ABSTRACT

On-time recognition and early control of the stresses in the paddy crops at the booting growth stage is the key to prevent qualitative and quantitative loss of agricultural yield. The conventional paddy crop stress recognition and classification activities invariably rely on human experts identifying visual symptoms as a means of categorization. This process is admittedly subjective and error-prone, which in turn can lead to incorrect actions being taken in stress management decisions. **Recognition and classification of yield affecting paddy crop stresses using field images** aims to design different deep convolutional neural network (DCNN) framework for automatic recognition and classification of various paddy crop stresses using the field images. Three different classifiers, the Convolutional Neural Network (CNN), pre-trained VGG-16, Mobilenet and Inception V3 models have been deployed to distinguish biotic stresses such as bacterial leaf blight, fungal blast and brown spot. The average stress classification accuracies of 89.12%, 84.44% and 76.34% have been achieved using the CNN, VGG-16, Mobilenet and Inception V3 classifiers, respectively.

INTRODUCTION

The paddy crop cultivation is an integral part of Indian agricultural economy. The paddy crops are exposed to various types of stresses, both biotic and abiotic as they grow in a wide range of environments characterized by different temperatures, climates, and soil-water conditions, whose combined effect can adversely affect crop performance and survival. The on time monitoring and recognition of these stresses, the supply of adequate farm inputs and rapid morphological diagnoses can reduce the adverse effects of stresses on the crops. The paddy crops under stress show obvious symptoms in the colour, shape and texture of the leaves. It is difficult to capture and quantify these micro symptoms using manual visual observation. The leaf injury and leaf colour variation are the two visual scoring symptoms used in the classification and determination of the stress categories.

OBJECTIVE

The objective of the proposed system are:

- To design different deep convolutional neural network classifiers for automatic recognition and classification of common biotic stresses such as bacterial leaf blight, fungal blast and brown spot.
- Comparing the results obtained from different classifiers(CNN, VGG-16, Mobilenet, Inception V3).
- To suggest remedies for the stresses.

LITERATURE SURVEY

Sambuddha et al. (2018) have proposed a deep machine vision based approach for identification, classification, and quantification of eight soybean plant stresses, including both biotic and abiotic using 25,000 images of stressed and healthy leaflets in the fields. The overall classification accuracy of 94.13% is achieved by the developed deep CNN model. Konstantinos (2018) has developed five different deep CNN models which include AlexNet, AlexNetOWTBn, GoogLeNet, Overfeat and VGG for the identification of plant disease combinations. The image dataset comprises 87,848 individual leaf images from58 different

using the simple leaves images of healthy and diseased plants classes of plant-disease combinations of 25 different plant species. The highest classification accuracy of 99.53% is achieved using the VGG CNN model. Yang Lu et al. (2017) have presented a paddy crop disease identification method using deep learning techniques. The image dataset
comprises 500 images of healthy and diseased rice leaves and stems. The developed CNN model is inspired by LeNet-5 and AlexNet CNN architectures. The work has considered 10 common paddy crop diseases and an average recognition accuracy of 95.48% is obtained using the trained CNN model. The work has also shown that the stochastic pooling enhances the generalization ability of the CNN model and prevents over-fitting.

The literature survey has revealed that the researchers have extensively used the deep learning techniques for plant or crop disease recognition and classification. In all the works, individual plant leaf images have been used for the disease or stress recognition and classification. But, the diseases or stresses can occur in all parts of the plants apart from the leaves. So far, there has not been any comprehensive study and referable results on recognition and classification of paddy crop biotic stresses (bacterial leaf blight, fungal blast and brown spot using field images and using Inception V3 architecture as a classifier along with CNN, VGG-16 and MobileNet architectures.

METHODOLOGY

To deal with the challenges in the detection of paddy crop stresses, a different deep learning-based approach has been introduced to recognize and classify the paddy crop stresses. Identifying disease from the images of the rice plant is one of the interesting research areas in computer and agriculture field. For that different image processing and machine-learning techniques are used in the automatic recognition and classification of stresses affecting paddy crops based on images of stressed paddy crops. There are different biotic and abiotic stresses which affects during the booting stages of paddy crop. Three major biotic stresses which affects the paddy crop are:

Bacterial leaf blight:

Fungal blast:

Brown spot:

KEY TERMS

Image Acquisition: An image database specifically for paddy crop stresses is available in Kaggle. We need to find more images from different sources.

