Volume: 10 Issue: VIII Month of publication: August 2022

DOI: https://doi.org/10.22214/ijraset.2022.46554

Smart Mobile System for Detection and Classification of the (Oryza Sativa) Rice Plant Disease Using Deep Learning and Transfer Learning

Vedanshee Agarwal[1], Mohan Kshirsagar[2], Reetu Jain[3], Shekhar Jain[4]
[1]per 1st Affiliation: The British School, New Delhi, India
[2]Completed BE in ENTC from SIT Lonavala, Having 4 Years of experience in the field of Robotics and Artificial Intelligence.
[3]Chief mentor, Founder and MD of On My Own Technology Private Limited Mumbai.
[4]Co-founder and CEO of On My Own Technology Private Limited Mumbai.

Abstract: In this research, we are analyzing and performing the classification of five different types of bacterial and fungal diseases (Bacterial Leaf Blight, Brown Spot, Rice Blast, Hispa, Healthy Leaf Sheath) which occur in the Oryza Sativa (Rice)plantation.

Early Identification and classification of the diseases will reduce 37% quantitative loss of rice plantations. Manual visual inspection has limitations of human vision, and by the time a disease is suspected by the farmer, chances are it's too late to take action. This creates a need for a smart system (mobile app), which will help in the accurate and timely detection of diseases and surmount the boundaries of human vision.

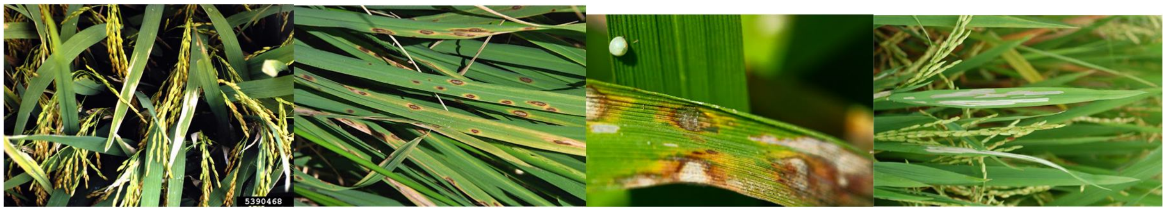
Our app is convenient for farmers as with one-click it will give you a classification of the disease using a Deep learning model running at the back end. Additionally, it will display the ideal quantity of pesticide/ remedy to save crops from the fungal disease as well as soil from being destroyed.

The algorithm created using Deep Learning (GoogLe Net) detects and classifies the five different rice sheaths, detecting features such as its shape, color and texture. We are utilizing GoogLeNet as a transfer learning approach to exploit the pre-train DCNN (Deep Convolutional Neural Network). To monitor how the DCNN differentiates between the five different sheaths we use the Grad-CAM technique. This classification system utilized the techniques of deep learning and excelled in the existing tedious image processing methods, giving an overall test accuracy of 80% for the five different leaf sheaths.

Keywords: Oryza Sativa; GoogLe Net; disease; application; rice; detection; classification.

I. INTRODUCTION

An increase in consumption of Oryza sativa (rice) has led farmers across Asian nations to farm Oryza sativa, with a result of a 2.5 percent growth in the advancement of the crop every year. With these increased percentages in Oryza sativa cultivation, farmers are facing new difficulties in terms of disease. Due to a lack of knowledge of various types of bacterial and fungal infection rice plant faces, during the harvest cycle, farmers over the globe lose 37% of the production. [6]



[Fig: A] Diseased Rice Leaf; Bacterial Leaf Blight, Rice Hispa, Brown Spot, Rice Blast

Rice (Oryza Sativa)[1] is one of the essential crops in India and Asia, losses due to the diseases severely affect the earnings. Over 90 percent of the world's rice is produced and consumed in Asia. Besides Asia, rice is grown in Africa, Latin America, the United States, and Australia.

In the European Union, the extent of production is limited. Identification of the infection is the key to counter the qualitative and quantitative loss of agricultural yields. The development of a deep learning-based system for identifying and classifying diverse diseases of the infected plants is an emerging research area in precision or smart farming using mobile gadgets.

