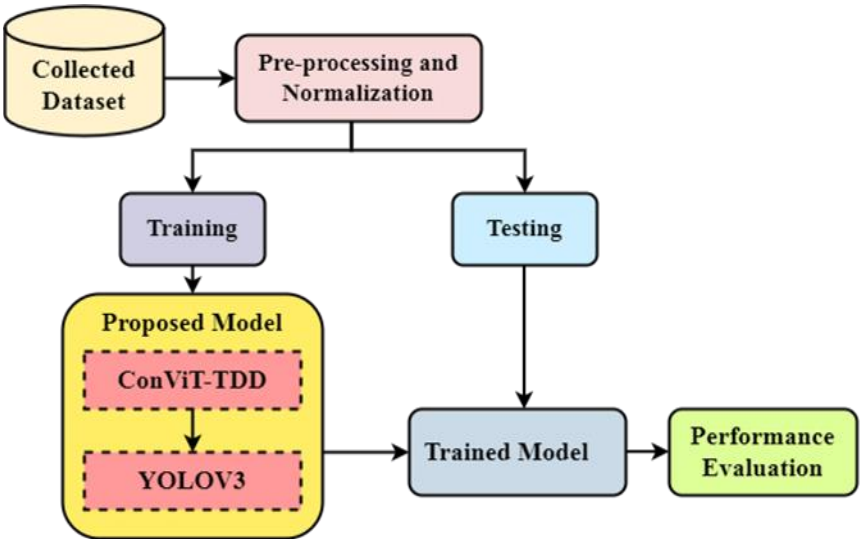


Figure 1. Overall workflow of the proposed model

Table 1: Notation List

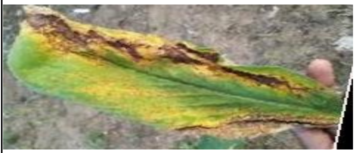
Notations	Description
(h, w, d)	Feature Map Size
h	Height
w	Width
d	Dimension
c	Channel
$\mathcal{PE}$	Positional Encoding
os	Position in the Input Series
	Index Size in the $\mathcal{PE}$ vector
x_{model}	Embedding size in Transformer
S_{model}	Each Position of Dimensional $\mathcal{PE}$ vector

U	Query Vectors
V	Key Vectors
Y	Value Vector
H	Number of Heads
NT	Integration of Heads
$\mathcal{W}^R$	Resulting Weight Matrix
$head_x$	Each Head Computation
$\mathcal{W}_x^U$	Weights Matrices to Query Vectors
$\mathcal{W}_x^V$	Weights Matrices to Key Vectors
$\mathcal{W}_x^Y$	Weights Matrices to Value Vectors
Att	Attention Operation
s_v	Dimensionality of Key Vectors and Query

A. Dataset Description

For the experimental purposes, random images of turmeric leaves are gathered from various internet sources. From this, we have gathered 800 turmeric plant images that is categorized into 4 classes like healthy, leaf blotch, leaf spot and rhizome rot. Each category comprises 200 images per class. Table 1 provides the gathered sample turmeric plant images of 4 classes.

Table 2: Gathered sample images of turmeric plants

Classes	Sample Images	
Healthy		
Leaf Blotch		
Leaf Spot		
Rhizome Rot		

B. Data pre-processing and Normalization

In this method, pre-processing involves adjusting the image contrast, illumination and color all these modifications will enhance the collected image details. Then, Gaussian filtering is employed to eliminate noise and smoothen the images. These approaches increased image quality by increasing feature extraction and model training efficiency. Normalization techniques on the original dataset assures the model's consistency and convergence throughout training task. The pixel values are adjusted to a specified range and the actual images are downsized to meet the specifications of DL model. Data augmentation, such as random movement, interpreting, scaling and flipping are performed to boost visual variety and richness. By doing this, the training set is increased and the likelihood of overfitting is decreased. Normalisation and data augmentation improve the capability of the model to analyse and identify photos of turmeric plant disease.

C. Proposed ConViT-TDD model

In this method, ConViT-TDD Model is developed using the structure of DCNN and ViT by introducing a self-attention mechanism and global view (regarding CNN) to enhance the model's efficiency. The figure 2 depicts the ConViT-TDD model that constitutes of typical convolution, bottleneck, ConViT-TDD block, adaptive pooling block and fully connected layers.