Image Preprocessing: For getting better results in further steps, image pre-processing is required because dust, dewdrops, insect’s excrements may be present on the plant; these things are considered as image noise. Furthermore, captured images may have distortion of some water drops and shadow effect, which could create problems in the segmentation and feature extraction stages. Effect of such distortion can be weakened or removed using different noise removal filters. They may below contrast in captured images

Bacterial leaf blight   Leaf Blast   Neck Rot

Figure: Different types of Paddy crop stresses

Image Segmentation: This can play an important role in plant disease detection. This process is to divide the image into particular regions or objects. The main aim of segmentation is to analyse the image data so one can extract the useful features from the data. There are two ways to carry out the image segmentation: (1) based on discontinuities and (2) based on similarities.

Feature Extraction: The feature extraction technique of image analysis focuses on identifying inherent characteristics or features of objects present within an image. These features can be used to describe the object. Generally, features are classified in to three categories are extracted: colour, shape, and texture. The colour is an important feature because it can differentiate one disease from another.

Classification: It classify the data into specific groups or classes. Classification is usually called as supervised learning approach. Classification is a two-step process: First the classifier model is generated which describes predefined set of classes
Clustering: It is a process of grouping data into different groups based on the similarity of the data. It means the data points with the similar objects are grouped into one group and dissimilar objects are grouped into another group.

PROPOSED METHODOLOGY

In the proposed system, we intend to recognise and classify three stresses affecting paddy crops, namely, bacterial leaf blight, fungal blast and brown spot. On the basis of the affected stress we also suggest remedies to overcome the stresses. The proposed methodology consists of two main stages, namely, image dataset preparation and deep learning-based classification.

Image Acquisition: The system considers the impact of 3 biotic stress classes on the paddy crops. We have collected images of paddy crops from various resources like Kaggle and many other resources. We have prepared a total of 800 images in our database, considering 200 images per stress class are acquired. A total of 200 healthy field images per paddy crop variety along with the stressed paddy field images have been included in the dataset.

Image preprocessing: RGB color image of the diseased rice plant is collected and resize the images by using 32x32 filters. Filter the images by removing background noise. Each image is labeled with its corresponding disease name.

Feature Extraction: The training dataset consist of images of the stressed and healthy paddy crops. The input layer takes an image in the size of (224 x 224 x 3), and the output layer is a softmax prediction on 1000 classes. From the input layer to the last max pooling layer (labeled by 7 x 7 x 512) is regarded as **the feature extraction part** of the model.

Rectified Linear Unit (ReLU): The activation function 'ReLu' has been considered in all the hidden Layers. It influences the learning ability of the neural network model. It changes the negative activation value to zero. That is all the pixels with a negative value will be replaced by zero.(x) = max(0,x).The ReLU activation formula is shown in equation()

$$y(i; j;d) = \max[0;x(i; j;d)l] \qquad (3.1)$$

with ReLU layer has no parameter inside, hence no need for parameter learning in this layer. The purpose of ReLU is to increase the non-linearity of CNN. Since the semantic information in an image is a highly non-linear mapping of pixel values in the input, we want the mapping from CNN input to its output also be highly non-linear.

Max-pooling Layer: The spatial size is preserved after convolution using hyper parameter padding 1. The spatial pooling is carried out by using five max pooling layers with the pooling size 2 × 2 pixels. The purpose of the pooling is to reduce the dimensionality of the input image to prevent over-fitting and improve computation speed. In max pooling, the pooling operator maps a sub-region to its maximum value. This layer performs nonlinear down-sampling on the input given and it reduces the size of the input feature maps but preserves the important feature. The function of the layer is to improve the generalization and to produce faster convergence. When the input feature map passes through this layer, max operation is applied which outputs a maximum among the input as shown in equation ()

$$Sj = max(i\ 2\ Rj)ai \qquad (3.2)$$

Where Rj represents pooling region j; S denotes the output pooled
Feature maps.