Manual visual inspection has limitations of human vision, and it is a time-consuming process to capture the disease. The objective here is to identify four primary infections on rice leaf, Bacterial Leaf Blight, Brown Spots, Rice Blast, and Hispa, along with the healthy leaf sheath, to classify using GoogLeNet. GoogLeNet is an Advanced Deep Convolutional Neural Network (DCNN). We are utilizing GoogLeNet as a transfer learning approach to exploit the pre-train DCNN.

In the color images of the coat, there is a high similarity between the different infected leaf sheath and their symptoms, which increases the difficulty of the detection and diagnosis. A reliable AI-enabled automated system for rice disease classification is essential for early detection of fungal and bacterial infection to save efforts, particularly judicious use of pesticides and the time required for individual inspection. This classification system utilized the techniques of deep learning and excelling in the existing tedious image processing methods.

II. WORKING PRINCIPLE

We have created an algorithm that identifies four primary infections on rice leaf, Bacterial Leaf Blight, Brown Spots, Rice Blast, and Hispa, along with the healthy leaf sheath, to classify using GoogLeNet (GoogLeNet is a Deep Convolutional Neural Network).

To start off, we first collected data sets for our 5 different rice classes, then after collecting the data, we resized it to 300 by 300 (pixels).[2]

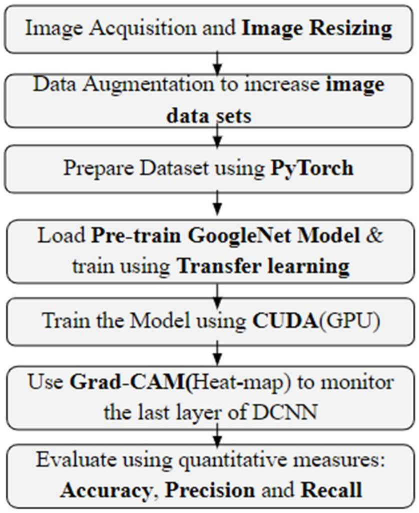


Image Acquisition and **Image Resizing**
Data Augmentation to increase **image data sets**
Prepare Dataset using **PyTorch**
Load **Pre-train GoogleNet Model** & train using **Transfer learning**
Train the Model using **CUDA**(GPU)
Use **Grad-CAM**(Heat-map) to monitor the last layer of DCNN
Evaluate using quantitative measures: **Accuracy**, **Precision** and **Recall**

[Fig: B] working flow chart of algorithm

Like shown in Fig B we used data augmentation and rotated each image from 0 - 270 degrees with a constant step angle creating maximum data sets. After that, we divide the data set into Test, Train and Validation sets. We train the Googlenet model using a train set and validation set using transfer learning. To train the model we use a GPU which is a High speed processing tool After training the model, we check the accuracy. The grad cam is the heat map technique to assist the accuracy value as well as to help the detection.

III. TECHNOLOGY

A. Machine Learning

The GoogLeNet (Inception V1) is the employed DCNN architecture in the proposed five types of rice sheath classification model using Fine-tuning the ConvNet approach of transfer learning. This architecture consists of 22 deep layers. It reduces the number of parameters from 60 million (AlexNet) to 4 million, faster than AlexNet, and much more accurate. It uses many kinds of methods of 1×1 convolution and global average pooling that enables it to create deep architecture. The architecture also has the 9 Inception layers followed by Auxiliary Classifiers. The dimension of the input layer cannot exceed the predicted width (W) and height (H) equal to, 224×224 where depth (D) of this layer is 3.

