Migration Letters

Volume: 21, No: S2 (2024), pp. 895-912

ISSN: 1741-8984 (Print) ISSN: 1741-8992 (Online)

www.migrationletters.com

An Efficient Crop Field Disease Prediction Using A Novel Parallel Mini Patch Aggregation Neural Network (Pmpann)

Gangadevi[1], J.Jospin Jeya[2], S.Raja Ratna[3*], S.Lakshmi[4], S.Abirami[5], M. Chitra[6]

ABSTRACT:

The major sources for gross domestic product among the developing countries across the world is agriculture. The nature and amount of crop cultivated along with its yield in each region varies with soil and climate. Additionally, the crop yield is affected by the disease. Paddy is the major food crops cultivated in several parts and it succumbs to several diseases. The farmers often find it hard to predict the nature of disease over the paddy crop. For supporting the farmer, a novel Deep Convolution Mini Patch Aggregation Neural Network (DCMPANN) classifier is proposed to predict and localize the disease in paddy crop using its image. The images of crop are collected in real-time and the novel hybrid tanh and sigmoid function is developed for the proposed classifier. Using performance metrics the performance of the DCMPANN model has been evaluated. The proposed DCMPANN model demonstrates the effectiveness on disease prediction on validated with existing systems.

Keywords: *Agriculture, Classification, Deep Learning (DL), Disease, Paddy.*

1. INTRODUCTION

The agricultural sector is significant source of income or various classes of people in the world. Agricultural production aims to feed around 10 billion people by 2050 (FAO, 2017). The disease occurs in the crops plant are majorly threatens the productivity. Because of the lack of necessary infrastructure in many nations across the world, the early identification of the diseases becomes tedious. The rising penetration of computer vision using Deep Learning (DL) technology has made possible to develop an automatic disease diagnostic system (Mohanty et al. 2016).Trad[1]itionally, skilled professionals visually inspect plant tissue to determine the severity of a plant infection. The study of human diseases is expensive and inefficient, which hinders the rapid advancement of modern agriculture. (Mutka & Bart, 2015). Precision agriculture, smart homes, and other industries have strong demands for automated disease diagnostic models due to the proliferation of digital cameras and advancements in computer vision.(Wang et al. 2017). Rice is an important food crop and also a most stable food for majority of the people in the world (Muthayya et al. 2014). As the population increases, the demand for the productivity also increased. However, causes of disease in the plant has been a major threat to the production of rice crops (Smith & Myers, 2019). Numerous diseases, such as bacterial leaf blight (produced by the bacteria Xanthomonas oryzae) and brown spot (caused by the fungus Helminthosporiose) can affect rice leaves. etc. This might seriously affect the plant and affects the quality and also the yield. It is very hard to effectively and accurately detect the diseases through the naked eye. Thus, an automatic detection using computer vision helps the farmers to effective prediction of the disease (Archana & Sahayadhas, 2018).

[1,2,4,5,6]Assistant Professor, [3]Associate ProfessorDepartment of Computer Science and Engineering,SRM Institute of Science and Technology, Ramapuram, Chennai

One approach for managing the loss in crop yield due to disease infection is to capture the images of infected leaves and learn more about the diseases (Strange & Scott, 2005). As an automated solution to this problem, cameras can be installed at specific homestead divisions to automatically take pictures. These images can be uploaded to a targeted illness analysis system, which will diagnose the condition and offer details on the condition along with possible treatments. . The detection of disease in the plants using the image processing techniques will help in the lessening of farmer's dependence to preserve the agricultural products. Image processing technique comprises of various process such as image acquisition, pre-processing, segmentation, feature extraction and classification (Kamal et al. 2018). These process are performed upon physical assessment of infected crops. Generally, the disease in the plant can be detected by using its significant parks like stems, leaves, etc. The indication of the plant disease varies from each other.In addition, each plant disease has its own specific characteristics, such as color, size, and texture. Few illnesses can cause a person to become yellow, while few can turn someone who is green into a brown color. (Prajapati et al. 2017). Furthermore, the illness can occasionally alter a leaf's form without altering its color, and vice versa. The area of the pant with disease features should be obtained once the infected sections have been divided. Manually predicting diseases with the naked eye is time-consuming, expensive, and produces erroneous results. Predicting the disease is a difficult task that is prone to inaccuracy. These problems result from a lack of understanding of the plant. Accordingly, as has been shown over the past few decades, plant diseases have an impact on production if they are not discovered or anticipated in advance. (Pinki et al. 2017).

A novel DL based framework called Parallel Mini Patch Aggregation Neural Network (PMPANN) was proposed here to predict the disease in the rice plant. This network includes a number of layers which employs shortcut connections during the traversal of data. With more number of layers and skipping through shortcuts, the network produces better results and avoids negative outcomes. You Only Look Once (YOLO), a network renowned for object detection applications, is proven to be incredibly fast in image processing. However, the significant drawbacks of the YOLO were that it low sensitivity rate. Hence, in the classification layer of the proposed PMPANN, a new layer called Logarithmic YOLO (Log-YOLO) is introduced in end for performing the final classification task. This Log-YOLO uses a new logarithm based loss function to enhance the recalling capacity of the prediction model. A fusion of ReLU and tanh activation functions is employed to train the network, which can improve the performance.

The remaining part of this paper is structured with section-2 discusses the existing disease classification model for plants; section-3 elaborates the methodology adopted to develop the crop prediction model, section-4 explains the results and discussion as obtained from the simulation and section-5 gives the conclusion and suggestion to proceed the work further.

2. LITERATURE REVIEW

Waheed, et al. (2020) introduced an optimized DenseNet model using the DL techniques for detecting the crop diseases like corn leaf spot, leaf rust and northern corn leaf. This model gives the classification accuracy of 98.06%. The classification accuracy was observed to be encouraging; however, this model does not consider the more types of diseases. Syarief & Setiawan, (2020) developed CNN based maize leaf disease classification model such as Virtual Geometry Group 16 (VGG16), Virtual Geometry Group 19 (VGG19), AlexNet, GoogleNet, Residual Network 50 (ResNet50), Inception-V3 and Residual Network 101 (ResNet101) using the data of 200 images. The AlexNet provides the sensitivity, specificity and accuracy of 95.08%, 93% and 93.5%.

Li et al. (2020) used Inception-V3 and VGGNet-16 for the classification of various level of ginkgo leaf disease using the training dataset comprises of 1322 images. The Incetion-V3 model obtained the consistent accuracies in both laboratory and field conditions. The accuracy of VGG model in laboratory and infield conditions was found to be higher as 98.44% and 92.19%; however, their results are quite different. Azim et al. (2021) introduced a disease prediction model for rice plant considering the diseases such as brown spot, bacterial leaf blight and leaf smut. The data utilized for developing model was obtained for UCI repository. The various features such as shape, color and texture are extracted from the input images. The XGBoost was utilized for classification and accomplished an accuracy of 86.58% with a small number of features. Panigrahi, et al. (2020) used supervised learning techniques like Decision Tree (DT), Naïve Bayes (NB), k-Nearest Neighbor (KNN), Random Forest (RF) and Support Vector Machine (SVM) for the prediction of disease in maize plant. The RF model gives high accuracy of about 79.23% than the other models. However, this accuracy was not found to be better than the existing models.