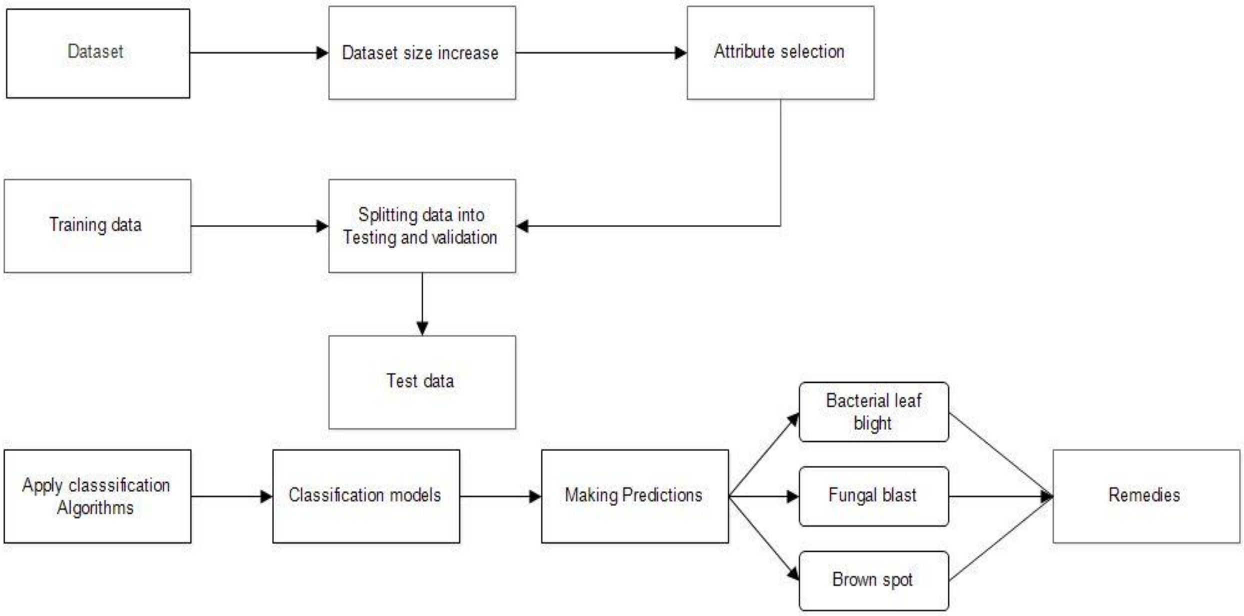
Softmax Regression: It is a form of logistic regression that normalizes an input value into vectors that follows a probability distribution whose total sums up to 1. The output values are between the range [0,1] which is nice because we can avoid binary classification and accommodate as many classes or dimensions in our neural network model. This is why softmax is also known as a multinomial logistic regression. The regression function is usually used to compute losses. As the name suggests, in softmax regression (SMR), we replace the sigmoid logistic function by the softmax function