Applying the data augmentation process, we train GoogLeNet. As we are going to use GoogLeNet in image classification with the five types of rice leaf sheath dataset, Network architecture should result in 5 output class labels. For this purpose, we need to update the network because GoogLeNet is pre-train for 1000 classes, the final classifier label that is Linear is having the 1000 nodes at the output layer. This last linear layer is code only for five nodes along with fine-tuning the weights of the pre-trained network by continuing the backpropagation.[3]

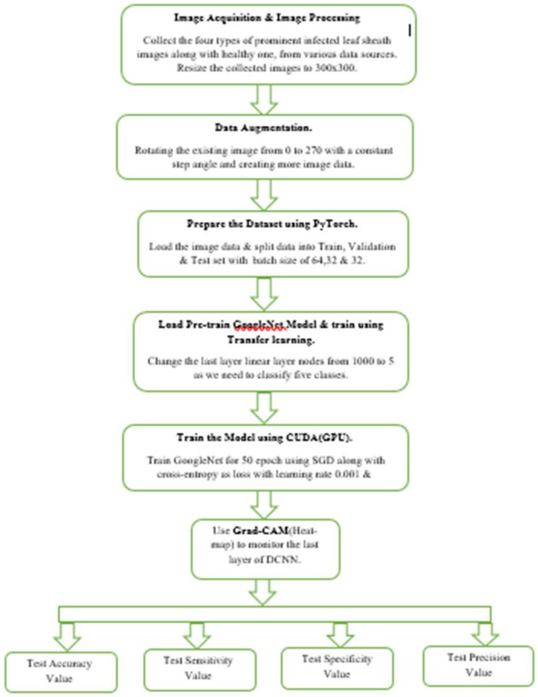


Image Acquisition & Image Processing
Collect the four types of prominent infected leaf sheath images along with healthy one, from various data sources. Resize the collected images to 300x300.
Data Augmentation.
Rotating the existing image from 0 to 270 with a constant step angle and creating more image data.
Prepare the Dataset using PyTorch.
Load the image data & split data into Train, Validation & Test set with batch size of 64,32 & 32.
Load Pre-train GoogleNet Model & train using Transfer learning.
Change the last layer linear layer nodes from 1000 to 5 as we need to classify five classes.
Train the Model using CUDA(GPU).
Train GoogleNet for 50 epoch using SGD along with cross-entropy as loss with learning rate 0.001 &
Use Grad-CAM(Heat-map) to monitor the last layer of DCNN.
Test Accuracy Value
Test Sensitivity Value
Test Specificity Value
Test Precision Value

[Fig C] data analysis

The weights are updated by implementing stochastic gradient descent (SGD) algorithm as an optimizer based on the dataset images along with cross-entropy will be used for the loss. At the last layer of DCNN, we are implementing Grad-CAM to get a visualization in terms of heat-map.

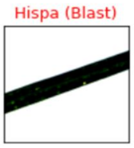
To train a DCNN model required a large image dataset. Unfortunately, no available dataset of a leaf sheath with more than thousands of labeled images. The method of transfer learning is the perfect solution for this challenging problem, where big size image dataset such as the ImageNet used in pre-train GoogLeNet. From different resources, we get image data mentioned beneath in reference. For example, Class Bacterial Leaf Blight, we have 207 images, Class Rice Blast we have 922 images, Class Brown Spot, we have 716 image files, Class Hispa, we have 580 image files, and class Healthy, we have 1488 image files. [5]

Rotating the existing image from 0 to 270 with a constant step angle and creating more image data is a good trick which we exercised here. Thus, for Class Bacterial Leaf Blight, we chose this set of angles {0,90,180,270}, 207x4, we get 828 images.

To train GoogLeNet, Using PyTorch, the dataset is randomly divided into training, validation, and testing where 70% of the dataset is for training,20% for validation, and 10% for testing. The batch size for training data is 64, validation and testing batch size is 32. SGD with 50 epochs and a learning rate of 0.001 is used to train the model. Unseen test data images are utilized by the trained model to get the classification accuracy.

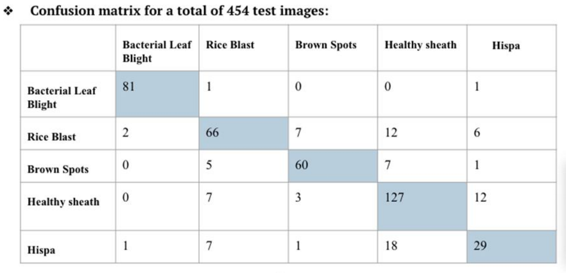
The whole thing is executed on the Kaggle Kernel, which has GPU NVIDIA P100 along with 16GB GPU memory with GoogLeNet (Inception V1) model provided by PyTorch 1.6.0.