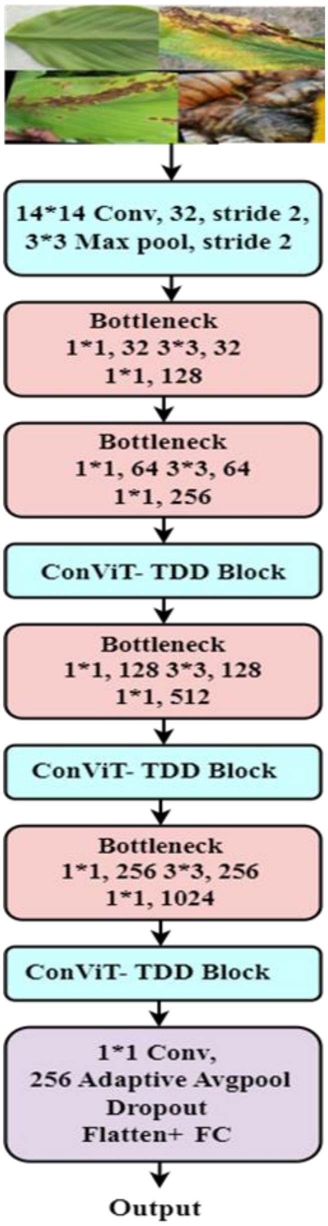


Figure 2. Structure of ConViT-TDD Model

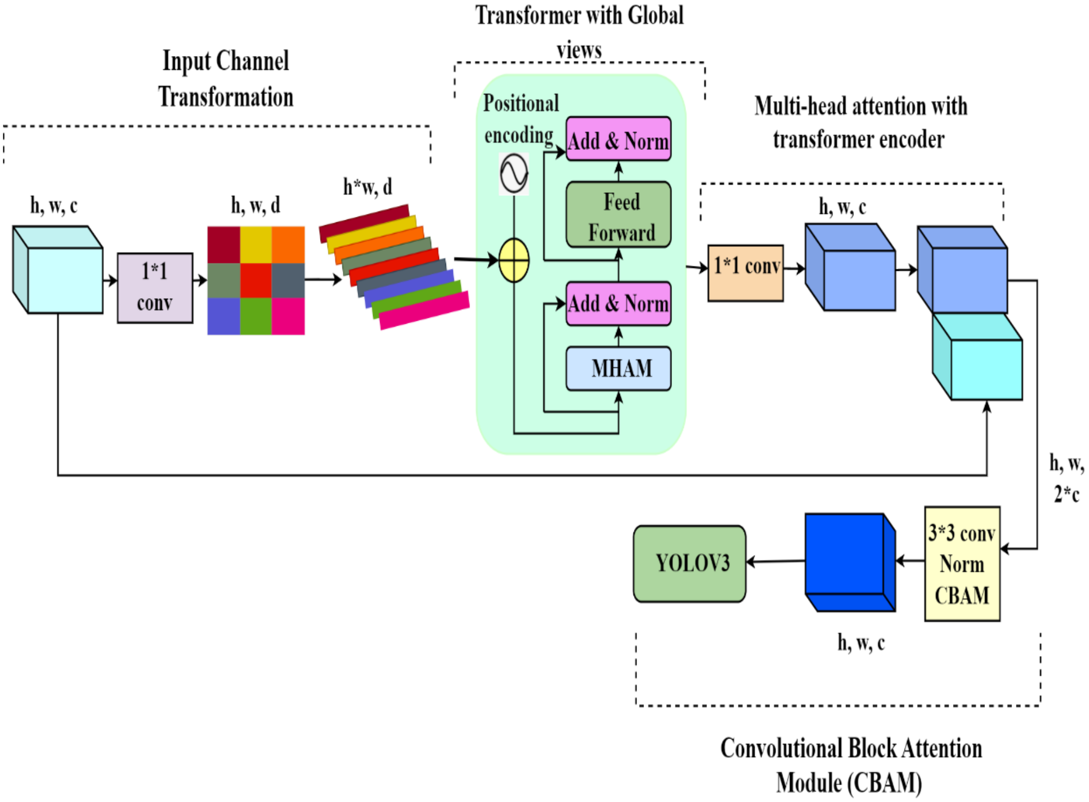


Figure 3. ConViT-TDD Block Design

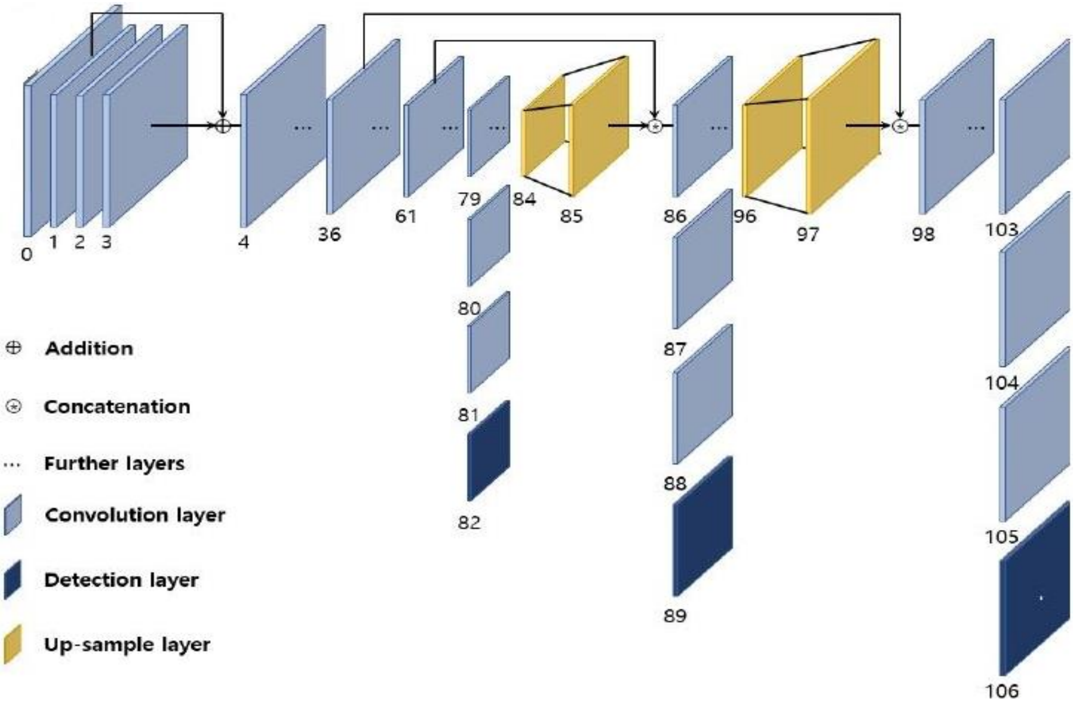


Figure 4. Structure of YOLOV3 for classification

Residual connections are used in this model assist to improve the network's training efficiency and solve the gradient-vanishing problem. The three convolutional layers in this ConViT-TDD model incluses 1 × 1 and 3 × 3 convolutional layer, and the 1 × 1 convolutional layer are mostly responsible for the bottleneck. These convolutional layers yield channels that are, in turn, $^1/_4$ of the real input channel, $^1/_4$ of the real input channel, and four times the amount of the general input. The bottleneck module will be changed to minimize computational complexity, boost network depth, and improve its feature extraction capability. The bottleneck module presented a structure that uses a high-dimensional expansion layer to return the input features to their original dimension after mapping them to a low-dimensional space through a bottleneck layer. Better detection performances can be achieved by decreasing the parameter count and computational demand by utilizing more parameters and increasing computational load.