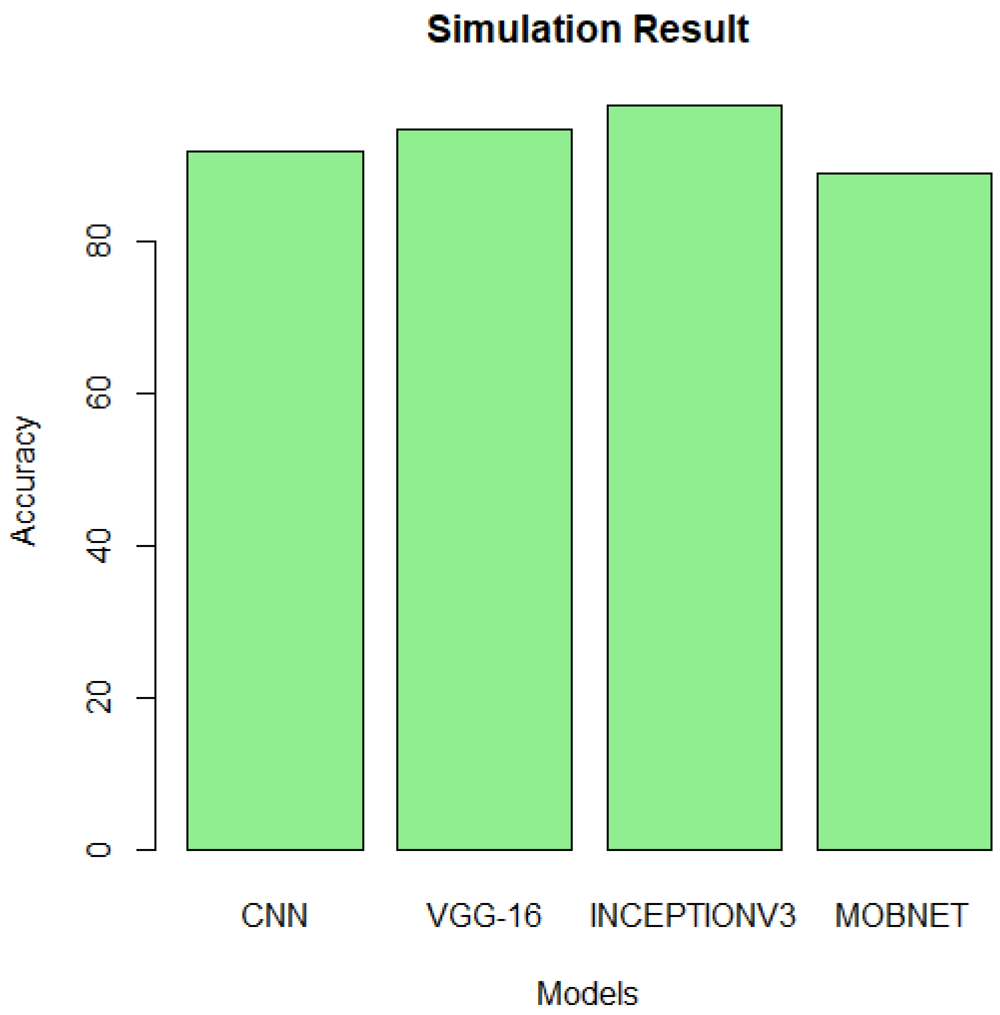
$$P(y = jz((i))) = s\ o\ f\ tmax(z((i))) = e(z((i)))=((\ j = 0)ke(z(k(i)))) \quad (3.3)$$

where we define the net input z as

$$z = w0x0+w1x1+:::+wmxm =(\ l = 0)mwlxl = wT\ x \quad (3.4)$$

x is the feature vector of 1 training sample, w is the weight vector, and w0 is the bias unit. Now, this softmax function computes the probability that this training sample x(i) belongs to class j given the weight and net input z(i).Thus we compute the probability p(y = j—x(i); wj) for each class label in j =1, ..., k. Note the normalization term in the denominator which causes these class probabilities to sum up to one.




Simulation Result
Accuracy
0
20
40
60
80
CNN
VGG-16
INCEPTIONV3
MOBNET
Models

```
Epoch 1/100

785/785 [==============================] - 12804s 16s/step -
loss: 0.3009 - accuracy: 0.9312 - val_loss: 1.5539 -
val_accuracy: 0.9665

Epoch 2/100

785/785 [==============================] - 473s 602ms/step -
loss: 0.1368 - accuracy: 0.9535 - val_loss: 1.5516 -
val_accuracy: 0.9617

Epoch 3/100

785/785 [==============================] - 478s 609ms/step -
loss: 0.1085 - accuracy: 0.9605 - val_loss: 1.3914 -
val_accuracy: 0.9649

Epoch 4/100

785/785 [==============================] - 478s 608ms/step -
loss: 0.0911 - accuracy: 0.9657 - val_loss: 1.2942 -
val_accuracy: 0.9649
```

```
Epoch 1/100

785/785 [==============================] - 12804s 16s/step -
loss: 0.3009 - accuracy: 0.9312 - val_loss: 1.5539 -
val_accuracy: 0.9445

Epoch 2/100

785/785 [==============================] - 473s 602ms/step -
loss: 0.1368 - accuracy: 0.9535 - val_loss: 1.5516 -
val_accuracy: 0.9417

Epoch 3/100

785/785 [==============================] - 478s 609ms/step -
loss: 0.1085 - accuracy: 0.9505 - val_loss: 1.3914 -
val_accuracy: 0.9549

Epoch 00003: early stopping
```