The following flow chart shows the flow of the technical operation and how the result and conclusions are drawn. [7]



Following are Grad-CAM (**Heat Map)** results for **Bacterial Leaf Blight classification.**

[Fig D] results for confusion matrix

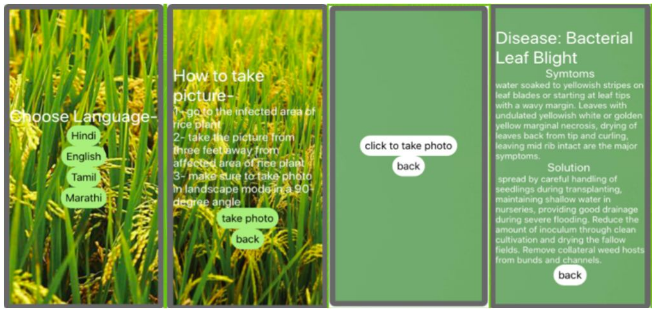


❖ **Confusion matrix for a total of 454 test images:**

	Bacterial Leaf Blight	**Rice Blast**	**Brown Spots**	**Healthy sheath**	**Hispa**
Bacterial Leaf Blight	81	1	0	0	1
Rice Blast	2	66	7	12	6
Brown Spots	0	5	60	7	1
Healthy sheath	0	7	3	127	12
Hispa	1	7	1	18	29

[Fig E] Data analysis and results

B. Mobile Application



Choose Language-

Hindi
English
Tamil
Marathi

How to take picture-
1- go to the infected area of rice plant
2- take the picture from three feet away from affected area of rice plant
3- make sure to take photo in landscape mode in a 90-degree angle

take photo
back

click to take photo
back

Disease: Bacterial Leaf Blight

Symtoms

water soaked to yellowish stripes on leaf blades or starting at leaf tips with a wavy margin. Leaves with undulated yellowish white or golden yellow marginal necrosis, drying of leaves back from tip and curling, leaving mid rib intact are the major symptoms.

Solution

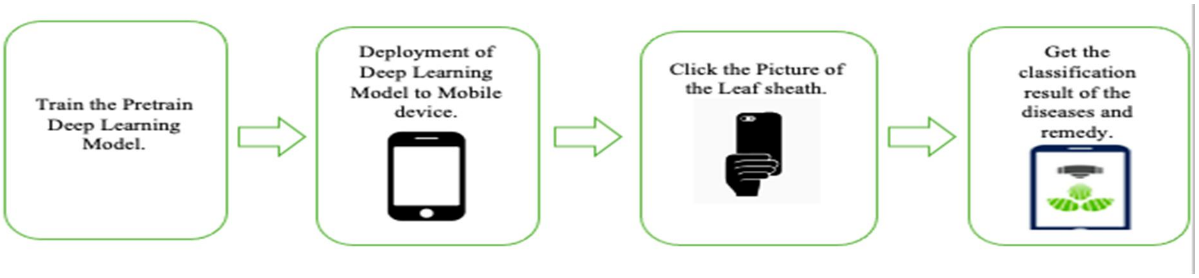
spread by careful handling of seedlings during transplanting, maintaining shallow water in nurseries, providing good drainage during severe flooding. Reduce the amount of inoculum through clean cultivation and drying the fallow fields. Remove collateral weed hosts from bunds and channels.

back

[Fig F] Mobile Application

In this app the farmer can choose their preferred language, and then with one click take a picture of the rice plant where they assume a disease may be forming. Just seconds after doing this, our algorithm will lead the farmer to a screen where they can find out if their plant has or doesn't have a disease. If there is a disease the farmer will be led to a screen where they can find out the remedies for it. By doing so, we are giving the farmer the biggest advantage to taking action in the early stage to take all precautions to save the rice plantation.

C. Working Flow chart of Application



Train the Pretrain Deep Learning Model.

Deployment of Deep Learning Model to Mobile device.

Click the Picture of the Leaf sheath.

Get the classification result of the diseases and remedy.