The figure 3 illustrates the structure of ConViT-TDD block in ConViT-TDD model. Figure 4 depicts the layout of YOLOV3 model. The ConViT-TDD block is divided into four different stages. Initially, 1×1 convolutional layer is used in the input channel transformation to reduce the input channel's dimension. When feeding data into the transformer encoder, this stage can reduce computational complexity and parameter count while preserving constant feature dimensionality.

In the meantime, global view transformers are taken into account. Considering the feature map following input channel transformation (h, w, d), a word vector of (h, w, d) can be obtained by unfolding the feature map on the h and w axis surface. After adding positional encoding, the word vector is encoded and sent to the encoder component. The equations (1) and (2) show the positional encoding $(\mathcal{PE})$.

$$\mathcal{PE}\ (ps, 2x) = sin(\frac{ps}{10000^{2x/S_{model}}}) \qquad (1)$$

$$\mathcal{PE}\ (ps, 2x+1) = cos\left(\frac{ps}{10000^{2x/S_{model}}}\right) \qquad (2)$$

where, ps is the position in the input series, x defines the size index in the $\mathcal{PE}$ vector and x_{model} indicates the embedding size in the transformer model. Each point of $\mathcal{PE}$ vector dimension is represented S_{model}. The above Eq. (1) and Eq. (2) utilizes sine and cosine functions with varying periods in different ranges, providing unique encoding for different positions. The transformer model captures the positional data by integrating the PE vector to the word-embedding vector. The illustration below shows the Multi-Head Attention Mechanism (MHAM) in a transformer encoder. in Eq. (3).