Sethy et al. (2020) examined the performance of the 13 CNN model developed using the transfer learning approach to classify the four various leaf diseases like blast, bacterial blight, brown spot and tungro. The deep features plus SVM gives the better performance. However, this model is applicable for detection of only few diseases in rice plants. Sulistyaningrum, et al. (2020) developed multi-SVM model by consolidating SVM for the classification of more than two classes with a total of 240 images.The classification system with multilevel SVM produces performance with an accuracy value of 86.51% for 70% of training and 30% of testing data. Chen, et al. (2020) used the DenseNet and Inception model to detect the diseases in rice plant using the 500 images of disease affected rice plant. The prediction accuracy of 94.07% was obtained. An accuracy of 98.63% was obtained, if the multiple disease was considered.

Rautaray, et al. (2020) introduced transfer learning technique to predict the diseases of the crops. An accuracy of 90% was achieved with the use of image processing application with transfer learning. Ramesh & Vydeki, (2020) developed a DNN model optimized with Jaya algorithm for the prediction of disease in the rice leaf. The DNN classifier was used in which the weights are updated utilizing Jaya Optimization Algorithm (JOA). The model gives an accuracy of 92% for sheath rot, 95.7% for bacterial light, 98.9% for blast infected, 90.57% for normal leaf and 94% for brown spot.

Babu et al. (2020) classified the most frequent rice plant disease like lea smut, sheat blight, leaf blight and brown spot using the deep learning techniques and achieved the accuracy of 99.45%. Daniya & Vigneshwari (2022) used the KNN and Butterfly algorithm to classify the diseases in the plant and achieved the high accuracy of about 98.92%.

Motivated by the convenience of CNN and IoT, Patil & Kumar, (2022) introduced a data fusion system called as “Rice-Fusion” for detecting the disease in the rice using the image dataset manually collected using the agro-meteorological sensors and camera. This method achieved the accuracy of 95.31%.

Kathiresan et al. (2021)presented a portable tool to farmers and agrarian associations for the detection of illnesses in rice leaves using a model based on transfer learning. An alternating number of illness samples was employed by employing the Generative Adversarial Network (GAN). The model had an average cross-validation accuracy of 98.79% when tested using the GAN augmented dataset. It was noted, nevertheless, that it is expensive. Sowmyalakshmi et al. (2022) introduced CNNIR-OWELM model for the diagnosis of diseases in the rice plant. The image of the plant is taken using IoT devices, which then send it to cloud servers for analysis. The extracted feature vectors are fed into FPA-WELM model to classify and an accuracy, specificity and sensitivity of 94.2%, 96.1% and 90.5% was achieved.

In the prediction of plant disease, the classical model like human vision model are generally employed. accuracy of human vision model relies on the visual perception of the expert. The soft computing method are used being utilized to detect the various classes of diseases. The CNN was found to work excellently with the visual images. The CNN can learn the features of images like color, orientation, spatial characteristics, temporal characteristics, shape, etc. The CNN contains multi-layer arrays to detect the objects by learning the pixels of the images. The difference between the CNN and other NN is that CNN accepts the input data as a 2D array and manipulate directly on the images rather than focused in feature production. However, the existing CNN model experience the drawbacks like it requires mode time for data training, gives low accuracy and high RMSE due to the dynamic characteristics. To overcome this drawbacks, the concept of processing the images through two smaller networks in parallel instead one larger network is implemented in the model.

3. METHODOLOGY

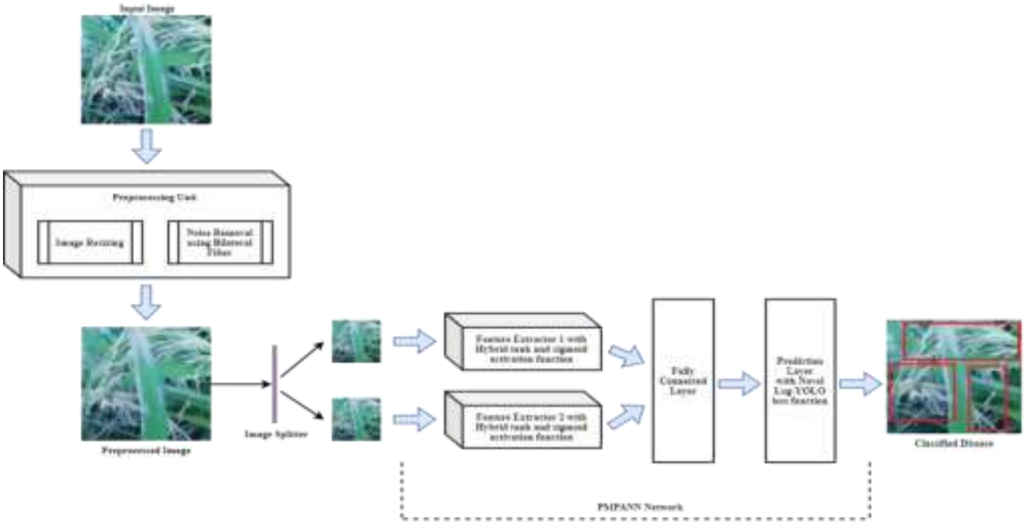

Input Image
Preprocessing Unit
Image Resizing
Noise Removal using Bilateral Filter
Preprocessed Image
Image Splitter
Feature Extractor 1 with Hybrid tanh and sigmoid activation function
Feature Extractor 2 with Hybrid tanh and sigmoid activation function
Fully Connected Layer
Prediction Layer with Novel Log-YOLO loss function
Classified Disease
PMPANN Network

Figure 1. Proposed Framework of Paddy Disease Prediction and Localization

The prediction and localization of paddy crop disease was the main focus of the proposed study. Brown spot (Cochiobolus miyabeanus), leaf smuggling, and bacterial leaf blight (Xanthomonas oryzae) are the three primary paddy diseases that are taken into consideration. (Entyloma oryzae). The DL framework proposed for identifying and localizing the disease is given in figure 1. The input image was initially passes through preprocessing unit which performs image resizing process and noise filtering process. Then, the preprocessed image sent to developed DL based PMPANN model classifier. The significance in implementation of the DL mechanism the ability to carried out feature extraction by itself. The input image passes through numerous layers of the PMPANN model for learning the characteristics of all the three diseases. Such extensive learning enhances the prediction accuracy.

The PMPANN models comprises of three sub units, (i) two feature extractors which work parallel, (ii) fully connected layer to combine the two feature maps into one, and (iii) novel Log-YOLO prediction layer to predict and localize the disease in the crop image. The use of parallel processing in the feature extraction greatly helps in the improving the speed of training, and the extensive learning of the network. The following sub sections explain the detailed working of the proposed framework.

3.1. Dataset Description

The dataset is collected by manually segregating the leaves that are infected into their respective disease classes. The samples collected from the Gujarati farmers and identified the English names then verified by consulting the experts of agriculture field. In order to process the photos, they are lowered to the required resolution. data augmentation is a technique used to improve accuracy. The related illnesses, which included leaf smut, bacterial leaf blight, and brown spot, were converted to JPG format. 24500 samples that were gathered from the Tirur Research Station make up the dataset. Table 1 contains a tabular description of the dataset in detail.

Table 1. Description of Annotated Leaves Dataset

Class	Original	Rotated	Total
Bacterial leaf blight	4500	3500	8000
Brown spot	4500	3000	7500
Leaf smut	3700	5300	9000
Total	12700	11800	24500

3.2. Data Preprocessing

Data preprocessing unit comprises of two sub units, namely, image resizing and noise filtering. The resizing of input images is performed for two main reasons; they are: to make all the images to be in uniform size for processing, to reduce the time taken and computational complexity caused by processing larger images. Once the images are resized to lower dimensions, they are passed through a noise removal filter called Bilateral filter. The primary purpose of the bilateral filter is to smoothen the pictures. It is a non-linear, edge-preserving, noise reduction filter. By using this method, neighboring pixels' weighted average intensities are substituted for the image's pixels. The expression for the bilateral filtering is,

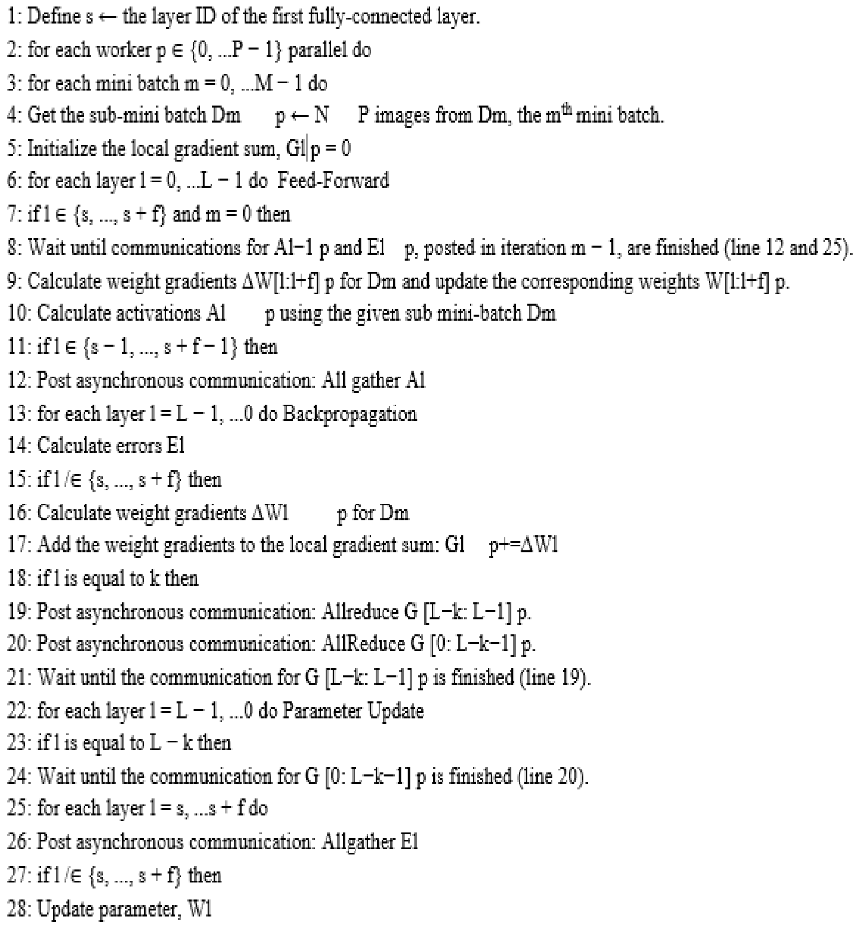
$$I_{preprocessed}(x) = 1/w_p \sum_{x_k \in \Omega} I(x_k) F_r(\|I(x_k) - I(x)\|) G_s(\|x_k - x\|) \qquad (1)$$

where the normalized weight w_p is defined as,

$$w_p = \sum_{x_k \in \Omega} F_r(\|I(x_k) - I(x)\|) G_s(\|x_k - x\|) \qquad (2)$$

I - resized image; $I_{preprocessed}(x)$ - preprocessed image; x - coordinates of the current image pixel; Ω - window centered in x; F_r - kernel range for smoothing the intensity differences;
G_s - spatial kernel for smoothing the coordinate differences.

Pseudo Code for Data Parallel CNN Training



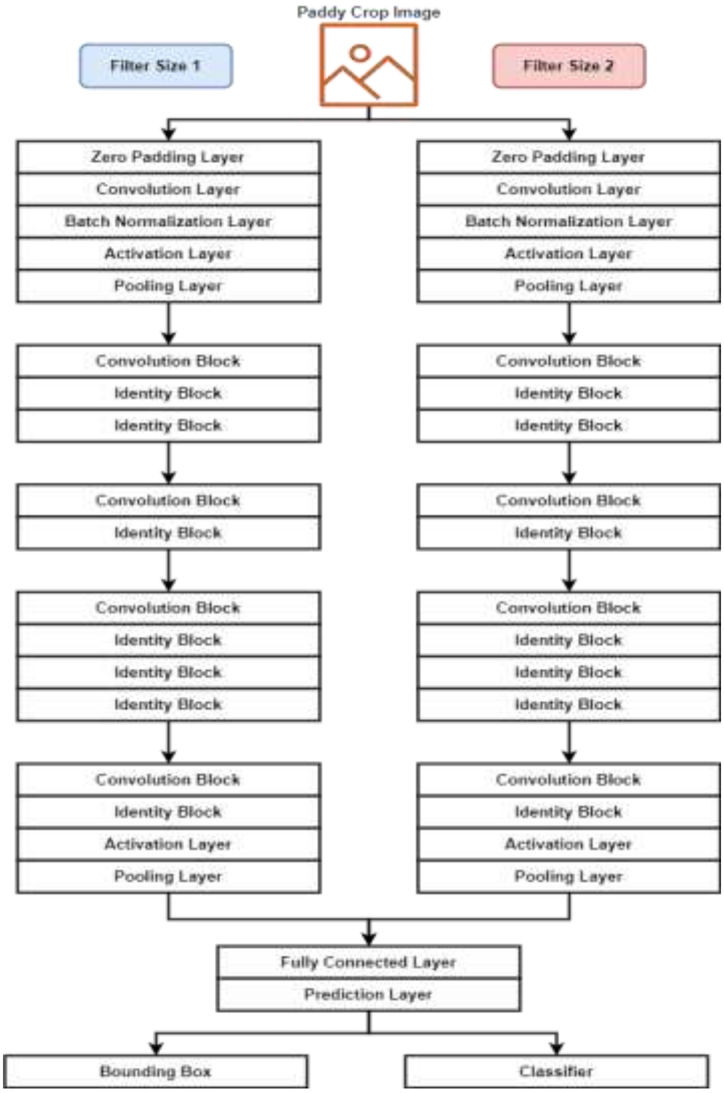
```
1: Define s ← the layer ID of the first fully-connected layer.
2: for each worker p ∈ {0, ...P − 1} parallel do
3: for each mini batch m = 0, ...M − 1 do
4: Get the sub-mini batch Dm      p ← N     P images from Dm, the $m^{th}$ mini batch.
5: Initialize the local gradient sum, Gl|p = 0
6: for each layer l = 0, ...L − 1 do  Feed-Forward
7: if l ∈ {s, ..., s + f} and m = 0 then
8: Wait until communications for Al−1 p and El    p, posted in iteration m − 1, are finished (line 12 and 25).
9: Calculate weight gradients ΔW[l:l+f] p for Dm and update the corresponding weights W[l:l+f] p.
10: Calculate activations Al       p using the given sub mini-batch Dm
11: if l ∈ {s − 1, ..., s + f − 1} then
12: Post asynchronous communication: All gather Al
13: for each layer l = L − 1, ...0 do Backpropagation
14: Calculate errors El
15: if l /∈ {s, ..., s + f} then
16: Calculate weight gradients ΔWl         p for Dm
17: Add the weight gradients to the local gradient sum: Gl     p+=ΔWl
18: if l is equal to k then
19: Post asynchronous communication: Allreduce G [L−k: L−1] p.
20: Post asynchronous communication: AllReduce G [0: L−k−1] p.
21: Wait until the communication for G [L−k: L−1] p is finished (line 19).
22: for each layer l = L − 1, ...0 do Parameter Update
23: if l is equal to L − k then
24: Wait until the communication for G [0: L−k−1] p is finished (line 20).
25: for each layer l = s, ...s + f do
26: Post asynchronous communication: Allgather El
27: if l /∈ {s, ..., s + f} then
28: Update parameter, Wl
```

3.3. Proposed Parallel Mini Patch Aggregation Neural Network (PMPANN)

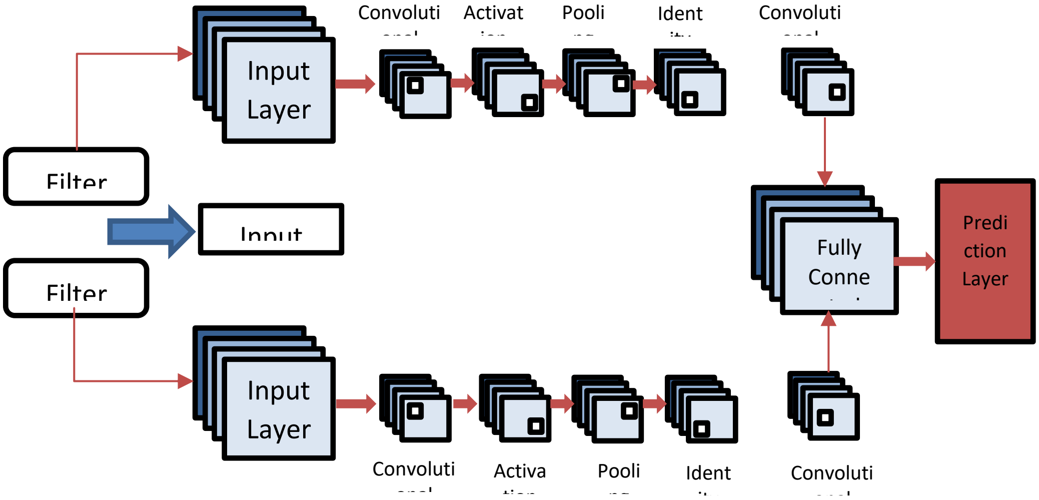
One main drawback of using CNN is the computational complexity. The computations carried out in a CNN is intense. This makes the network processing slower than expected. As an effort to reduce the computational intensity and to reduce the time taken for training, the processing of image in the network is taken in parallel fashion. Parallelization is introduced in the proposed network by processing the same input in two different filter sizes (3x3 and 5x5) with two smaller convolutional layers operated independently at the same time. This will result in reduced computational time for the training phase. With the proposed algorithm, the given image is classified based on the sub types of disease of paddy crop with accuracy similar or close to inter-observer agreements.

Pseudo Code for Mini-Batch CNN Training

```
for each mini batch $m = 0, ...M - 1$ do
    Initialize $\Delta W = 0$
    Get the mth mini batch, Dm.
    for each layer $l = 0, ...L - 1$ do
        Calculate activations Al based on Dm.
    for each layer $l = L - 1, ...0$ do
        Calculate errors El.
        Calculate weight gradients $\Delta Wl$.
    for each layer $l = 0, ...L - 1$ do
        Update parameters, Wl and Bl
```



(a)



(b)

Figure 2. Layers of the proposed PMPANN

This PMPANN model employs data parallelism strategy which splits the data into two chunks and processes parallel data training. The computational overload of the network is reduced, since the data size is reduced into half for processing. This aspect improves the speed of training. Hence, the preprocessed input image is split into two smaller images and are processed with two smaller networks. The two smaller dimensional images pass through feature extractors of the network with two different filter sizes 3x3 and 5x5.

The feature extractor outputs are consolidated by a fully connected layer. This final feature map is passed through the prediction layer for establishing the bounding box of the disease found and the name of the disease recognized. The data parallelization helps in faster training (less time and high speed) and reduced computational overload.

The learning applied for the proposed network is residual learning. Consider $\mathrm{H(y)}$ as a mapping to be fit using stacked layers of the network, where y indicates the inputs to these layers. The explicit residual function of $\mathrm{H(y)}$ is given by

$$\mathrm{F(y) \coloneqq H(y) - y} \tag{3}$$

Therefore, the overall function of the shortcut link becomes, $\mathrm{F(y) + y}$.

If the stacked layers are built as identity mappings, a deeper network will be developed. This deeper network will have training error lesser than that achieved for a shallower network. If the identity mappings are optimum, then the solvers will bring the weights of the nonlinear multi-layers towards 0 to drive towards identity mappings.

The building block of the stacked layers is expressed as,

$$\mathrm{x = F(y, \{w_i\}) + y} \tag{4}$$

Where, x is the output vector, $\mathrm{w_i}$ refers to the weights, and $\mathrm{F(y, \{w_i\})}$ indicates the residual mapping that is to be learned.

$$\mathrm{F = w_{i+1} \delta(w_i y)} \tag{5}$$

Here, δ is the hybrid tanh and sigmoid activation function expressed as,

$$\delta(z) = \tanh \tanh (\sigma(z)) \quad (6)$$

Where, z is the input, $\sigma(.)$ is sigmoid activation defined as follows.

$$\sigma(z) = \frac{1}{1+e^{-z}} \quad (7)$$

The function performed by the shortcut connection and element wise summation is given as $F + x$. Thus the shortcut links do not introduce any extra parameter or computational complexity. The outputs of these stacked layers of two feature extractors are the two feature mappings of the two input images.These two learned feature mappings are combined using a fully connected layer to obtain the final feature map. The final feature map is sent to a prediction layer which is based on Log-YOLO with a novel loss function. Because of the nature of convolutional function of the proposed network, multiple bounding boxes are predicted in parallel fashion. Once the bounding boxes of the defects in the plant leaves are detected, the name of the disease is identified.

The size of the input image is (128 128 3) 49152 pixels, in final layer of the network the size of the image size is (4 4 1024) 16384 pixels. The table 2 shows the type, kernel size, and the output size of the layers of the developed PMPANN.

Table 2. Parameters of the Proposed P-DCMP-ANN

Name	Type	Kernel Size	Output Size
Zero Padding	Padding	3x3	306 x 303 x 3
Conv1	Convolution	3x3	150 x 150 x 3
Batch_norm	Batch Normalization	3x3	150 x 150 x 64
Activation	Activation Function	-	150 x 150 x 64
Max_pooling	MaxPooling2D	2*2	74 x 74 x 64
res2a_branch2a	Conv2D	3x3	74 x 74 x 64
bn2a_branch2a	Batch Normalization	3x3	74 x 74 x 64
activation_1	Activation	-	74 x 74 x 64
res2a_branch2b	Conv2D	3x3	74 x 74 x 64
bn2a_branch2b	Batch Normalization	3x3	74 x 74 x 64
activation_3	Activation	-	74 x 74 x 64
res2a_branch2c	Conv2D	3x3	74 x 74 x 64
res2a_branch1	Conv2D	3x3	74 x 74 x 256
bn2a_branch2c	Batch Normalization	3x3	74 x 74 x 256
bn2a_branch1	Batch Normalization	3x3	74 x 74 x 256
Add	Add	3x3	74 x 74 x 256
Activation_3	Activation	-	74 x 74 x 256
Avg_Pool	AveragePooling2D	3x3	1 x 1 x 2048
Fully_conn	FullyConnectedLayer	1x1	2048
Log_Yolo_clf_0	Flatten	1x1	2048
Log_Yolo_clf_1	Dense	1x1	2048
Log_Yolo_clf_2	Dropout	1x1	2048
Log_Yolo_clf_3	Dense	1x1	1573

The major advantage of PMPANN is the identity shortcut link, where one or more network layers are skipped. The stacking of layers will not degrade the performance of the network because of stacking identical mappings onto the current model of network will make the new network model perform in the same manner. This denotes that the deeper neural network models will not give training errors more compared to the shallower networks.

PMPANN explicitly allows to perform such operations. In the proposed architecture, the final classification layer used is a novel Log-YOLO classification layer. The feature mapped image output is sent as input to the novel Log-YOLO layer of classification. Initially, Log-YOLO divides the input image into several fragments of images in the form of a square grid with a particular dimension. The Log-YOLO classifier reliably and accurately locates and identifies the patterns in the image will might help in the recognition in the type of the disease. The proposed loss function of Log-YOLO helps in improving the recall rate of classification.

3.3.1. Training Phase

The approach of disease identification in Log-YOLO necessitates to segment the image into several grid cells and every grid cells is essential for the detection and prediction of the bounding boxes. The predictions of grid size and bounding box are done per grid cell in order to preserve sufficient resolution with lesser output prediction for training. Having a grid will perform some detection in the same cell of grid and for the purpose of simplicity (and power computation).

The suggested Log-YOLO employs a bespoke objective function that is fine-tuned to increase stability (penalize loss from grid cells that do not have an object) and weigh dimension error in smaller boxes more than that in larger boxes. Keras and TensorFlow have the conventional loss definitions.

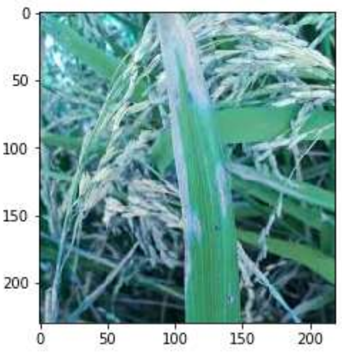
During training, this loss function is optimized to improve the predictions of the network. As the iterations increase, the loss decreases gradually. When the loss becomes zero, the developed network model is fully trained.

3.3.2. Testing Phase

Once the training is complete, the trained network is ready for the real-time deployment. In this phase, a test image from the test dataset is sent to the trained network without the ground truth image and the class label. The fully trained network will be able to find the class of the disease present in the crop and also finds the exact location of the disease spot in the paddy crop leaves.

4. RESULTS AND DISCUSSION:

The suggested PMPANN is implemented using high configuration GPU and Python programming for the purpose of classifying illnesses in paddy crops. This section covered the findings from the simulation of the suggested PMPANN model of disease prediction and localization. The input image of the paddy leaf used to classify the disease is seen in figure 3.



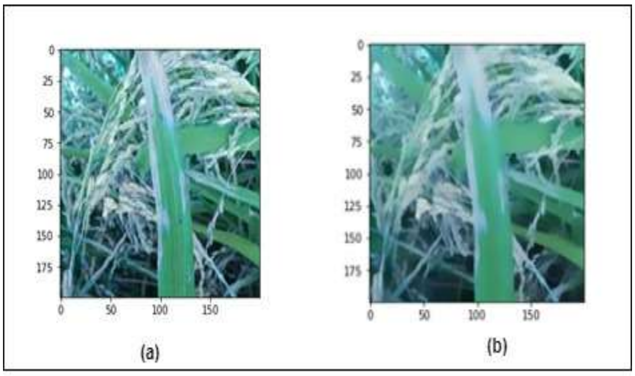
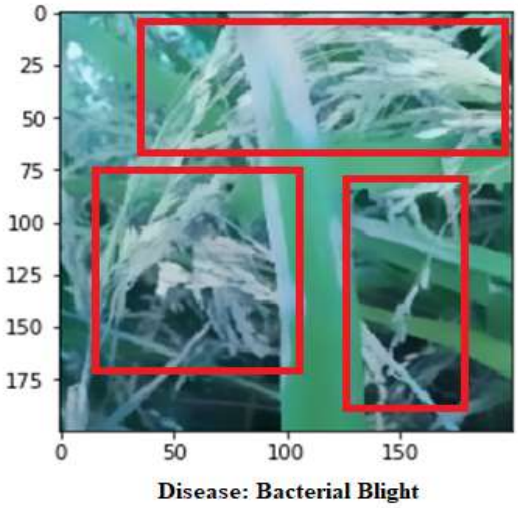


Figure 3. Input Image of Paddy Leaf **Figure 4. Pre-processed Images (a) Resized Image (b) Noise**

Filtered Image

This input image is sent through pre-processing unit, which comprises of sub units for image resizing and noise filtering using bilateral filtering. The resized image is displayed in figure 4 (a) and noise removed image is displayed in figure 4 (b). This pre-processed image is sent to PMPANN model for disease classification and localization. The output of the classification and the localization of the disease spot using bounding box is presented in figure 5.



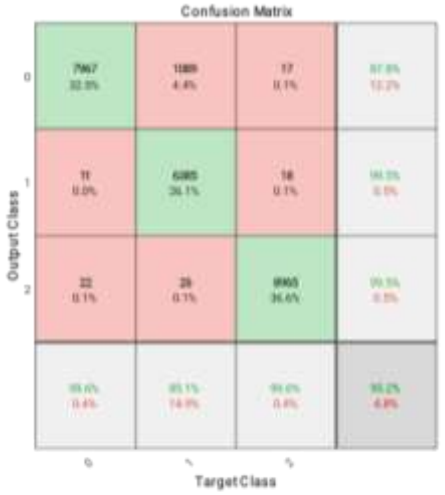


Figure 5. Classification Output **Figure 6. Confusion Matrix of the Proposed**

PMPANN Model

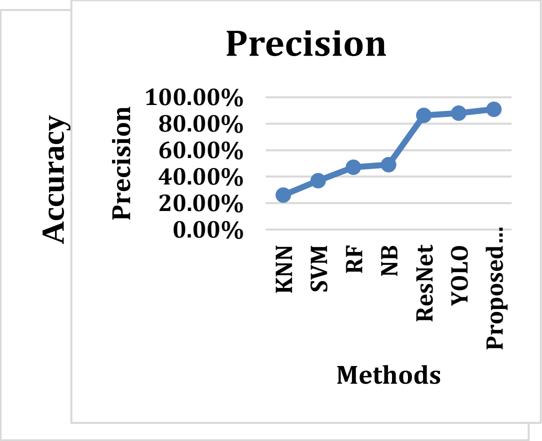
The classification performance is evaluated in terms of a confusion matrix. The confusion matrix determined for the proposed PMPANN model is displayed in figure 6. This matrix summarizes the model's performance in four parameters namely rue Positive (TP), True Negative (TN), False Positive (FP) and False Negative (FN). The TP are correctly classified values of the event class, where TN are incorrectly classified values of the event class. The FP are correctly classified values of the non-event class, where TN are incorrectly classified values of the non-event class. Using the TPs, TNs, FPs, and FNs, four performance measures namely, accuracy, precision, recall, and F1-score are calculated. The performance of the PMPANN classification model was compared with the state-of-the-art methods and existing classifiers like KNN, SVM, RF, NB, ResNet, and YOLO. The parameters considered for the comparison are: accuracy, precision, recall, f1-score, time taken for classification, mAP, and speed (frames per second).

The ratio of the total of TP and TN to the total of TP, FP, TN, and FN is known as the prediction accuracy.

$$\text{Accuracy}(\%) = \frac{\text{TP+TN}}{\text{TP+TN+FP+FN}} \times 100 \qquad (8)$$

The comparison among the different classifier in terms of accuracy is given in figure 7. The proposed PMPANN classifier showed the accuracy rate of 95% which is very high compared to the existing models. The accuracy rate of the proposed PMPANN classifier shows an improvement of 66.67%, 52.63%, 45.61%, 43.86%, 7.79%, and 4% compared to KNN, SVM, RF, NB, ResNet, and YOLO, respectively.

Figure 7. Comparison on prediction accuracy Figure 8. Comparison on prediction precision



Precision of prediction is defined as the ratio of TP to the sum of TP and FP.

$$\text{Precision}(\%) = \frac{\text{TP}}{\text{TP+FP}} \times 100 \qquad (9)$$

The comaprison on precision performance over given data is shown in figure 8. The precison of the proposed model is about 91% which is nearly twice than that of random forest and Naïve Bayes. The precision rate of the proposed PMPANN classifier shows an improvement of 71.43%, 59.34%, 48.35%, 46.15%, 5.27%, and 3.41% compared to KNN, SVM, RF, NB, ResNet, and YOLO, respectively.

Recall rate of prediction is defined as the ratio of TP to the sum of TP and FN.

$$\text{Recall}(\%) = \frac{\text{TP}}{\text{TP+FN}} \times 100 \qquad (10)$$

The comparison among the different classifier in terms of recall is given in figure 9. The PMPANN classifier showed the recall of 91.65% which is very high compared to the existing models. This improvement in recall rate was possible because of the proposed novel Log-YOLO loss function. The recall rate of the proposed PMPANN classifier shows an improvement of 65.22%, 51.09%, 43.48%, 42.39%, 4.35%, and 7.61% compared to KNN, SVM, RF, NB, ResNet, and YOLO, respectively.

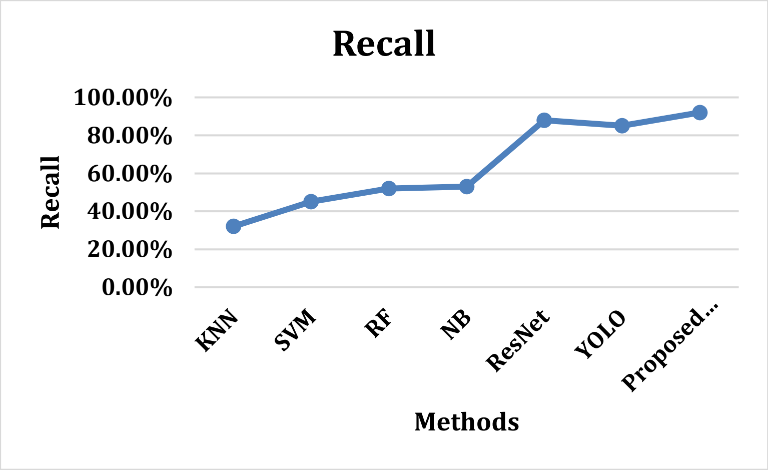


Figure 9. Comparison on prediction recall

F1-score of prediction is defined as the ratio of twice the TP to the sum of twice the TP with FP and FN.

$$\mathrm{F1-score}(\%) = \frac{2\mathrm{TP}}{2\mathrm{TP}+\mathrm{FP}+\mathrm{FN}} \times 100 \qquad (11)$$

The comparison on F1-score performance over given data is shown in figure 10. The F1-score of the proposed model is about 91.24% which is nearly twice than that of random forest and Naïve Bayes. The F1-score of the proposed PMPANN classifier shows an improvement of 72.53%, 54.94%, 49.45%, 46.15%, 7.14%, and 1.09% compared to KNN, SVM, RF, NB, ResNet, and YOLO, respectively.

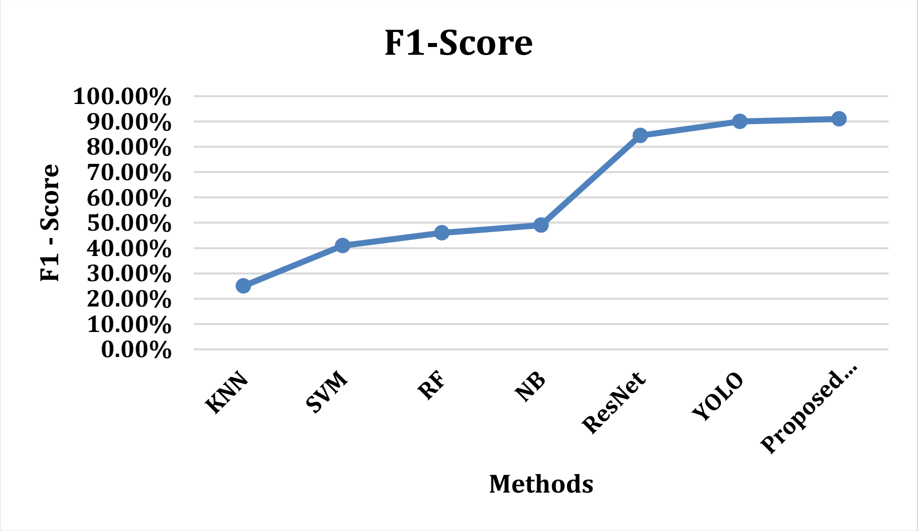


Figure 10. Comparison on prediction F1-score

The Figure 11 shows the comparison of the classifiers KNN, SVM, RF, NB, and proposed PMPANN based on the time taken for the classification of one input image. It can be seen that the proposed PMPANN shows 43.8%, 36.55%, 14.4%, 25.2%, 18.22%, and 10.24% improvement than KNN, SVM, RF, NB, ResNet, and YOLO in computational time, respectively.

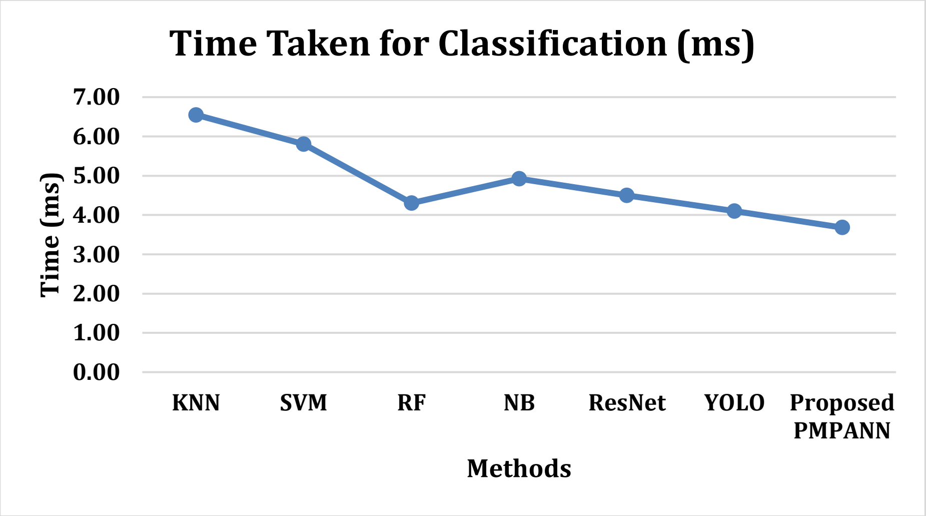


Figure 11. Comparison on classification time

The figure 12 shows the comparative depitction of mAP obtained for KNN, SVM, RF, NB, and the proposed PMPANN. The proposed PMPANN has achieved 94%, whereas, the mAP values of KNN, SVM, RF, and NB are 47.6%, 58.3%, 61.7%, and 65.8% respectively. The mAP of the proposed PMPANN shows 49.36%, 37.97%, 34.36%, 30%, 8.83%, and 5% improvement compared to KNN, SVM, RF, NB, ResNet, and YOLO respectively.

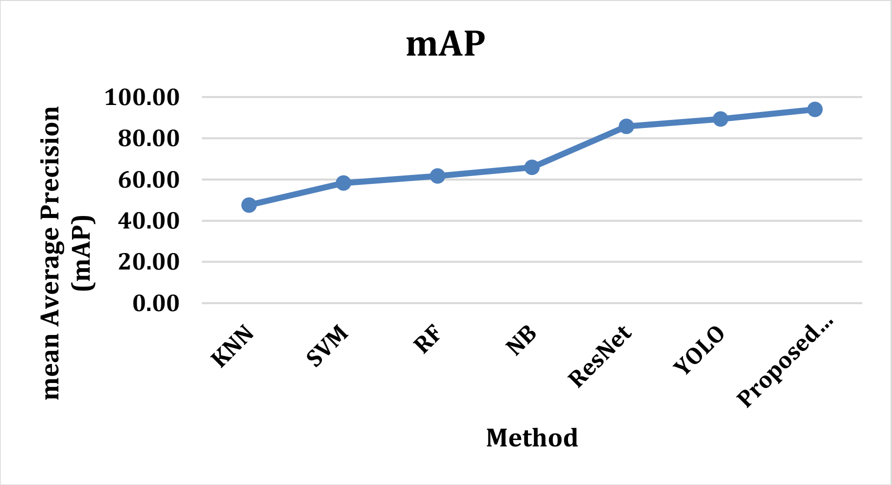


Figure 12. Comparison on mean Average Precision (mAP)

The figure 13 shows the comparative illustration of speed (frames per second) obtained for KNN, SVM, RF, NB, and the proposed PMPANN. The proposed PMPANN can process 25.8 frames per second, whereas, the KNN, SVM, RF, and NB can process 15.3, 18.2, 20.9, and 23.1 frames per second respectively. Thus the speed of the proposed PMPANN shows 40.69%, 29.45%, 18.99%, 10.46%, 16.28%, and 10.85% of improvement compared to KNN, SVM, RF, NB, ResNet, and YOLO respectively.

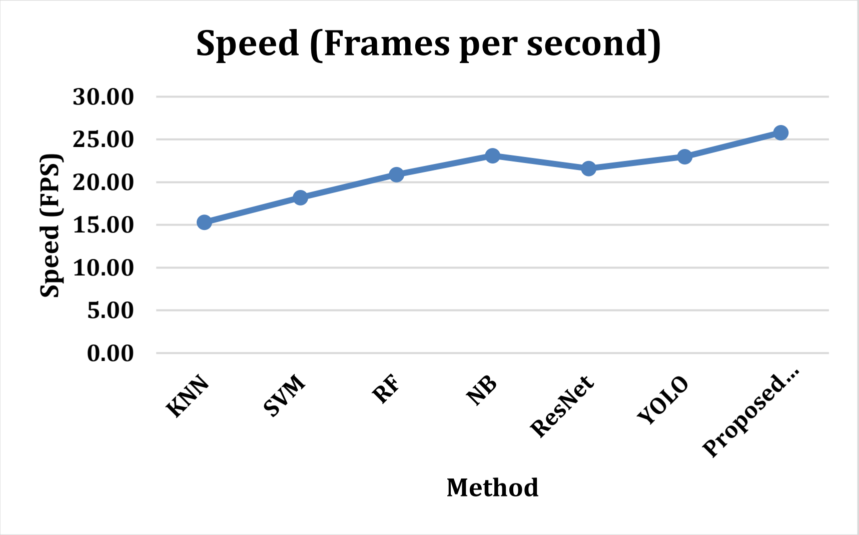


Figure 13. Comparison on speed

The figure 14 shows the comparison of parallel strategy and sequential strategy in terms of speed and time taken for classification. The networks ResNet and YOLO employ sequential strategy, whereas, the proposed PMPANN employs parallel strategy. The data parallelism concept reduces the processing overload for the two feature extractor units and thus enables the model to process more number of frames in reduced amount of time.

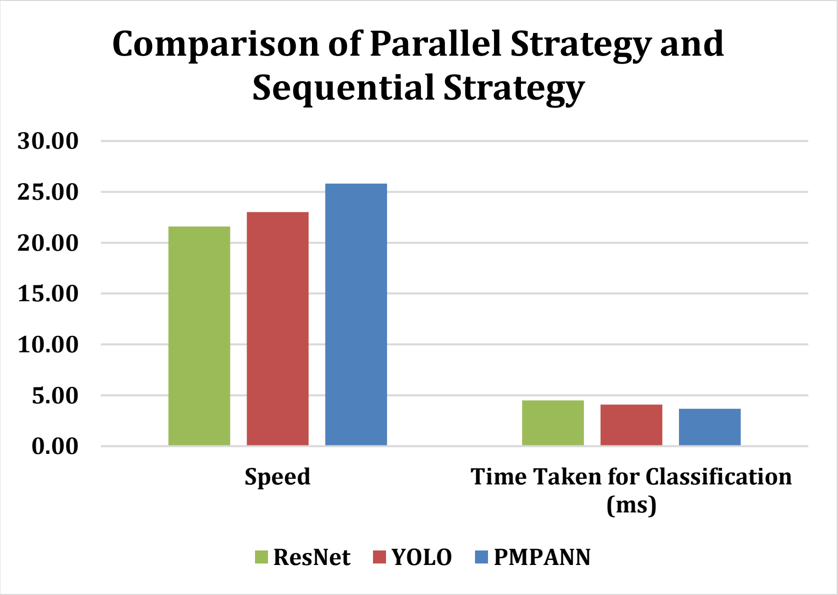


Figure 14. Comparison Parallel Strategy and Sequential Strategy

Table 3 provides the accuracies achieved by various prediction models for each class in the dataset. It can be seen that the accuracies attained by the proposed PMPANN prediction model are better when compared to those obtained by the existing state-of-the-art ML models.

Table 3. Comparison of prediction accuracy of proposed and existing methods

Method	KN N	SVM	RF	NB	ResN et	YOL O	Propose d

							PMPANN
Bacterial leaf blight	32.7	44.8	51.2	54.6	88.01	91.2	96.33
Brown spot	33.9	46.7	50.6	53.2	84.9	87.5	96.53
Leaf smut	31.5	48.2	53.3	52.8	89.4	93	97.88
Total Accuracy	31.66	45	51.67	53.33	87.6	90.7	95

Table 4 compares the parameters of accuracy, precision, recall, F1-score, mean average precision (mAP), frames per second, and classification time. It is evident that the proposed approach processes more number of frames by maintaining its accuracy. As a result, a consistent and better performance is seen in the proposed PMPANN model through the obtained values of mAP, accuracy, precision, recall, and F1-score, mAP, speed, and classification time.

Table 4. Comparison of performance measures of proposed and existing methods

Method	**KNN**	**SVM**	**RF**	**NB**	**ResNet**	**YOLO**	**Proposed PMPANN**
Accuracy (%)	31.66	45	51.67	53.33	87.6	90.7	95
Precision (%)	26	37	47	49	86.2	87.9	91
Recall (%)	32	45	52	53	88	85	92
F1-score (%)	25	41	46	49	84.5	90	91
Mean Average Precision (mAP) (%)	47.6	58.3	61.7	65.8	85.7	89.3	94
Frames Per Second	15.3	18.2	20.9	23.1	21.6	23	25.8
Time taken for Classification(ms)	6.55	5.8	4.3	4.92	4.5	4.1	3.68

It can be seen from the results that, when YOLO is employed individually for classification, the recall rate is seen to be lesser compared to ResNet, while other parameters are seen to be considerably better than ResNet. On the other hand, the proposed PMPANN model employs a novel Log-YOLO loss function uses logarithmic function to improve better estimates of probability, thereby correctly identifying TPs. This characteristic not only improves the recall rate, but also improves the parameters that involve TP, such as, accuracy, precision, F1-score, and mAP. The use of parallel feature extractors that employ residual learning processes two smaller images in reduced time period. Thus the parallelization concept not only minimizes the time taken for classification, but also increases the speed of processing and enables the model to process up to 25.8 frames per second, while the non-parallel models such as ResNet and YOLO could process only 21.6 and 23 frames per second, respectively.

5. CONCLUSIONS

This study suggested using a brand-new deep learning framework called PMPANN to identify and forecast rice crop disease. This innovative framework combined the benefits of faster saturation (convergence) and differentiability by utilizing a hybrid tanh and sigmoid activation function. A novel loss function was employed to improve the model's recall rate. Furthermore, the parallelization concept was utilized by the proposed classification model to process the incoming image inputs. To cut down on calculation time, the images were analyzed in parallel using two smaller CNN-based models. The model's performance is assessed using a number of performance. Among all the classifiers, the proposed PMPANN model was found to provide better results. In the future, this model can be developed as an application, where the farmers can upload the pictures themselves to get the predicted output.

REFERENCES:

1. Archana KS, Sahayadhas A. Automatic rice leaf disease segmentation using image processing techniques. International Journal of Engineering & Technology. 2018;7(3.27):182-5.
2. Archana KS, Sahayadhas A. Computer Vision For Predicting Unhealthy Region Of Rice Leaves- A Review. JEP.
3. Azim, M. A., Islam, M. K., Rahman, M. M., & Jahan, F. (2021). An effective feature extraction method for rice leaf disease classification. Telkomnika, 19(2), 463-470.
4. Chen, J., Zhang, D., Nanehkaran, Y. A., & Li, D. (2020). Detection of rice plant diseases based on deep transfer learning. Journal of the Science of Food and Agriculture, 100(7), 3246-3256.
5. FAO F. The future of food and agriculture–Trends and challenges. Annual Report. 2017 Oct 15.
9. Li, K., Lin, J., Liu, J., & Zhao, Y. (2020). Using deep learning for Image-Based different degrees of ginkgo leaf disease classification. Information, 11(2), 95.
10. M. Mutka and R. S. Bart, “Image-based phenotyping of plant disease symptoms,” Frontiers in Plant Science, vol. 5, article no. 734, 2015.
11. Mohanty SP, Hughes DP, Salathé M. Using deep learning for image-based plant disease detection. Frontiers in plant science. 2016 Sep 22;7:1419.
12. Muthayya S, Sugimoto JD, Montgomery S, Maberly GF. An overview of global rice production, supply, trade, and consumption. Annals of the new york Academy of Sciences. 2014 Sep;1324(1):7-14.
13. Panigrahi, K. P., Das, H., Sahoo, A. K., & Moharana, S. C. (2020). Maize leaf disease detection and classification using machine learning algorithms. In Progress in Computing, Analytics and Networking (pp. 659-669). Springer, Singapore.
14. Ramesh, S., & Vydeki, D. (2020). Recognition and classification of paddy leaf diseases using Optimized Deep Neural network with Jaya algorithm. Information processing in agriculture, 7(2), 249-260.
15. Rautaray, S. S., Pandey, M., Gourisaria, M. K., Sharma, R., & Das, S. (2020). Paddy Crop disease prediction–A Transfer Learning Technique. International Journal of Recent Technology and Engineering (IJRTE). 8 (6).
16. Rawat W, Wang Z. Deep convolutional neural networks for image classification: A comprehensive review. Neural computation. 2017 Sep;29(9):2352-449.
17. Sethy, P. K., Barpanda, N. K., Rath, A. K., & Behera, S. K. (2020). Deep feature based rice leaf disease identification using support vector machine. Computers and Electronics in Agriculture, 175, 105527.
18. Smith MR, Myers SS. Global health implications of nutrient changes in rice under high atmospheric carbon dioxide. GeoHealth. 2019 Jul;3(7):190-200.
19. Sulistyaningrum, D. R., Rasyida, A., & Setiyono, B. (2020, March). Rice disease classification based on leaf image using multilevel Support Vector Machine (SVM). In Journal of Physics: Conference Series (Vol. 1490, No. 1, p. 012053). IOP Publishing.
20. Syarief, M., & Setiawan, W. (2020). Convolutional neural network for maize leaf disease image classification. Telkomnika, 18(3), 1376-1381.
21. Waheed, A., Goyal, M., Gupta, D., Khanna, A., Hassanien, A. E., & Pandey, H. M. (2020). An optimized dense convolutional neural network model for disease recognition and classification in corn leaf. Computers and Electronics in Agriculture, 175, 105456.
22. Wang G, Sun Y, Wang J. Automatic image-based plant disease severity estimation using deep learning. Computational intelligence and neuroscience. 2017 Jul 5;2017.
23. Yakkundimath, R., Saunshi, G., Anami, B., & Palaiah, S. (2022). Classification of Rice Diseases using Convolutional Neural Network Models. Journal of The Institution of Engineers (India): Series B, 1-13.
24. Babu, S., Maravarman, M., & Pitchai, R. (2022). Detection of Rice Plant Disease Using Deep Learning Techniques. Journal of Mobile Multimedia, 757-770.

25. Daniya, T., & Vigneshwari, S. (2022). Amalgamation of Rice Plant Disease Classification and Detection with KNN and Butterfly Algorithm. In Innovations in Computer Science and Engineering (pp. 127-135). Springer, Singapore.
26. Strange, R. N., & Scott, P. R. (2005). Plant disease: a threat to global food security. Annu. Rev. Phytopathol., 43, 83-116.
27. Patil, R. R., & Kumar, S. (2022). Rice-Fusion: A Multimodality Data Fusion Framework for Rice Disease Diagnosis. IEEE Access.
28. Kathiresan, G., Anirudh, M., Nagharjun, M., & Karthik, R. (2021, May). Disease detection in rice leaves using transfer learning techniques. In Journal of Physics: Conference Series (Vol. 1911, No. 1, p. 012004). IOP Publishing.
29. Sowmyalakshmi, R., Jayasankar, T., PiIllai, V. A., Subramaniyan, K., Pustokhina, I., Pustokhin, D. A., & Shankar, K. (2021). An Optimal Classification Model for Rice Plant Disease Detection. CMC-Computers Materials & Continua, 68(2), 1751-1767.
30. Kamal, M. M., Masazhar, A. I., & Rahman, F. A. (2018). Classification of leaf disease from image processing technique. Indonesian Journal of Electrical Engineering and Computer Science, 10(1), 191-200.
31. Prajapati, H. B., Shah, J. P., & Dabhi, V. K. (2017). Detection and classification of rice plant diseases. Intelligent Decision Technologies, 11(3), 357-373.
32. Pinki, F. T., Khatun, N., & Islam, S. M. (2017, December). Content based paddy leaf disease recognition and remedy prediction using support vector machine. In 2017 20th International Conference of Computer and Information Technology (ICCIT) (pp. 1-5). IEEE.