[Fig G] Working Flow Chart

The whole deep learning algorithm can be used at the backend of the mobile application, and the front end of the mobile application will give the classification result of the disease along with symptoms and diagnosis. Here we just made the prototype of the app in MIT app inventor, where we have an imitation of the deployment of the pre-train deep learning model on the mobile application made by the MIT app inventor. The problem is addressed next i.e., Future Prospects. Now, the Following is the flow in the application.

IV. METHODOLOGY

The achievement of the proposed model has validated using four quantitative measures: Accuracy, sensitivity, specificity, and precision. These measures are computed using the following forms:[2]

$$\text{Accuracy (ACC)} = \frac{t_p+t_n}{t_p+f_p+f_n+t_n}$$

$$\text{Sensitivity (TPR)} = \frac{t_p}{t_p+f_n}$$

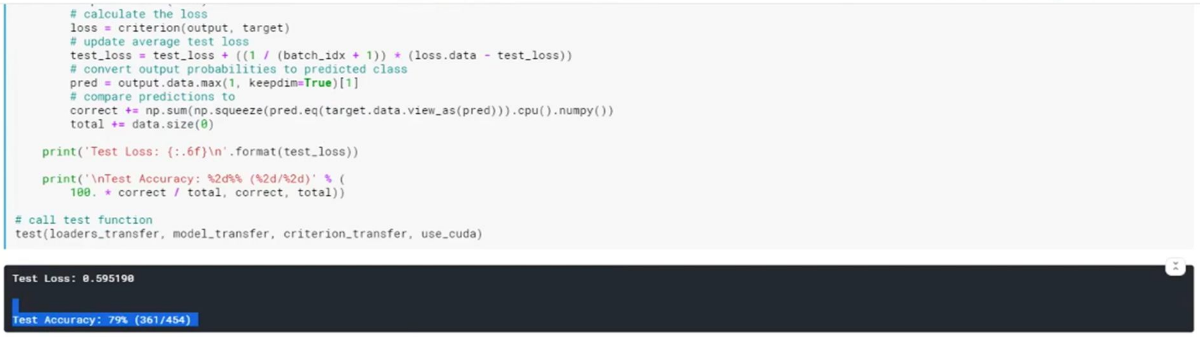
$$\text{Specificity (TNR)}= \frac{t_n}{f_n+t_n}$$

$$\text{Precision (PPV)} = \frac{t_p}{t_p+f_p}$$

Tp = True Positive
Tn = True Negative
Fp = False Positive
Fn = False Negative

V. RESULTS

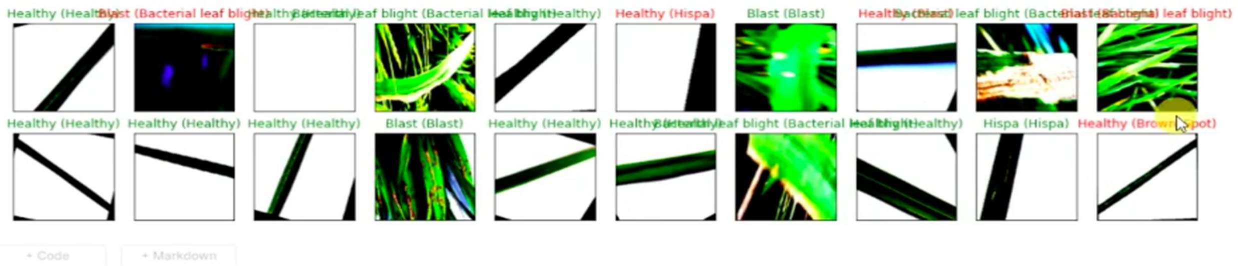
The overall accuracy achieved by our model is 79%. The individual accuracy per class is as follows:



```
        # calculate the loss
        loss = criterion(output, target)
        # update average test loss
        test_loss = test_loss + ((1 / (batch_idx + 1)) * (loss.data - test_loss))
        # convert output probabilities to predicted class
        pred = output.data.max(1, keepdim=True)[1]
        # compare predictions to
        correct += np.sum(np.squeeze(pred.eq(target.data.view_as(pred))).cpu().numpy())
        total += data.size(0)

    print('Test Loss: {:.6f}\n'.format(test_loss))

    print('\nTest Accuracy: %2d%% (%2d/%2d)' % (
        100. * correct / total, correct, total))

# call test function
test(loaders_transfer, model_transfer, criterion_transfer, use_cuda)
```

```
Test Loss: 0.595190


Test Accuracy: 79% (361/454)
```