$$MHAM_{(U,V,Y)} = INT(J_1, \ldots.., J_H)\mathcal{W}^R \qquad (3)$$

Where, U, V and Y signifies the query, key and value vectors accordingly. H is the number of heads, INT indicates the combination of the heads together, and the resulting weight matrix is denoted by W^R. Each head's computation is given. in Eq. (4)

$$head_x = Att(U\mathcal{W}_x^U, V\mathcal{W}_x^V, Y\mathcal{W}_x^Y) \qquad (4)$$

Where, $\mathcal{W}_x^U, \mathcal{W}_x^V, \mathcal{W}_x^Y$ are the weights matrices utilized to carry out the necessary linear modifications to the query, value, and key vectors. The attention operation used in the computation is represented by the symbol Att in Eq. (5).

$$Att\ (U, V, Y) = softmax\left(\frac{UV^T}{\sqrt{s_v}}\right)Y \qquad (5)$$

In Eq. (5), s_v regarded as the dimensionality of the key vectors and the query.

Depending on the processing method used for the original input, the output can be transformed back to its original feature map size (h, w, d) after it has been obtained. $1 * 1$ convolution is used in the integration and resultant stages to restore the feature channel number to its initial size. Residual concatenation with the input feature map comes next, Prior to batch normalisation using the Convolutional Block Attention Module (CBAM) attention strategy for global feature fusion and extraction, the $3 * 3$ convolutional kernel is employed for feature integration and dimension removal. By learning channel weights and spatial weights, respectively, the channel and spatial attention modules enhance the model's attention to distinct channel characteristics and spatial positions, consequently modifying the feature representation of those channels and spatial positions. To improve feature integration and extraction, the CBAM attention module can change the weights of the channel and spatial characteristics. Table 2 depicts the definite values for h, w, channel (c) and d.

Table 3: Pre-decided Specifications of the Transformer encoder

Consecutive Number of ConViT-TDD Block	1	2	3
h	56	28	14
w	56	28	14
c	256	512	1024
d	32	64	128
Number of layers	64	128	256
Feed Forward Layer Size	3	5	4

Table 4: Hyper-parameters Configuration

Parameters	Output Dimension
ConViT-TDD model	
Input Image	$224 * 224 * 3$
Convolutional Layer 1	$224 * 224 * 32$
Batch Norm	$224 * 224 * 32$
ReLU + Maxpool	$224 * 56 * 32$
Bottleneck 1-1	$112 * 112 * 128$
Bottleneck 1-2	$112 * 112 * 128$
Bottleneck 1-2	$112 * 112 * 128$
Bottleneck 2	$56 * 56 * 256$
ConViT-TDD Block 1	$56 * 56 * 256$
Bottleneck 3	$28 * 28 * 512$
ConViT-TDD Block 2	$28 * 28 * 512$
Bottleneck 4	$14 * 14 * 1024$
Conv 2	$14 * 14 * 256$
AdaptiveAvgPool + Dropout	$1 * 1 * 256$
Linear variations	2
Learning rate	0.0001
Momentum	0.7

Batch size	64
Optimizer	Adam
Loss Function	Mean Square Error
YOLOV3	
Momentum	0.9
Batch size	8
Initial rate	0.001
Decay	0.0005
Loss Function	Cross Entropy

The table 4 displays the number of parameters in each ConViT-TDD model layer, while also tracking the development of picture feature maps at every level. Hence, by employing the ConViT-TDD model, the vital information with accurate position of the most prominent features the will be preserved efficiently. The extracted features will be fed into YOLOV3 model for the prediction and classification of turmeric plant diseases. The below algorithm provides the complete ConViT-TDD model turmeric plant diseases identification.

Algorithm: ConViT-TDD model for turmeric plant diseases prediction

Input: Collected dataset

Output: Detection and classification of disease and non-diseases turmeric plants

Step 1: The plant diseases images will be pre-processed, normalized and augmented to increase the diversity of image quality

Step 2: Then, the pre-processed images will be divided into training and testing stages.

Step 3: ConViT-TDD model utilizes DCNN and ViT models to derive the attributes from the pre-processed images.

Step 4: ConViT-TDD block comprises to four stages to alleviate the complexity in network training.

Step 4.1: The input channel transformation is performed to decrease the feature and parameter dimension issues.

Step 4.2: Transformers are combined with global CNN views with positional encoding and encoder element to enhance information loss.

Step 4.3: The channel and spatial attention modules enhance the model's attention adjusting the feature representation of different channels and spatial positions.

Step 4.4: CBAM optimizes the integration and extraction of features by adjusting the weights of channel and spatial features.

Step 5: The extracted features from ConViT-TDD module will be fed into YOLOV3 training or diseases classification.

Step 6: YOLOV3 is trained from training images using the Adam optimizer to get the trained model

Step 7: The trained model of YOLOV3 is used to classify/estimate the desired labels

Step 8: The classified label of test images is validated by verifying actual label of test images.

Step 9: The prediction of turmeric plant diseases is achieved using corresponding labels.

4. Result and Discussion

This section evaluates the performance of the ConViT-TDD model by implementing it in MATLAB 2019a with the gathered images (outlined in Section 3.1). The data is divided into 80% for training and 20% for testing. Additionally, a comparative analysis is performed to assess the enhancement of the ConViT-TDD model against existing models using the evaluation metrics provided below.

Accuracy: It is the fraction of proper partition and categorization of turmeric diseased plant samples (plant images) over the total samples tested.

$$Accuracy = \frac{True\ Positive\ (TP) + True\ Negative\ (TN)}{TP + TN + False\ Positive\ (FP) + False\ Negative\ (FN)} \qquad (11)$$

In Eq. (11), The number of healthy plant samples that were accurately classified into this category is referred to as TP, while the number of diseased plant samples that were correctly classified as appropriate diseased classes is referred as TN. Similarly, FP indicates the number of samples of diseased plants. categorized as healthy and FN indicates the number of healthy plant samples categorized as diseased classes.

Precision: It calculates the proportion of TP predictions (accurately identified turmeric plant disease samples) among all positive predictions made by the model.

$$Precision = \frac{TP}{TP + FP} \qquad (12)$$

Recall: It assesses the proportion of TP predictions relative to all actual positive cases in the dataset.

$$Recall = \frac{TP}{TP + FN} \qquad (13)$$

F1-score: It is the combined average of precision and recall.

$$F1 - score = \frac{2 \times Precision \times Recall}{Precision + Recall} \qquad (14)$$

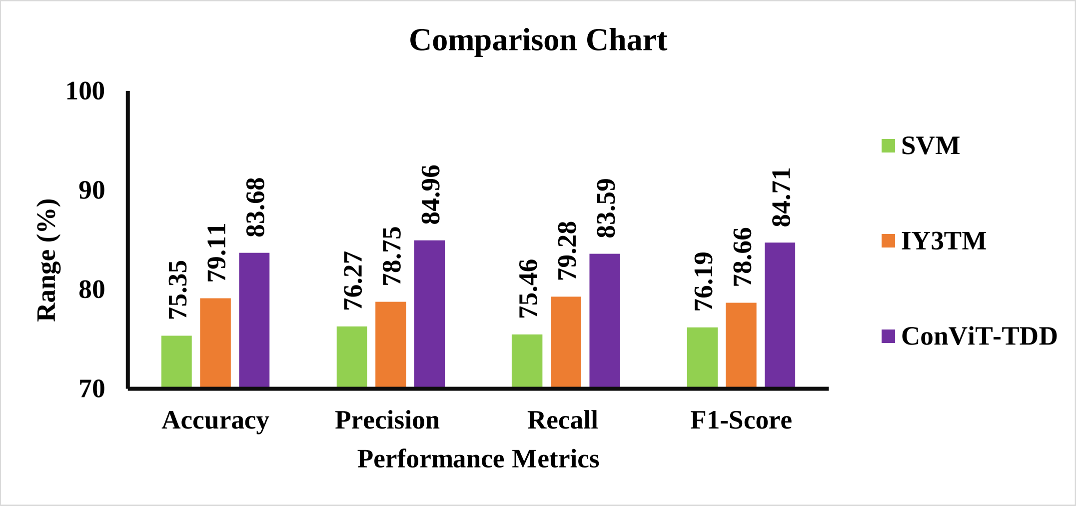


Figure 5. Evaluation of Existing and suggested models for Turmeric Rhizome Rot Diseases

In figure 5, the comparison of existing models like SVM [18] and IY3TM [15] and proposed ConViT-TDD models for turmeric rhizome rot diseases prediction. In this method, ConViT-TDD is 11.06%, 11.39%, 10.77%, 11.18% greater than the SVM and 5.78%, 7.89%, 5.44%, 7.69% greater than the IY3TM in terms of accuracy, precision, recall and f1-score. This analysis proves that the proposed ConViT-TDD efficiently provides greater results than other models for turmeric rhizome rot diseases prediction.

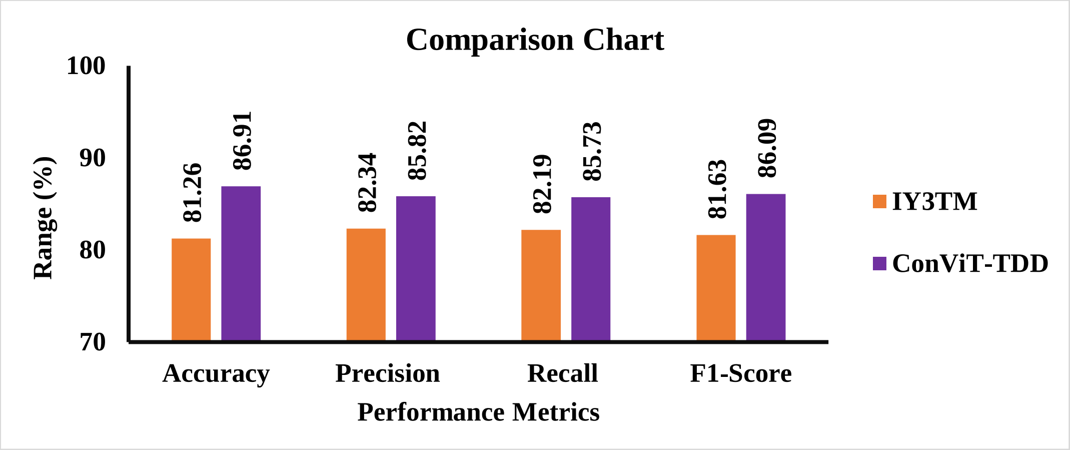


Figure 6. Evaluation of suggested and existing models for Turmeric Leaf Spot Diseases

In figure 6, the comparison of existing models like and IY3TM [15] and proposed ConViT-TDD models for turmeric leaf spot diseases detection. In this method, ConViT-TDD is 6.95%, 4.22%, 4.31% and 5.46% higher than the IY3TM in terms of accuracy, precision, recall and f1-score. This analysis proves that the proposed ConViT-TDD efficiently provides greater results than other models for turmeric leaf spot diseases prediction.

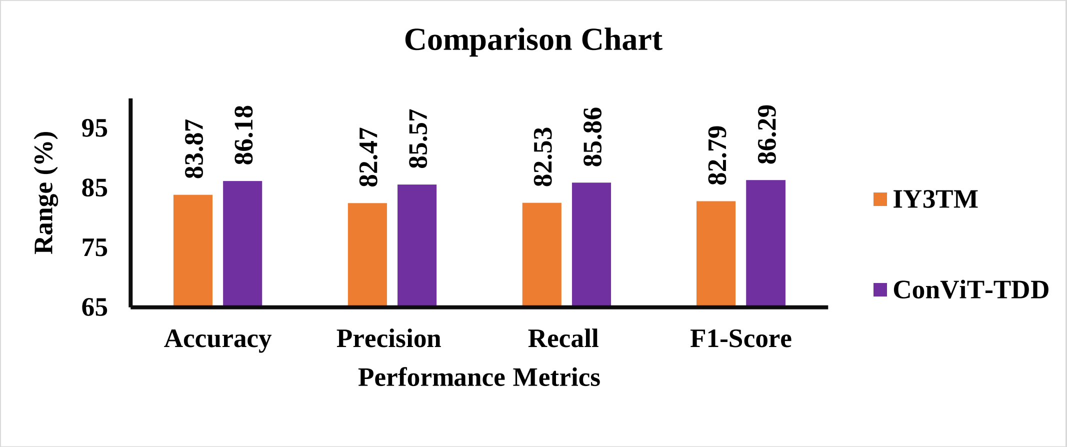


Figure 7. Evaluation of suggested and existing Models for Turmeric Leaf Blotch Diseases

The evaluation between the existing models like and IY3TM [15] and proposed ConViT-TDD models for turmeric leaf blotch diseases detection is depicted in figure 7. In this method, ConViT-TDD is 2.75%, 3.76%, 4.03%, and 4.23% higher than the IY3TM in terms of accuracy, precision, recall and f1-score. This analysis proves that the proposed ConViT-TDD efficiently provides greater results than other models for turmeric leaf blotch diseases prediction.

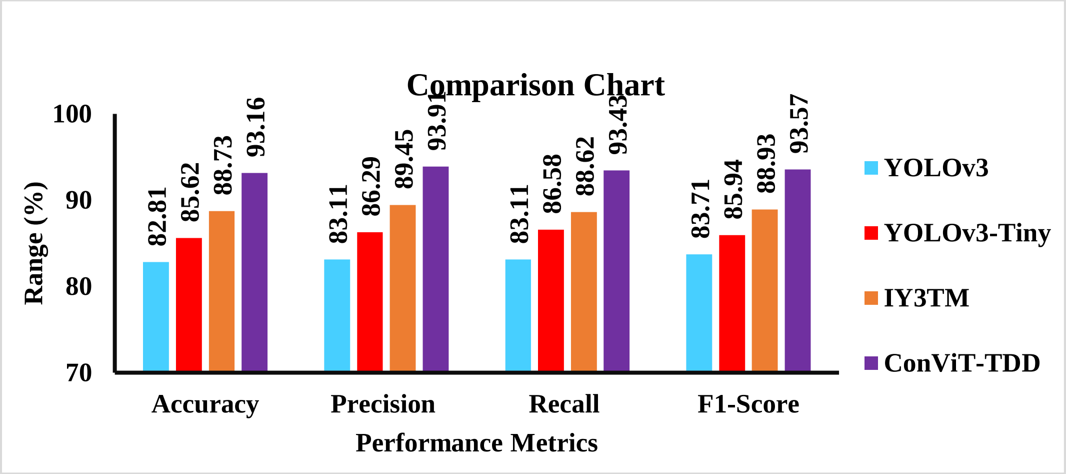


Figure 8. Evaluation of suggested and existing models on Collected Turmeric Plant Image Dataset

Figure 8 portrays the performance of ConViT-TDD model against different classification models like YOLOv3 [22], YOLOv3-Tiny [23], IY3TM [15] and proposed ConViT-TDD are tested on the collected turmeric Dataset in terms of accuracy, precision, recall and F1-score. YOLOv3 [22], YOLOv3-Tiny [23], IY3TM [15] models are applied and evaluated on collected dataset, to test these models performance for the turmeric plant diseases detection. It is noticed that the accuracy of ConViT-TDD is increased up to 12.49%, 8.81% and 4.99% than YOLOv3, YOLOv3-Tiny, IY3TM respectively. The precision of ConViT-TDD is improved by 12.99%, 8.83% and 4.98% compared to the YOLOv3, YOLOv3-Tiny, IY3TM algorithms, accordingly. The recall of ConViT-TDD is enhanced by 12.42%, 7.91% and 5.43% in contrasted to YOLOv3, YOLOv3-Tiny, IY3TM accordingly. Finally, F1-Score of ConViT-TDD is increased to 11.77%, 8.87% and 5.21% than YOLOv3, YOLOv3-Tiny, IY3TM respectively. From this analysis, it is observed that the proposed ConViT-TDD provides best performances than other existing models on collected data for turmeric plant diseases detection.

5. Conclusion

In this research, ConViT-TDD model is introduced to derive the relevant features for the turmeric plant diseases prediction. This model consists of four parts: input channel transformation, global-minded transformers, channel as well as spatial attention module. The inputted channel transformation minimizes computational complexity, while global-minded transformers generate a word vector. Training channel and spatial weights, respectively, allows the channel and spatial attention mechanisms to enhance the model's attention to various features and spatial placements. The model precisely locates critical traits for disease prediction in turmeric plants while successfully minimising information loss. The model detects and classifies the turmeric plant diseases, with an overall accuracy of 93.16% compared to classical models.

Funding: This research received no external funding

Conflicts of Interest: The authors declare no conflict of interest.

References

[1] S. Mitra, L. E. da Silva, H. S. Tuli, and K. Dhama, “Potential health benefits of dietary *Curcuma longa* (turmeric) supplementation on health and production of poultry: A mini-review,” *Indian Vet. J.*, vol. 99, no. 2, pp. 7–13, 2022.

[2] R. P. Yadav and G. Tarun, “Versatility of turmeric: A review the golden spice of life,” *J. Pharmacogn. Phytochem.*, vol. 6, no. 1, pp. 41–46, 2017.

[3] M. S. Sagor, M. M. Hossain, and T. Haque, “Evaluation of growth, yield and quality of turmeric genotypes (*Curcuma longa* L.),” *J. Trop. Crop Sci.*, vol. 8, no. 1, pp. 8–15, 2021.

[4] C. Sellaperumal *et al.*, “Plant-parasitic nematodes, a looming threat to turmeric cultivation in India: Results of recent surveys,” *Nematology*, vol. 1, no. aop, pp. 1–9, 2023.

[5] S. Anusuya and M. Sathiyabama, “Protection of turmeric plants from rhizome rot disease under field conditions by β-D-glucan nanoparticle,” *Int. J. Biol. Macromol.*, vol. 77, pp. 9–14, 2015.

[6] S. V. Gudkov *et al.*, “Optical methods for the detection of plant pathogens and diseases,” *Agriengineering*, vol. 5, no. 4, pp. 1789–1812, 2023.

[7] G. Kuricheti and P. Supriya, “Computer vision-based turmeric leaf disease detection and classification: A step to smart agriculture,” in *Proc. 3rd Int. Conf. Trends Electron. Informatics (ICOEI)*, 2019, pp. 545–549.

[8] L. Mohammed and Y. Yusoff, “Detection and classification of plant leaf diseases using digital image processing methods: A review,” *ASEAN Eng. J.*, vol. 13, no. 1, pp. 1–9, 2023.

[9] S. Kumar and R. Kaur, “Plant disease detection using image processing: A review,” *Int. J. Comput. Appl.*, vol. 124, no. 16, 2015.

[10] V. Janani and N. S. Mangai, “Analysis and classification of rhizome rot disease for turmeric plant using artificial intelligence,” in *Proc. Int. Conf. Sustainable Comput. Data Commun. Syst. (ICSCDS)*, 2022, pp. 411–414.

[11] M. R. Chougala and A. C. Ramachandra, “Prediction of leaf blotch in turmeric plant,” in *Emerging Res. Comput., Inf., Commun. Appl.: ERCICA 2020, Vol. 2*, Singapore: Springer, 2022, pp. 815–826.

[12] G. P. Kanna *et al.*, “Advanced deep learning techniques for early disease prediction in cauliflower plants,” *Sci. Rep.*, vol. 13, no. 1, p. 18475, 2023.

[13] P. K. Jayapal *et al.*, “Analysis of RGB plant images to identify root rot disease in Korean Ginseng plants using deep learning,” *Appl. Sci.*, vol. 12, no. 5, p. 2489, 2022.

[14] H. Waheed *et al.*, “Deep learning-based disease, pest pattern and nutritional deficiency detection system for *Zingiberaceae* crop,” *Agriculture*, vol. 12, no. 6, p. 742, 2022.

[15] V. Devisurya, R. Devi Priya, and N. Anitha, “Early detection of major diseases in turmeric plant using improved deep learning algorithm,” *Bull. Pol. Acad. Sci. Tech. Sci.*, vol. 70, no. 2, 2022.

[16] C. Rajasekaran *et al.*, “Turmeric plant diseases detection and classification using artificial intelligence,” in *Proc. Int. Conf. Commun. Signal Process. (ICCSP)*, 2020, pp. 1335–1339.

[17] S. Revathy *et al.*, “An effective approach for turmeric growth detection using multilevel linear algorithm,” *Zenodo*, 2022, doi: 10.5281/zenodo.7188915.

[18] S. Patil and G. Patil, “Identification of turmeric rhizomes using image processing and machine learning,” *Eng. Proc.*, vol. 59, no. 1, p. 34, 2023.

[19] M. Bouni *et al.*, “Impact of pretrained deep neural networks for tomato leaf disease prediction,” *J. Electr. Comput. Eng.*, 2023.

[20] M. M. Islam *et al.*, “DeepCrop: Deep learning-based crop disease prediction with web application,” *J. Agric. Food Res.*, vol. 14, p. 100764, 2023.

[21] N. Ullah *et al.*, “An effective approach for plant leaf diseases classification based on a novel DeepPlantNet deep learning model,” *Front. Plant Sci.*, vol. 14, 2023.

[22] X. Gong and S. Zhang, “An analysis of plant diseases identification based on deep learning methods,” *Plant Pathol. J.*, vol. 39, no. 4, p. 319, 2023.

[23] F. Khan *et al.*, “A mobile-based system for maize plant leaf disease detection and classification using deep learning,” *Front. Plant Sci.*, vol. 14, p. 1079366, 2023.

[24] A. S. Kini, K. V. Prema, and S. N. Pai, “Early stage black pepper leaf disease prediction based on transfer learning using ConvNets,” *Sci. Rep.*, vol. 14, no. 1, p. 1404, 2024.

[25] D. Sutiaji *et al.*, “WECNN-PDP: Weighted ensemble convolutional neural networks models to improve the plant disease prediction,” in *E3S Web Conf.*, vol. 482, p. 03003, 2024.