Bacterial leaf blight: 95%
Rice Blast: 75%
Brown Spot :80%
Hispa: 40%
Healthy: 94%



The proposed method is to beat the present complex image processing methods. The Deep Learning approach is successfully employed to get the results. The number of data images for Class Bacterial Leaf Blight was less. To surmount this problem and to obtain more data images, an augmentation process is applied. To exploit the advantage of Deep Convolutional Neural Network (DCCN), GoogLeNet is trained by using the Transfer learning method. [4]

We are optimistic that we can perform this classification using a smartphone or any other mobile device as the GoogLeNet is more reliable on these devices. The GoogLeNet is suitable for low processing power devices. Along with this, we can use the Grad-CAM procedure to monitor the features of the last convolution layer of the GoogLeNet. The data augmentation and transfer learning using GoogLeNet along with Grad-CAM is making an overall stable system for the precision farming of (Oryza Sativa) rice fields. The concept of smart farming using IoT devices can be a practical reality, where we can use cloud services to help the farmers to tackle this disease.

VI. CONCLUSION

The farmers across India lose 37% of their annual rice plantation and by adapting this methodology 79% of it can be saved. Our algorithm can identify four primary infections on rice leaf, Bacterial Leaf Blight, Brown Spots, Rice Blast, and Hispa, along with the healthy leaf sheath, to classify we use GoogLeNet (GoogLeNet is a Deep Convolutional Neural Network). In our results, we conclude that Deep learning technique and the grad cam is nicely employed to get correct classification of the rice plant diseases along with the healthy leaf sheath. By using data augmentation to get higher results we are receiving a final classification test accuracy of 79%. In future scope, The Grad-Cam technique to get the heatmap at the last layer of the DCNN which is currently on paper, can be possible to implement in PyTorch with more datasets which can achieve higher accuracy.

VII. ACKNOWLEDGMENT

I would like to take this time to thank On My Own Technology, and its mentors for guiding me throughout this project, and always being there for me when I was stuck and helping my problem shoot. Additionally, I would thank my parents for encouraging me and being a support system. Finally, I would like to thank IRIS, as well as IIT Tech Fest for recognizing my project.

REFERENCES

[1] Goluguri, N. V. Raja Reddy. "Image Classifiers and Image Deep Learning Classifiers Evolved in Detection of Oryza Sativa Diseases: Survey." Artificial Intelligence Review, 28 May 2020, https://link.springer.com/article/10.1007/s10462-020-09849-y.

[2] Prajapati, H. B. (2015, November 16). "Detection and classification of rice plant diseases". Retrieved from https://www.researchgate.net/publication/318437440_Detection_and_classification_of_rice_plant_diseases

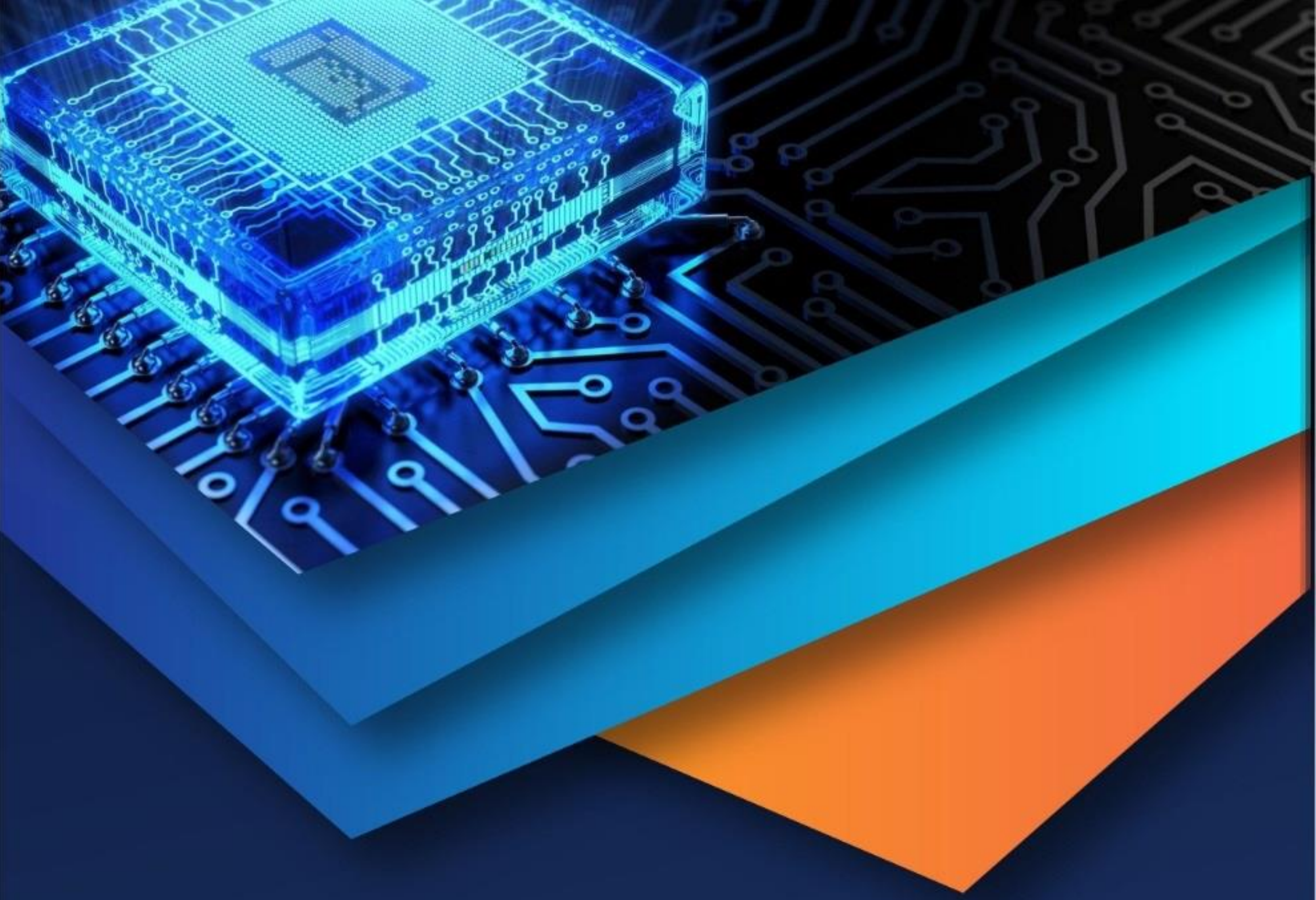
[3] B.D. Jadhav, Amrita A. Joshi. "Monitoring and Controlling Rice Diseases Using Image Processing Techniques." Institute of Electrical and Electronics Engineers, 1 May 2017, https://ieeexplore.ieee.org/document/791501

[4] Pests & diseases. (2018, April 23). Retrieved from https://plantix.net/en/library/plant-diseases

[5] Dept. of Agriculture, G. of K. (n.d.). DISEASE:BACTERIAL LEAF STREAK. Retrieved from http://www.cpsskerala.in/OPC/pages/riceDiseasebacleafstreak.jsp

[6] Kaggle. (n.d.). Rice Leaf Diseases Dataset. Retrieved from https://www.kaggle.com/vbookshelf/rice-leaf-diseases

[7] Riaz Khan. (n.d.). Analysis of ResNet and GoogleNet models for malware detection. Retrieved from https://www.researchgate.net/publication/328927734_Analysis_of_ResNet_and_GoogleNet_models_for_malware_detection



10.22214/IJRASET

INDEX COPERNICUS

45.98

IMPACT FACTOR:
7.129

IMPACT FACTOR:
7.429

INTERNATIONAL JOURNAL
FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY