A Convolutional Neural Network Model for Prediction of Paddy Crop Disease

BY

Md. Ahasanul Kobir Opy

ID: 191-15-2427

Md. Shahin Alom

ID: 191-15-2595

This Report Presented in Partial Fulfillment of the Requirements for the Degree of Bachelor of Science in Computer Science and Engineering

Supervised By

Shayla Sharmin
Senior Lecturer
Department of Computer Science & Engineering
Daffodil International University

Co-Supervised By

Ms. Tania Khatun
Assistant Professor
Department of Computer Science & Engineering
Daffodil International University

DAFFODIL INTERNATIONAL UNIVERSITY

DHAKA, BANGLADESH

February 06, 2023

APPROVAL

This Project/internship titled **"A Convolutional Neural Network Model for Prediction of Paddy Crop Disease"**, submitted by Name: Md. Ahasanul Kobir Opy, ID No: 191-15-2427 & Name: Md. Shahin Alom, ID No: 191-15-2595 to the Department of Computer Science and Engineering, Daffodil International University has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of B.Sc. in Computer Science and Engineering and approved as to its style and contents. The presentation has been held on ***February 06, 2023.***

BOARD OF EXAMINERS

8/2/23

Dr. Touhid Bhuiyan **Chairman**
Professor and Head
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Internal Examiner

Arif Mahmud
Assistant Professor
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Internal Examiner

Mr. Mushfiqur Rahman
Senior Lecturer
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

External Examiner

Dr. Mohammad Shorif Uddin
Professor
Department of Computer Science and Engineering
Jahangirnagar University

DECLARATION

We and hence affirm that we engaged on this initiative at **Daffodil International University** under the direction of **Shayla Sharmin, Senior Lecturer** in the **Department of Computer Science & Engineering.** We likewise pledge that neither the proposed system nor any isolated constituent has ever been submitted to another institution for cognizance for a degree or diploma.

Supervised by:

Shayla Sharmin 8.2.23

Shayla Sharmin

Senior Lecturer

Department of Computer Science & Engineering

Daffodil International University

Co- Supervised by:

MS. Tania Khatun

Assistant Professor

Department of Computer Science & Engineering

Daffodil International University

Submitted by:

Opy

Md. Ahasanul Kobir Opy

ID: 191-15-2427

Department of Computer Science & Engineering

Daffodil International University

Md. Shahin Alom

ID: 191-15-2595

Department of Computer Science & Engineering

Daffodil International University

ACKNOWLEDGEMENT

Initially, we would like to exhibit our deepest gratitude to Almighty for providing us the opportunity to successfully finish the final project via His divinity.

We thank **Shayla Sharmin, Senior Lecturer,** Department of Computer Science & Engineering, and express our sincere gratitude. Our decision to pursue this project was inspired by our supervisor's immense acumen of and genuine intrigue in the subject of machine learning. This endeavor was afforded by his inexhaustible indulgence, academic leadership, persistent affirmation, persistent and proactive oversight, helpful advice, insightful advice, reading numerous mediocre versions, and fixing them at all levels.

We would like to express our gratitude to **Professor Dr. Touhid Bhuiyan,** Head of the Department of CSE, as well as to the other faculty member and employees of the CSE department at Daffodil International University, for their gracious consultation in wrapping up our research.

We desire to thank everyone who was involved in this deliberation while doing the courses at Daffodil International University.

Lastly, we must humbly salute our parents' unflinching cooperation and endurance.

ABSTRACT

Rice is a dietary staple all through Indian Subcontinent, notably in Bangladesh. The staple food consumed by 130 million people in Bangladesh is rice. To feed its 130 million people, Bangladesh currently produces 25 million tons of rice. If massive amounts of rice weren't thrown away every year due to disorganization, the number may be higher. Bangladesh is an evolving nation, hence the dropout index hasn't dropped slightly, whereas. The majority of cultivators are ignorant of the various types of rice diseases. Therefore, it is genuinely intriguing research to learn more about the condition using the resource of afflicted needles in farming activities. This study report consists of a pilot for the identification and classification of Paddy sickness utilizing inflamed leaves and algorithmic learning devices. We must not overlook the three rice ailments known as Blast disease, Plant Hopper ailment, and Leaf Folder condition. Right here, we've adopted a category-based deep learning version rooted in CNN. Notably, we utilized a diversity of switch mastering versions for classification. The meticulous pre-Mannering of the photograph is the most crucial aspect of this study. After conducting numerical inference, we mentored our strategy and evaluated it utilizing the dataset. In the end, we examined other approaches, but CNN produced handy findings for our dataset, with an exactness of roughly 99.89%.

TABLE OF CONTENTS

CONTENTS	PAGE
Title	i
Approval	ii
Board of examiners	ii
Declaration	iii
Acknowledgements	iv
Abstract	v
List of Figures	x-xi
List of Tables	xii
CHAPTER 1: INTRODUCTION	**1-4**
1.1 Introduction	1
1.2 Motivation	1-2
1.3 Rationale of the Study	2
1.4 Research Questions	2
1.5 Expected Output	3
1.6 Project Management and Finance	3
1.7 Report Layout	3-4
CHAPTER 2: BACKGROUND	**5-9**
2.1 Preliminaries	5
2.2 Related Works	5-7
2.3 Comparative Analysis and Summary	7-8
2.4 Scope of the Problem	8
2.5 Challenges	9
CHAPTER 3: RESEARCH METHODOLOGY	**10-23**
3.1 Research Subject and Instrumentation	10
3.2 Data Collection Procedure	10-11

3.3 Statistical Analysis 11-12

3.3.1 Sample Images 13-17

3.3.1.1 Magnaporthe Grisia (Blast Disease) 13-14

3.3.1.2 Cnaphalocrocis medinalis (Leafroller Disease) 15-16

3.3.1.3 Fulgoromorpha (Plant Hopper Disease) 17

3.4 Applied Mechanism 18

3.4.1 CNN Model Configuration 18

3.4.1.1 CNN Model 19

3.4.1.2 Architecture of Our Model 20

3.4.2 Transfer Learning Model 20

3.4.2.1 Importing VGG16 20

3.4.2.2 Importing Densenet201 21

3.4.2.3 Importing Xception 21

3.5 Implementation Requirements 22

3.5.1 Image Resizing 22

3.5.2 Normalization 22

3.5.3 Splitting of the dataset 23

3.5.4 Featuring Data and Labeling the class 23

CHAPTER 4: EXPERIMENTAL RESULTS AND DISCUSSION 24-43

4.1 Experimental Setup 24

4.1.1 VGG16 24

4.1.1.1 Architecture of VGG16 24-25

4.1.1.2 Model Summary of VGG16 25

4.1.1.3 Model Performance of VGG16 26

4.1.1.4 Visualization of VGG16 26

4.1.1.5 Model Output of VGG16 27

4.1.3 ResNet50 32

4.1.3.1 Architecture of Densenet201 32

4.1.3.2 Model Summary of Densenet201 33

4.1.3.3 Model Performance of Densenet201 33

4.1.3.4 Visualization of Densenet201 34

4.1.3.5 Model Output of Densenet201 35

4.1.4 Xception 39

4.1.4.1 Architecture of Xception 39

4.1.4.2 Model Summary of Xception 40

4.1.4.3 Model Performance of Xception 40

4.1.4.4 Visualization of Xception 41

4.1.4.5 Model Output of Xception 42

4.1 Experimental Results and Analysis 43

4.2 Discussion 43

CHAPTER 5: IMPACT ON SOCIETY, ENVIRONMENT AND SUSTAINABLITY 44-45

5.1 Impact on Society 44

5.2 Impact on Environment 44

5.3 Ethical Aspects 44

5.4 Sustainability Plan 45

CHAPTER 6: SUMMARY, CONCLUSION, RECOMMENDATION AND IMPLICATION FOR FUTURE RESEARCH 45-46

6.1 Summary of study 45

6.2 Conclusion 46

6.3 Implementation for Further Study 46

REFERENCES 47-48

LIST OF FIGURES

FIGURES	PAGE NO
Figure 3.3.1: Pie chart of total data.	11
Figure 3.3.2: A Bar chart of total data division	12
Figure 3.3.1.1: Sample Images of Magnaporthe grisea (Blast disease)	14
Figure3.3.1.2: Sample image of Cnaphalocrocis medinalis (Leaf Folder Disease)	16
Figure3.3.1.3: Sample image of Fulgoromopha (Plant Hoppers)	17
Figure 3.4.1.1.1: CNN Model Configuration File	19
Figure 3.4.1.2.1: Architecture of CNN Model	20
Figure 3.4.2.1: Importing VGG16 Model	20
Figure 3.4.2.3: Importing Densenet201 Model	21
Figure 3.4.2.4: Importing xception Model	21
Figure 3.5.2: Image Rescaling	22
Figure 3.5.4: Features and labels of CNN	23
Figure 4.1.1.1: Architecture of VGG16 Model	24
Figure 4.1.1.2: Summary of VGG16 model	25
Figure 4.1.1.3: Performance of VGG16	26
Figure 4.1.1.4.1: Accuracy of VGG16	26
Figure 4.1.1.4.2: Loss of VGG16	26
Figure 4.1.1.5: Some output of VGG16	27
Figure 4.1.3.1: Architecture of Densenet201 Model	32

Figure 4.1.3.2: Summary of Densenet201 model	33
Figure 4.1.3.3: Performance of Densenet201	33
Figure 4.1.3.4.1: Accuracy of Densenet201	34
Figure 4.1.3.4.2: Loss of Densenet201	34
Figure 4.1.3.5: Some output of Densenet201	35
Figure 4.1.5.1: Architecture of xception Model	39
Figure 4.1.5.2: Summary of xception model	40
Figure 4.1.5.3: Performance of xception model	40
Figure 4.1.5.4.1: Accuracy of xception model	41
Figure 4.1.5.4.2: Loss of xception model	41
Figure 4.1.5.5: Some output of xception model	42

LIST OF TABLES

TABLES	PAGE NO
TABLE 2.3: LITERATURE REVIEW OF VARIOUS PAPERS ON CLASSIFICATION TECHNIQUE AND THEIR ACCURACY	8
TABLE 3.2: COLLECTED DATA	11
TABLE 3.3: DIVISION OF DATASET FOR TRAIN, TEST and VALIDATION	12
TABLE 4.2.1: DETAILES TABLE OF APPLIED MODEL	43
TABLE 4.2.2: COMPARISON TABLE OF APPLIED MODEL	43

TABLES	PAGE NO
TABLE 2.3: LITERATURE REVIEW OF VARIOUS PAPERS ON CLASSIFICATION TECHNIQUE AND THEIR ACCURACY	8
TABLE 3.2: COLLECTED DATA	11
TABLE 3.3: DIVISION OF DATASET FOR TRAIN, TEST and VALIDATION	12
TABLE 4.2.1: DETAILES TABLE OF APPLIED MODEL	43
TABLE 4.2.2: COMPARISON TABLE OF APPLIED MODEL	43

CHAPTER 1
INTRODUCTION

1.1 Introduction

Bangladesh is a heavily agricultural area, which implies that a significant portion of the population is committed in the trade. We yield rice on around 75% of our agricultural land [1]. because Bangladesh's staple ingredient is upward progress. Bangladesh was ranked as the fourth-largest rice producer in the United States in 1980. According to reports, Bangladesh is the world's sixth-largest rice supplier, according to [2] statistics. Despite the fact that Bangladesh is almost self-sufficient in rice production, the basic diploma farmers still face some challenges in delivering rice. Heavy Rainfall, Drought, Aridity, Flood, Illness, and many others are among them. Other than those environmental concerns, the key concern with rice farming is disease. However, small-scale farmers lack the scientific expertise necessary to grow rice or resist disease. We chose to apply machine learning techniques to solve the problem because we are living in a micro modern period.

1.2 Motivation

Bangladesh is the 0.33 worldwide United States of America. That is increasing. As a nation in development, the United States must overcome various obstacles as it advances. As farming is still the main source of income in the United States, its agriculture is a key component of its development. However, it is ironic that we continue to rely on the guiding cultivation equipment, which is a very slow method of identifying illness, assuming there is any. There needs to be a convenient and simple technique to discern the trouble because it's not always possible for humans to recognize highly important things or symptoms. Every day more things are getting digitalized. However, our agriculture sector is still trailing behind it. We must endure a significant loss each year due to the lack of appropriate agricultural competence among our root stage farmers. Keeping this in mind, we came up with the idea of doing something for our agricultural through the application of our computer-based technological knowledge. Then we opted to rely on the facts in the Rice category. Without a catastrophic event, only a few dangerous illnesses harm rice plants periodically, hence we want to endure a steep decline. Because of this, we created a device mastering version that could take into account rice leaf disease. The goal of this research is to quickly and accurately identify illnesses. Three ailments were used in our research. which are:

- Blast sickness or Magnaporthe Grisea
- Leaf Folder disorder or Cnaphalocrocis Medinalis
- Plant Hopper illness or Fulgoromorpha

1.3 Rational of the Study

Rural areas currently have access to smartphones. Despite the fact that most of our rural population lacks even a basic comprehension of education, technology offers enough information for them. Sincerely, that is the reason we created an Android application to carry out the software way. This mobile application will use a camera to take pictures of leaves, or users can choose specific leaf pictures from a gallery to view. Additionally, they are able to determine whether the plant is still healthy or not. The treatment and eventual resistance of the sickness can be known after the diagnosis of the illness.

Another intriguing aspect of the idea is that farmers can contact any time, surrounding agrarian listings with questions by phone or the helpful chat feature. For our Android app to be more user-friendly, English and Bangla should be at least two of the key dialects.

1.4 Research Questions

Throughout the course of research, we will encounter numerous questions. The first query we should ask concerning our record-keeping is. It is not easy to compile a sizable dataset on a particular illness. Due to this fact, not all types of illnesses are currently prevalent in a single location. So in any case, we have gathered them. The second question that arises is then what kind of neural network do we need to use. To us, this plainly marked the beginning of a new chapter. however, we compile data on it with the help of our supervisor. The method we must use to complete it has then appeared to be complicated in a few distinct ways. The Python language was demonstrated to us next by our manager. Then, we compile data on Anaconda, Scipy, and Numpy and conduct 10–12 research papers.

1.5 Expected Outcomes

Most peasants in our region are struggling to make ends meet. Our goal may therefore be to be open to everyone. The majority of them know virtually little about rice farming. Our report's very last conclusion is the potential to diagnose paddy disease using paddy leaves and to provide a step-by-step rehabilitation. Our research can also assess whether a leaf is healthy or not. To provide a truly user-friendly interface to learn about the issue, we developed an Android application.

1.6 Project Management and Finance

The enormous expedition was overseen by **Shayla Sharmin**, our rightful supervisor. He led us throughout the entirety of the mission. Instead, we incurred a few costs, which is not a big deal. After receiving their schedule, we headed there in order to visit the Bangladesh Rice Research Institute (BRRI), where we have gathered our information [1]. To get there cost a little money. In addition, our walking tool (computer) isn't really set up to handle that difficult task. Therefore, we replaced our hard drive with a 120GB SSD, which cost about 2100/=, and an 8GB DDR3 RAM module, which cost roughly 1600/=. 4000 Taka have been used up in an unspecified period of time in the future. All expenses are incurred through us.

1.7 Report Layout

In an attempt acquire detailed overview of our research on "Detection of illness on Paddy leaves using tool Learnings" in chapter 1. Additionally, we talked about the inspiration behind our research, our hopes, and our anticipated findings.

We got talking about the genesis of our inquiry in chapter 2. We are able to focus on the related works, the rapid evaluation of the synopsis, the aspects used in our research, the breadth of the issues, and the challenging circumstances of doing this study.

Financial ruin three will empower us to discuss our research's methodology. We will briefly discuss the tools we used, the methods by which we gathered and used our data, a statistical analysis of the data we obtained, the types of policies we employed, and the requirements for implementing our indicated variant.

In going bankrupt 4, we will discuss information about the unorthodox outcome of our research. Moreover, we reassess our outcome.

The reflection of our activities on society and the surroundings may be the most important talking points in chapter 5. If our investigations have any ethical components, they should be stated right here.

We can accomplish our evolutionary process in financial disaster 6 and upload some more advanced delivers that might be used in future ventures.

At the end of this document, we provide a few references that we found online and in various prior studies.

CHAPTER 2
BACKGROUND

2.1 Preliminaries

In our analysis, we employed the Convolutional Neural community in particular (CNN). Here, the procedural programming of choice was Python. In order to calibrate our model, we used Tensor float 2.zero. We have collected data from the Bangladesh Rice Studies Institute (BRRI). We received a lot of assistance from Dr. Samul Haque Sir, a prominent scientific officer. He gave us the go-ahead to take pictures in their research lab. From there, we gathered over 12,000 actual photos. We used some open-source equipment for preprocessing aspects like labeling, noise reduction, scaling, and adding a few effects. In our study, names of the diseases were utilized to mark the elegance in a controlled mastering set of rules.

2.2 Related Works

As far back as we could tell from the BRRI analytics, the first similar task was accomplished in 2017 with the aid of a BUET student; but, due to some stability issues, the application was never posted. A daffodil student undertook some extensive research on paddy disease detection. For a long time, there were many inquiries about our work that might have been related to my research. A few of them are listed below:

In paper [1], Dr. Neha Mangla, Pryanka B. Raj, Soumya G. Hegde, and Pooja R. completed a Paddy Leaf Disease Detection Machine using Image Processing and System Learning. The framework was evaluated using accuracy and assessment, and it was observed that the framework had improved to **ninety-four.16** percent accuracy in classifying the three rice disorder. According to Neha et al. (2019), paddy leaf sickness can be detected through photo management and device control. This study's focus is on the Rice effect, Sheath scourge, and brown spot.

A conceptual mechanism for the detection and classification of rice illnesses was envisioned in the work [19] by Harshadkumar B. Prajapati, Jitesh P. topped head, and Vipul K. Dabhi, and it was backed by images of sick rice plants. Once developed, this ideal equipment is better.

Comparison of diverse image processing techniques through experimentation. They used SVM to classify the illness, and they received 93.33 percent honing symmetry and 76.333 percent trying accuracy. Additionally, they performed well enough: fold skip approval fork=5 and ok=10. Additionally, they developed a user-friendly GUI for information on all intermediate phases, including the entry of photos and disease classification.

In a paper [3] A scrutinize on spotting rice plant diseases with photos Statistics and processing Tanmoy Bera, Ankur Das, Jaya Sil, and Asit Okay Das are mining techniques. The key to anticipating both a qualitative and quantitative decline in agricultural productivity is observable proof of the diseases. They used the classifiers k-NN and MDC, and they achieved accuracy rates of 87.02 and 89.23%, respectively. Phadikar et al. presented an automated contamination kind of leaf brown spot and leaf blast in [17]. One thousand images of diseased rice leaves were used to gauge the accuracy of the Bayes and SVM classifiers. They suggested a method for identifying type regulations for Egyptian rice illnesses using C4.5 decision tree calculations. they had regarded it as seven agronomic trends for analysis's sake. Facts pick up and entropy have been used as trait willpower assessments. In my opinion, 96.4% and 97.25% of the neural network and C4.5 choice tree algorithms were successfully executed.

Paddy Plant Leaf: Automated Blast Malady Detection and a Color-Reducing Mechanism Maninder Lal Singh's Method is detailed in the publication [4], which was generated using Amandeep Singh. An unique mechanism for propagating paddy crop blast disease has been proposed in this paper. The suggested computation's accuracy has been found to have increased by **96.6%** in light of comparisons with data from [15] and [16]. A cursory look reveals that:

- A promising approach for the sensing of a certain type of shade is color attenuation.
- Despite certain inaccuracies in the discovery, the sensitivity of 96.6% is astonishing in its beauty.
- There is a novel approach to component appearance acting that outperforms conventional side discovery technologies.

These techniques can be employed in live initiatives in real-time leveraging histogram-based tactics that have been evaluated in and for color discovery leveraging once it has been verified the circumstances that occurred in diverse environmental timeframes.

Hereditary algorithms or neuron-fuzzy optimization methods can both have their capacities increased.

Amrita A. Joshi and B.D. Jadhav's work, "Monitoring and Controlling Rice Diseases Using Photo Processing Strategies," describes their findings. On the basis of this observation, a framework has been developed for investigating the rice bacterial scourge, rice effect, brown spot diseases, and rice sheath degradation. The kind set of policies have been established using image-processing tools like segmentation, feature extraction, and two classifiers. Color and zone clever shape abilities were extracted from the highlight extraction and applied as input to the classifier. A remote database has been used for upskilling and tweaking purposes for each illness. Classifiers, adequate-NN, and MDC were used in the proposed approaches for the four rice illnesses listed, and, in my opinion, they performed with a precision of 87.02% to 89.23%. The suggested tactic is juxtaposed to barely a few survived procedures that are relevant to the finding of rice illnesses, and it is determined that the envisioned method is fundamental in terms of time ambiguity, calibration, and the number of illnesses safeguarded.

2.3 Comparative Analysis and Summary

Since the scrambling, numerous device-studying techniques have been used on a dataset that is thought to be unique to find disease on paddy leaves.

A SVM model was completed on a dataset of 1500 extracts from four classes in paper [2]. In this instance, the accuracy increased to 79%.

Here, a total of one thousand images from eleven distinct lessons were used in paper [3]. Despite the little dataset, the accuracy ended up being 87%. One hundred twenty data points from three training sessions are employed in study [4]'s very small data set. The accuracy increases to 90% and the collection of rules becomes k-NN. Four distinct magnificences—Tungo, Blast, Brown spot, and Bacterial leaf—were utilized in paper [5]. The accuracy rose to 80.3 percent.

A dynamic planetary software was given to understand the diseases in the paper [6]. If so, there is evidence that the exactant of paddy sickness is 91%.

About 98% accuracy was provided for our project, which was reasonable.

Here is a table showing the imposed set of instructions and accuracy from each unique paper:

TABLE 2.3: LITERATURE REVIEW OF VARIOUS PAPERS ON Parameters of CLASSIFICATION TECHNIQUE AND THEIR ACCURACY

Parameters of classification	**Accuracy**
minor axis, major axis, and edge [7].	76.59%
Relative magnitude deviation and since ingredients of pigment [8].	97.83%
Texture, shape and Color features [9].	97%
Three layers: hidden layers, layers, and output layers [10].	70–80%
Texture values are estimated utilizing VI index [11].	84%
Zone-wise shape features and Color [12].	89.23% 87.02%
1. (feature, value) pair 2. minimal feature subset [13].	More than 90%
Shape and texture and Color features of each image [Our Work]	**99%**

2.4 Scope of the Problem

Our farmers may gain financially from the walking machine since it allows them to detect paddy leaf disease. However, there are several procedures that must be followed. They involve collecting photographs as input, saving them, carrying out various tasks, and taking certain actions to align the image with our dataset. Although all of these tactics are esoteric, American farmers generally lack sufficient technical knowledge. Because the Android mobile utility platform can be used to access the whole research methodology. Therefore, if they are familiar with Android software, they should have no trouble completing the task and determining the exact problem.

2.5 Challenges

We consolidated data for this endeavor from the Bangladesh Rice Research Institute (BRRI). We gathered about ten thousand agitated photos from their greenhouse. In this study, we essentially created artwork for three different diseases. Rice Leaf Folder disease, Blast disease, and Leaf Fly disease. Farmers are attempting to deal with the inconvenience of paddy leaf diseases. Our nation wastes a lot of flowers as a result of these problems. Therefore, we are attempting to resolve this issue by efficiently identifying the appropriate illness in accordance with technical methods.

Our primary difficult platform moved into gathering the dataset in order to complete the work. It has become impossible to complete the project without dealing with a significant amount of data. As all of our statistics are raw data, we didn't accumulate a single image from the internet. Another difficult task was making a time to see Dr. Samul Haque Sir at (BRRI). He is the Bangladesh Rice Studies Institute's senior clinical officer (BRRI). We didn't take images of the (BRRI) green home without his consent. Then, our PC became improperly set up. For photo education, we will have to deal with numerous challenges. Finishing the assignment might be more appropriate for us if our laptop had an excessive configuration.

CHAPTER 03
RESEARCH METHODOLOGY

3.1 Research Subject and Instrumentation

In this context, we leverage a deep convolutional neural network to integrate our self-assembled dataset. That serves as the main framework for our study. We have tested the output accuracy of our current version with a few other models. The next section can be used to define these models' overall form and functionalities. As we've already stated, the primary goal of our research is to categorize three great diseases found in paddy leaves that contain 10,000 photographs. In order to train the most important CNN model, we used 6300 images. then 2700image for verification needs. 100 images are then utilized for testing. 900 images were employed for validation capabilities in our switch learning models (VGG16, VGG19, and ResNet50), whilst 9,000 images were used for educational features. The reduction of the 100 photos has been put to the test.

3.2 Data Collection Procedure

The primary objective for us changed early on in the research to include acquiring the data. The image was compiled using only you. We obtain the raw data from the Bangladesh Rice Studies Institute (BRRI). We drove there on the assumption of our advisor, and we didn't snap a picture from their laboratory until we had consent. For the purposes of their research, they archived the paddy plant. With one in all a variety of consequences, including ISO, Focal length, exposure, aperture, and extraordinary frame length, we took the picture using an Android device called the Samsung A50.

TABLE 3.2: COLLECTED DATA

Title of the Class (English)	Title of the Class (In Scientific method)	Class label	Total Image
Blast	Magnaporthe grisea	Zero	3300
Leaf folder	Cnaphalocrocis medinalis	One	3300
Plant Hoppers	Fulgoromorpha	Two	3400

3.3 Statistical Analysis:

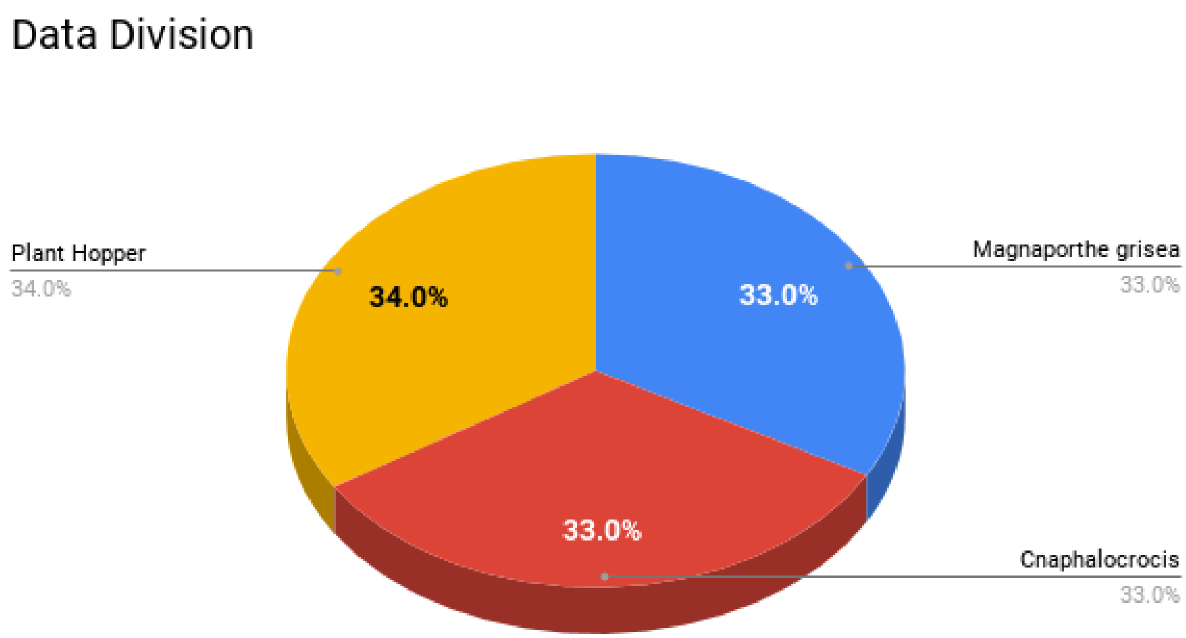

Figure 3.3.1: Pie chart of total data.

The Parameter Segmentation of our research is shown in Table 3.3.

TABLE 3.3: DIVISION OF DATASET FOR TRAIN, TEST and VALIDATION

Name of the Class	**Label of the Class**	**Train Data**	**Validation Data**	**Test Data**	**Total Data**
Magnaporthe grisea	Zero	2100	900	300	3300
Cnaphalocroc is medinalis	One	2100	900	300	3300
Fulgoromorp ha	Two	2100	900	400	3400

Below is a bar chart comparing data partition:

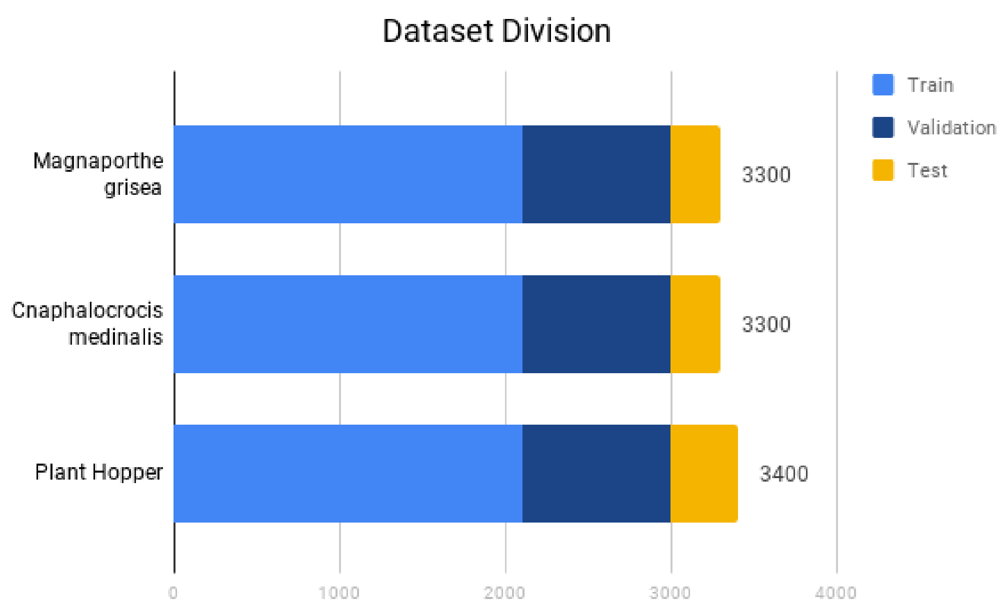

Figure 3.3.2: A Bar chart of total data division

3.3.1 Sample Image:

3.3.1.2 Magnaporthe Grisia (Blast Disease):

Magnaporthe grisea is a fungus that is hazardous to plants and promotes a chronic case in rice. It is also known by the names rice rotten neck, rice blast fungus, rice seedling blight, blast of rice, oval leaf spot of graminea, pitting sickness, ryegrass blast, and Johnson spot. It is currently understood that M. grisea is made up of a complex of cryptic species, at least some of which are natural species with distinct genetic lines that no longer interbreed. Complex contributions that were isolated from Digitaria were more precisely classified as M. grisea [8]. Magnaporthe oryzae was given to the very last members of the complex that had been isolated from rice and a innumerable other harbors. Both of these names are now being used by distinct writers, therefore it is unclear which one to use for the rice blast pathogen. Magnaporthe grisea cluster members can transmit disease such as blast disorder or blight disease to other vital cereals for agriculture, particularly wheat, rye, barley, and pearl millet. The rice blast leads in yearly crop losses that are remarkable. It is projected that enough rice would be broken each year to nourish more than 60 million people. According to theories, the fungus can be found in 85 different countries globally. On all parts of the shoot, the first symptoms appear as white to gray-green lesions or patches with darker edges, while older lesions are elliptical or spindle-shaped, whitish to gray, and have necrotic borders. The entire leaf can potentially be killed by lesions that become larger and combine. All of the plant's above-ground additions include placards and other markings. The culm breaks off at the inflamed node as a result of nodal infection (rotten neck). It also impairs proliferation since the parasite delivers fewer embryos as a reaction. This occurs as a result of the disease-preventing genuine grain maturation. Below are a few instances of this disorder's pattern photos:

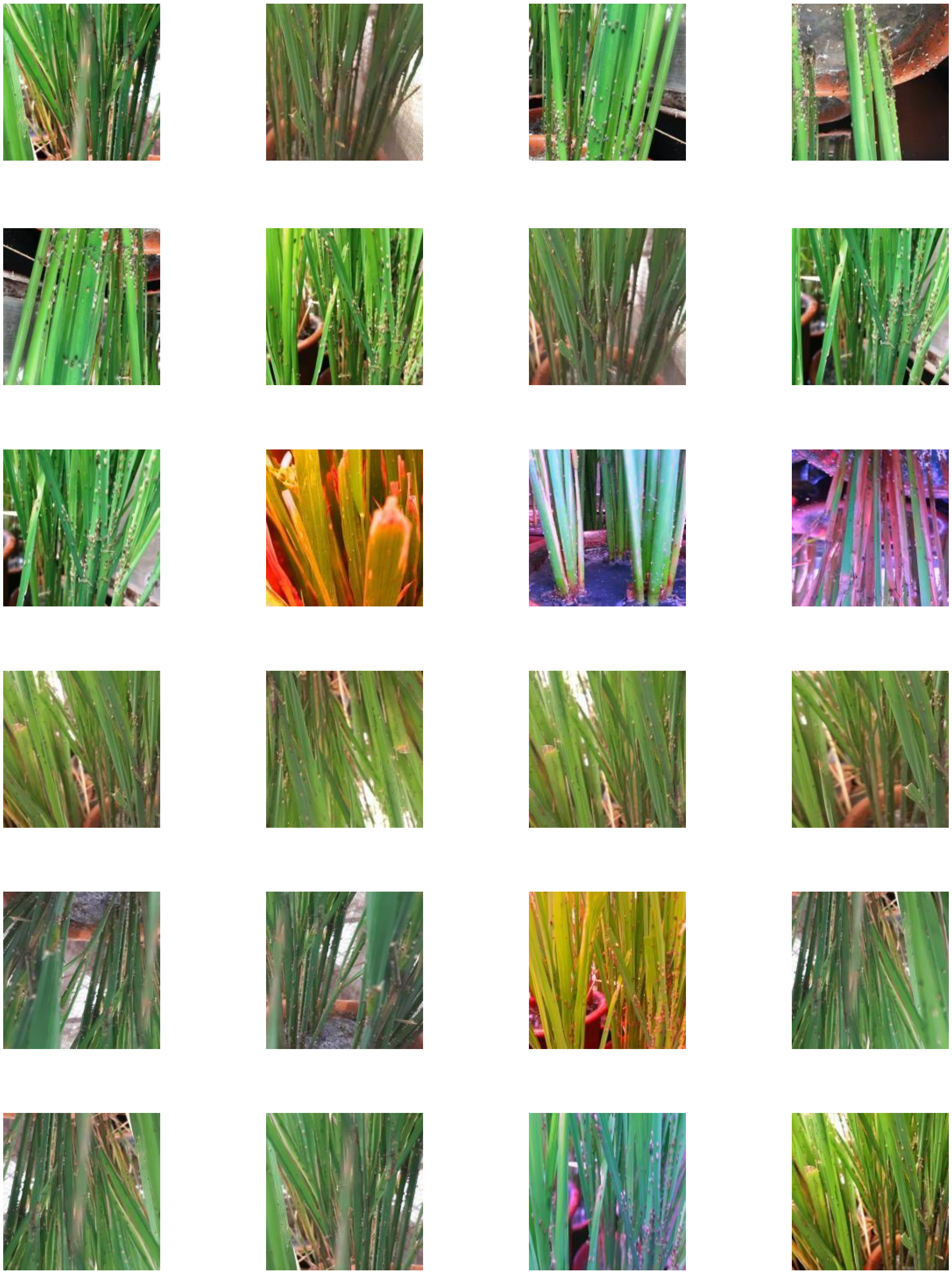

Figure 3.3.1.1: Sample Images of Magnaporthe grisea (Blast disease)

3.3.1.2 Cnaphalocrocis medinalis (Rice Leafroller Disease):

The rice leafroller, or Cnaphalocrocis medinalis, is a genera of moth that pertains to the Crambidae family. It is established in South-east Asia, along with Sri Lanka, Taiwan, Thailand, Hong Kong, and much of Australia. The wingspan reaches about 16 mm. The larvae on the species of Zea mays, Oryza sativa, Triticum, Saccharum, and Sorghum are regarded as a pest. The moth is very active, straw-colored or bright yellow, and has one spectacular wavy line inside its hindwing and two incredible wavy lines inside its forewing. The wing span of it is 15 mm. On the underside of the leaves, which may be scaly white in color, eggs are placed individually or in clusters and are aligned axially rows. The fecundity has 56 eggs ready. Four to eight days pass during incubation. We find 5–6 larval instars; the larvae are ready in 22–23 days. It spends 6-7 days pupating inside the infected leaf fold. The fully developed caterpillar is 16.5mm long and has a green color. The entire cycle of lifestyles was finished in around five weeks. This pest affects the entire spectrum of the crop. The newly hatched caterpillar flattens and folds the leaf after hatching. When attacking younger seedlings, it folds three to four adjacent sprout leaves and grinds the green tissue away, making the infected leaves emerge white. Multiple leaves are harmed by a single caterpillar. The plant's strength is diminished as the damaged flora dries up. In the end, the gain declines. The revenue loss could be up to ten to fifty percent. The boot leaf degree is much more complicated.

A few instances of graphics are shared below:

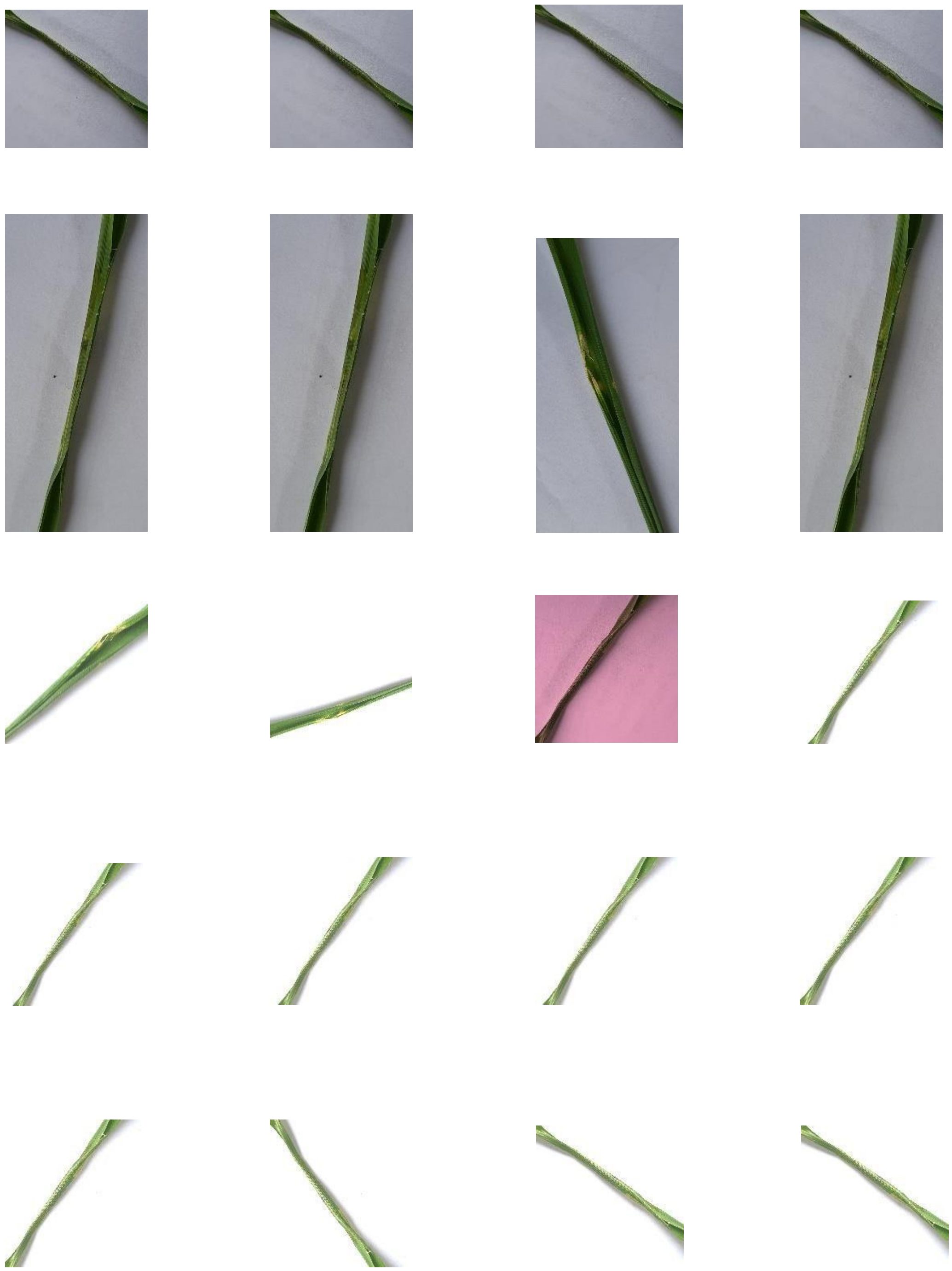

Figure 3.3.1.2: sample photo of Cnaphalocrocis medinalis (Leaf Folder ailment)

3.3.1.3 Fulgoromorpha (Plant Hopper Disease):

Any insect with more than 12,500 identified species worldwide and belonging to the suborder Auchenorrhyncha of the infraorder Fulgoromorpha is referred to as a plant hopper. The word "hop" refers to their striking similarity to leaves and characteristic shrubs in their framework as well as the fact that they frequently "HOP" for short distances, much like grasshoppers do. Plant hoppers, however, typically move very slowly to avoid drawing attention. All members of this association, which is dispersed around the planet, are plant-feeders, however only a small number are regarded as pests. The Fulgorodia superfamily is the only one that makes up the infraorder most efficiently. The bifurcate anal vein within the the dilated and the annexure are the two characteristics that most consistently distinguish Fulgoroidas from the potential Acuhenorrhyncha.

The approach graphic is furnished:

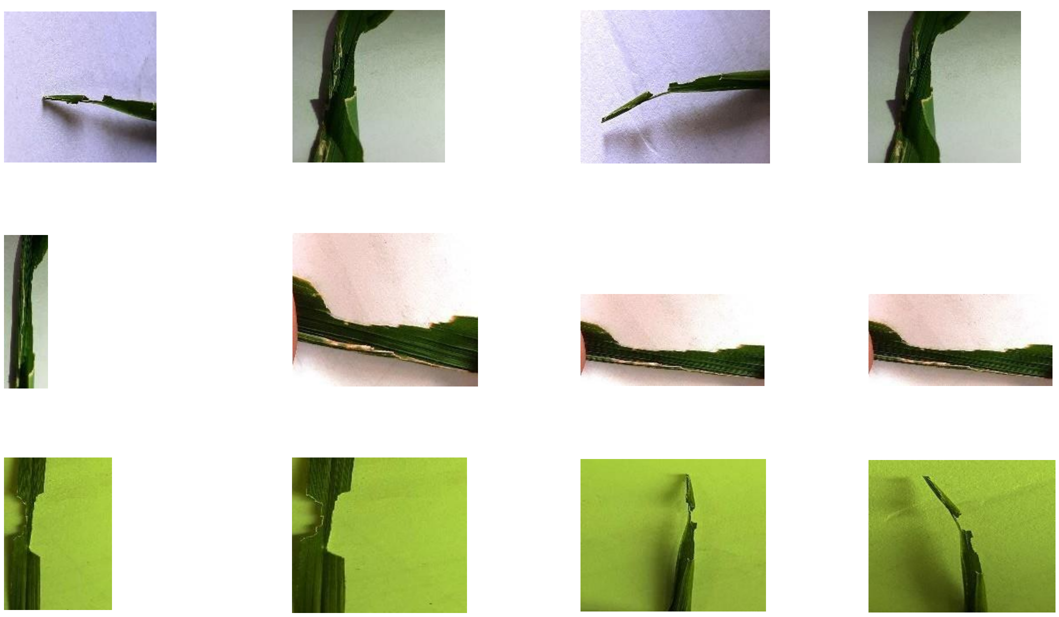

Figure 3.3.1.3: pattern photo of Fulgoromopha (Plant Hoppers)

3.4 Applied Mechanism

In this review, several models were created. Convolutional Neural Networks (CNNs) are the key model that we have worked with. To check the accuracy of the result, the other fashions are used. The patterns are often:

- VGG16
- Densenet201
- Xception

3.4.1 CNN Model Configuration:

This section gives us the opportunity to talk about the model we've used. We were given two convolution layers, two layers of two-period maximum pooling, three thick layers, and total three magnificence. Finally, we get to an activation layer where the "gentle ax" quality is used as an activation characteristic. We chose Softmax because we could employ a variety of specific entrance charges, such as excellent, poor, or zero. But we need a cost that ranges from 0 to at least 1. A F real fee of vector is transformed into an acceptable vector of real cost by this feature, which must total to at least one.

Instead, we employed "ReLU," which stands for a Rectified linear activation feature, in our hidden layers. If the input is positive, this characteristic will immediately return it as an output. If the input is subpar, it will produce 0 in any other situation.

3.4.1.2 CNN Model:

The model that we have created is:

```
model = Sequential()
# 3 convolutional layers
model.add(Conv2D(64, (3, 3), input_shape = X.shape[1:]))
model.add(Activation("relu"))
model.add(MaxPooling2D(pool_size=(2,2)))
model.add(Conv2D(64, (3, 3)))
model.add(Activation("relu"))
model.add(MaxPooling2D(pool_size=(2,2)))
model.add(Flatten())
model.add(Dense(64))
model.add(Activation("relu"))
model.add(Dense(32))
model.add(Activation("relu"))
model.add(Dense(3))        #output layer with 3 neurons, for 3 classes
model.add(Activation("softmax"))    #softmax is an Activation layer
```

Figure 3.4.1.1.1: CNN model Configuration File

Here, the first component we have are convolutional layers. Then, we received a feature called "ReLu" for activation. This activation function "ReLu" returns a tensor from an input. The rectified linear unit activation characteristic is used in ReLu. Then, using 2, we obtain a maximum poling of period 2.
Convolutional neural layers follow, with "ReLU" acting as an upregulation function repeatedly. However, a most pooling layer of length 2 by 2 has been provided to us.

Then, since we have three teachings, there are three layers that are undoubtedly linked. The Softmax activation function was provided on the final totally linked layers. The Softmax property changes a real vector into a vector of accurate probabilities. Due to the fact that the last layer activation for a class network can be viewed as a hazard distribution, Softmax is typically used.

3.4.1.2 Architecture of Our Model:

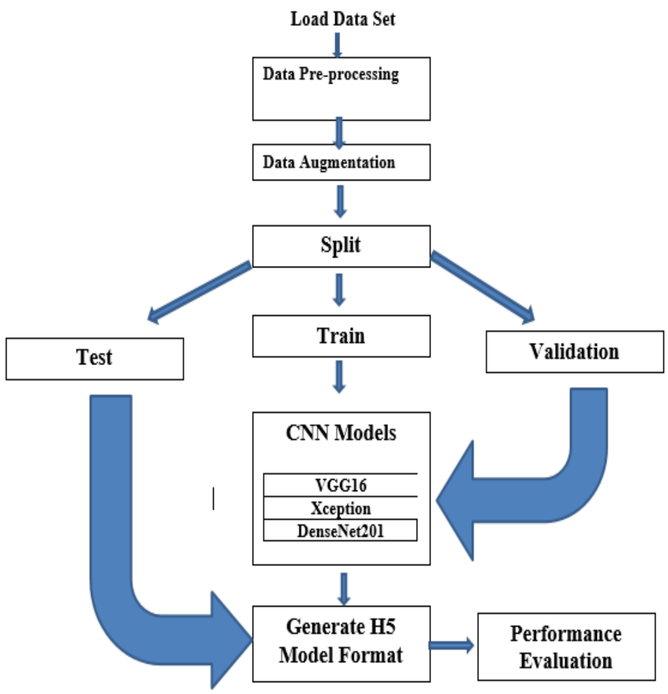

Figure 3.4.1.2.1: Architecture of CNN Model

3.4.2 Transfer Learning Models:

There are various switch getting to know fashions. As we've got were given used four switch getting to know models (VGG19, VGG16, Resnet50, Xception), we can have a quick dialogue on them inside the subsequent element.

3.4.2.1 Importing VGG16:

```
vgg = VGG16(input_shape=IMAGE_SIZE + [3], weights='imagenet', include_top=False)
```

Figure 3.4.2.1: Importing VGG16 Model

3.4.2.2 Importing Densenet201:

```
resnet = ResNet50(input_shape=IMAGE_SIZE + [3], weights='imagenet', include_top=False)
```

Figure 3.4.2.3: Importing ResNet50 Model

3.4.2.3 Importing Xception:

```
base_model=tf.keras.applications.xception.Xception(weights='imagenet',include_top=False)
```

```
WARNING:tensorflow:From C:\Users\Sarah\Anaconda3\lib\site-packages\tensorflow\python\ops\resource_variable_ops.py:435: colocate
_with (from tensorflow.python.framework.ops) is deprecated and will be removed in a future version.
Instructions for updating:
Colocations handled automatically by placer.
Downloading data from https://github.com/fchollet/deep-learning-models/releases/download/v0.4/xception_weights_tf_dim_ordering_
tf_kernels_notop.h5
83689472/83683744 [==============================] - 1246s 15us/step
```

```
for layer in base_model.layers:
  layer.trainable = False
```

Figure 3.4.2.4: Importing Xception Model

3.5 Implementation Requirements

Preprocessing of the findings is a cornerstone of this analysis. This is the prerequisite in machine learning algorithms, which follows data collecting. The next sections define the preprocessed approach that we employed for our experiments.

3.5.1 Image Resizing

We have material about distinct pixel sizes. The only way the tool can identify an image is by using its matrix value. Therefore, applying one kind size of picture in machine learning algorithms is not a great exercise at this time. We clip the picture.

3.5.2 Normalization

Prior to teaching the information set, information standardization is a crucial task. Every parameter (in this case, Pixel), which is guaranteed by the normalization procedure, receives the same data branch. By first removing each pixel's proposed value and then dividing the result by the standard deviation, normalization is calculated. For photo entry, the pixel input must be excellent. Therefore, we must choose a scale that ranges from 0 to at least 1. Three RGB channels make up a colored image (R=red, G=green, B=blue). Nevertheless, there are several 0 to 255. In our instance, we expanded the Pickle record that encompasses the photo pixel and divided it by 255 to replicate the data.

```
]: X = pickle.load(open("X.pickle", "rb"))
   y = pickle.load(open("y.pickle", "rb"))

   X = X/255.0            #normalizing data (a pixel goes from 0 to 255)
```

Figure 3.5.2: Image rescaling

3.5.3 Splitting of the dataset

Datasets are used to engage the machine so that it can distinguish it. The machine observes the dataset and adapts from it. Next, we were instructed to evaluate how effectively the tool had assumed from the dataset. The system's findings from the model should then be verified. Thus, we must corroborate the model's findings. The device's valuable information must next be put to the test. It is for this reason that the records set must be divided into the three sets of education, validation, and testing.

3.5.4 Featuring Data and labeling the class:

We need to add a variety of photographic capabilities to our self-made, top-ranked CNN version. The capabilities must then be labeled according to their elegance. We achieved this by employing the neat line of code below:

```
X = [] #features
y = [] #labels
for features, label in training_data:
    X.append(features)
    y.append(label)
#X = np.array(X).reshape(-1, IMG_SIZE, IMG_SIZE, 1)
```

Figure 3.5.4: Features and labels of CNN model.

CHAPTER 4

EXPERIMENTAL RESULTS AND DISCUSSION

4.1 Experimental Setup:

We must also craft several alternative outfits in order to test our concept. This area of the computer industry isn't really new. We've already talked about how some earlier work also had an impact on it. In our studies, there are several PR knowledgeable models that must be applied.

4.1.1 VGG16:

Vertex Geometric Organization, or Vgg16, has Sixteen Strata. The first 13 strata are convolutional layers, while the final 3 are almost parallel layers. This is a widely used pre-train model for switch learning. In 2014, it was put out by Andrew Zisserman and Karen Simonyan in an ImageNet competition. It represented the year's first and second functions. On a PC with an entirely excessive configuration, the correct model took two to three weeks to assemble.

4.1.1.2 Architecture of VGG16:

The structure of this VGG16 underlying model:

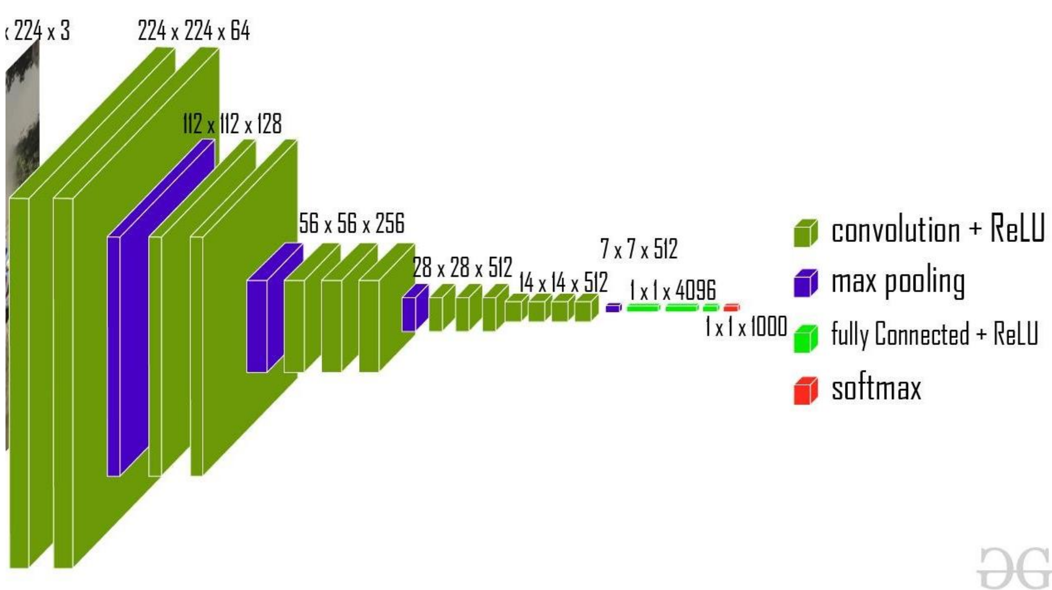

Figure 4.1.1.1: Architecture of VGG16 Model:

This model is consisting of three genuinely associated layers and thirteen convolutional models with the Amplification characteristic ReLu. Overarching sixteen layers.

4.1.1.2 VGG16 Estimator:

Layer (type)	Output Shape	Param #
input_1 (InputLayer)	(None, 224, 224, 3)	0
block1_conv1 (Conv2D)	(None, 224, 224, 64)	1792
block1_conv2 (Conv2D)	(None, 224, 224, 64)	36928
block1_pool (MaxPooling2D)	(None, 112, 112, 64)	0
block2_conv1 (Conv2D)	(None, 112, 112, 128)	73856
block2_conv2 (Conv2D)	(None, 112, 112, 128)	147584
block2_pool (MaxPooling2D)	(None, 56, 56, 128)	0
block3_conv1 (Conv2D)	(None, 56, 56, 256)	295168
block3_conv2 (Conv2D)	(None, 56, 56, 256)	590080
block3_conv3 (Conv2D)	(None, 56, 56, 256)	590080
block3_pool (MaxPooling2D)	(None, 28, 28, 256)	0
block4_conv1 (Conv2D)	(None, 28, 28, 512)	1180160
block4_conv2 (Conv2D)	(None, 28, 28, 512)	2359808
block4_conv3 (Conv2D)	(None, 28, 28, 512)	2359808
block4_pool (MaxPooling2D)	(None, 14, 14, 512)	0
block5_conv1 (Conv2D)	(None, 14, 14, 512)	2359808
block5_conv2 (Conv2D)	(None, 14, 14, 512)	2359808
block5_conv3 (Conv2D)	(None, 14, 14, 512)	2359808
block5_pool (MaxPooling2D)	(None, 7, 7, 512)	0
flatten_1 (Flatten)	(None, 25088)	0
dense_1 (Dense)	(None, 3)	75267

Figure 4.1.1.2: Summary of VGG16 model

4.1.1.3 Performance of VGG16

```
WARNING:tensorflow:From C:\Users\Sarah\Anaconda3\lib\site-packages\tensorflow\python\ops\math_ops.py:3066: to_int32 (from tenso
rflow.python.ops.math_ops) is deprecated and will be removed in a future version.
Instructions for updating:
Use tf.cast instead.
Epoch 1/3
100/100 [==============================] - 3199s 32s/step - loss: 0.7626 - accuracy: 0.9422 - val_loss: 1.1921e-07 - val_accura
cy: 0.9960
Epoch 2/3
100/100 [==============================] - 4262s 43s/step - loss: 0.1251 - accuracy: 0.9921 - val_loss: 1.1921e-07 - val_accura
cy: 0.9940
Epoch 3/3
100/100 [==============================] - 3420s 34s/step - loss: 0.0756 - accuracy: 0.9953 - val_loss: 1.1921e-07 - val_accura
cy: 0.9940
```

Figure 4.1.1.3: Performance of VGG16

4.1.1.4 Visualization Graph of Accuracy and Loss:

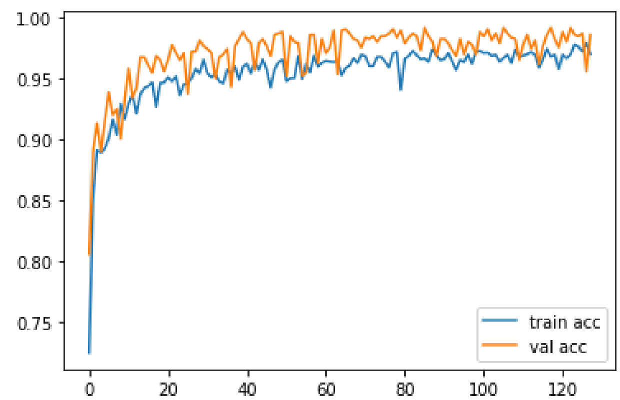

Figure 4.1.1.4.1: Accuracy of VGG16

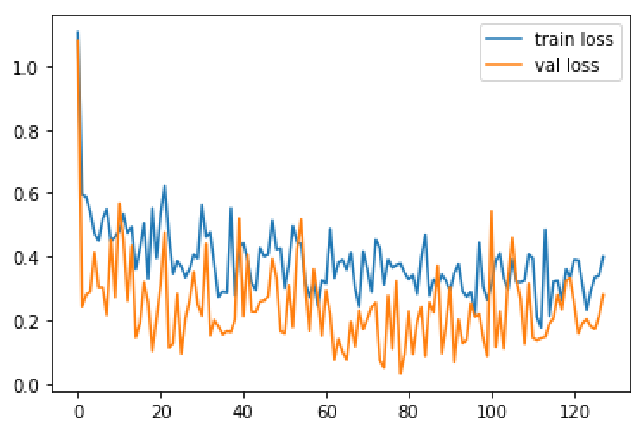

Figure 4.1.1.4.2: Loss of VGG16

4.1.1.5 Output of VGG16:

Numerous findings by this approach can be seen below:

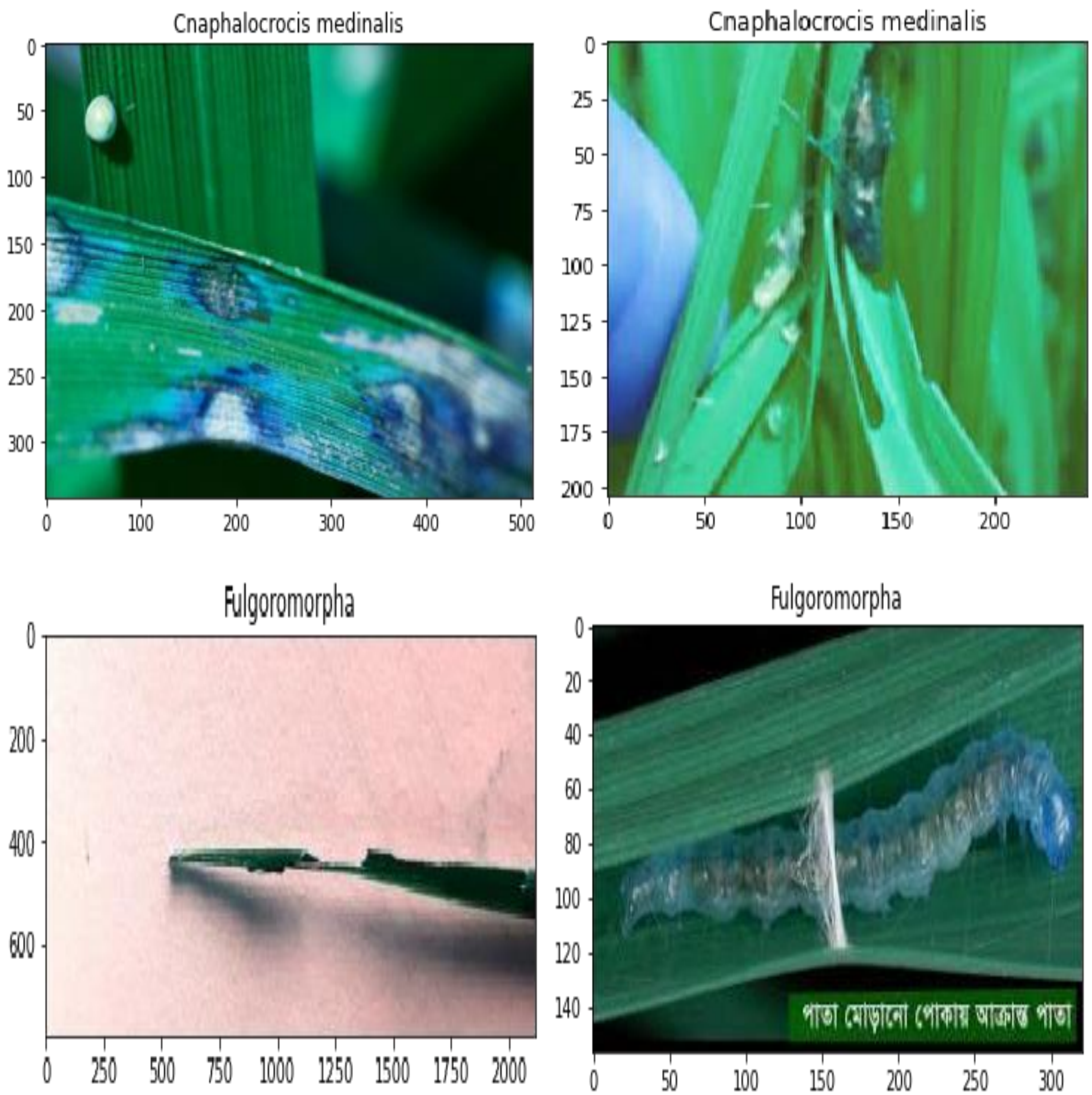

Figure 4.1.1.5: Output of VGG16

4.1.3 DenseNet201

DenseNet-201 is a convolutional neural network architecture developed by Gao Huang et al. and introduced in the paper "Densely Connected Convolutional Networks" (Huang et al., 2016). The architecture is known for its ability to efficiently learn deep networks and has been widely used for image classification and segmentation tasks.

4.1.3.2 Architecture of DenseNet201

Here is a diagram of the DenseNet-201 architecture:

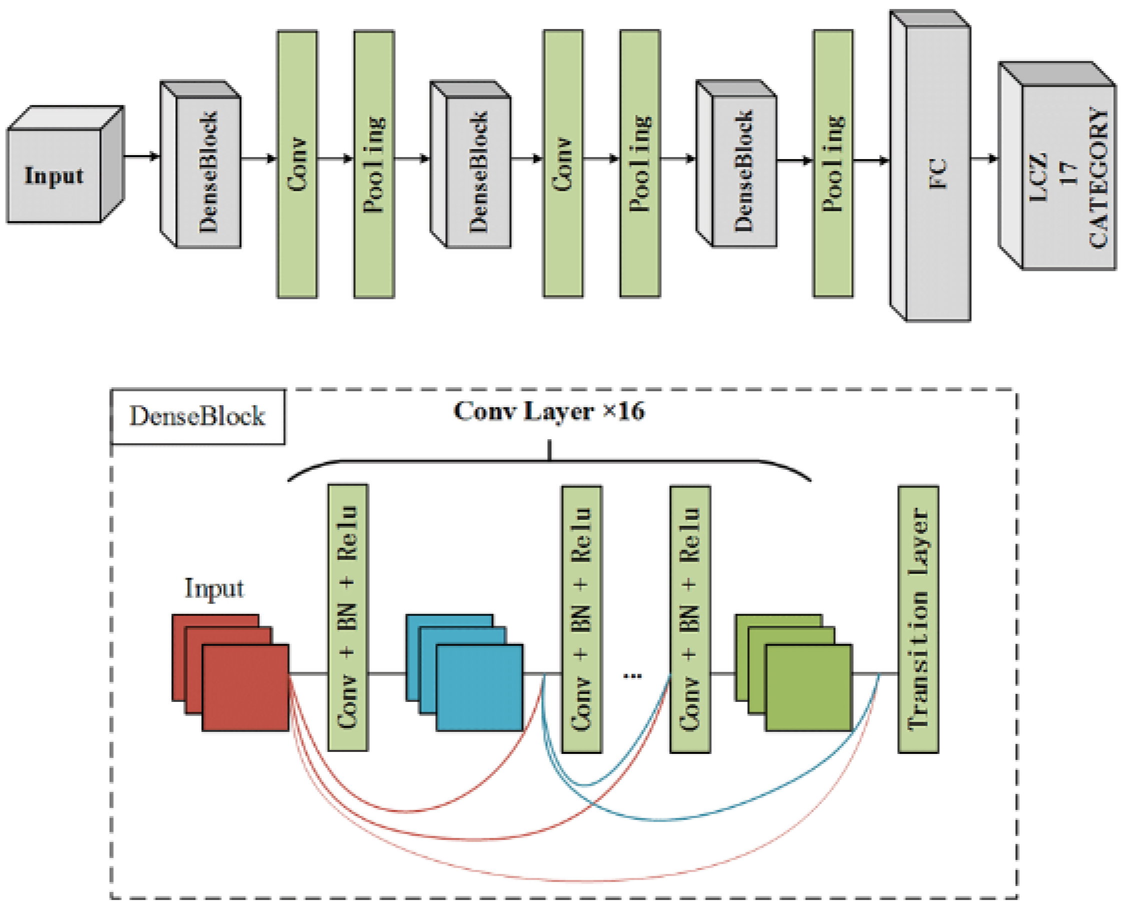

Figure 3.3.4.1: Architecture of DenseNet201

The DenseNet-201 architecture consists of a series of dense blocks, where each dense block contains a series of convolutional layers that are densely connected to the preceding layers. In a dense block, each layer receives the feature maps of all preceding layers as input, allowing the network to learn more efficient representations of the input. The output of the dense blocks is then passed through a series of transition layers, which reduce the resolution of the feature maps and increase the number of channels. The input to the network is an image, which is passed through the dense blocks and transition layers to extract features. The extracted features are then passed through a global average pooling layer and a fully connected layer, which perform classification on the features.

DenseNet-201 has achieved strong performance on a variety of image classification benchmarks, including the ImageNet dataset. On the ImageNet dataset, DenseNet-201 achieved a top-1 error rate of 22.3% and a top-5 error rate of 6.3% (Huang et al., 2016).

4.1.3.3 Input pipeline of DenseNet201

The input pipeline for DenseNet-201 typically involves preprocessing the input images before they are fed into the network. This preprocessing includes steps such as resizing the images to a fixed resolution, cropping the images to a square, and normalizing the pixel values.

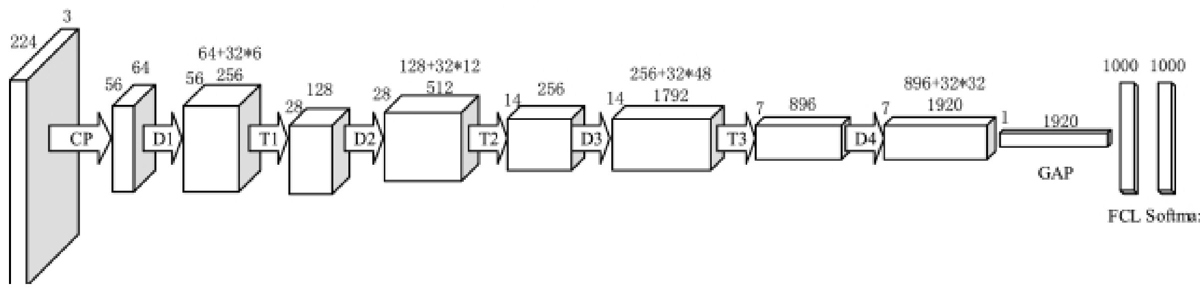

Figure 3.3.4.2: Input pipeline of DenseNet201

After the images have been preprocessed, they are passed through the network in the form of a tensor, where they are convolved and down sampled through a series of dense blocks and transition layers. The tensor is then passed through a global average pooling layer, which reduces the resolution of the feature maps, and a fully connected layer, which performs classification on the extracted features.

The final output of the network is a probability distribution over the classes, indicating the likelihood that the input image belongs to each class.

In addition to the main classification branch of the network, some architectures also include a branch for localization, which is trained to predict the bounding box coordinates of an object in the input image. The localization branch is made up of additional convolutional and fully connected layers that are added onto the main classification branch of the network.

4.1.3.4 Performance of DenseNet201

DenseNet-201 is a convolutional neural network architecture that has been widely used for image classification and segmentation tasks.

DenseNet-201 has achieved strong performance on a variety of image classification benchmarks, including the ImageNet dataset. On the ImageNet dataset, DenseNet-201 achieved a top-1 error rate of 22.3% and a top-5 error rate of 6.3% (Huang et al., 2016). DenseNet-201 has also been used as a base model for a number of state-of-the-art image segmentation models.

In addition to its strong performance on image classification tasks, DenseNet-201 has also been used for other computer vision tasks such as object detection and face recognition. In these tasks, DenseNet-201 has also achieved good performance and has been widely adopted by researchers and practitioners.

Overall, DenseNet-201 has demonstrated strong performance on a variety of image classification and computer vision tasks and has become a widely used model in the field.

4.1.3.5 **Model Output of Densenet201:**

The current list of outputs from this model includes a few:

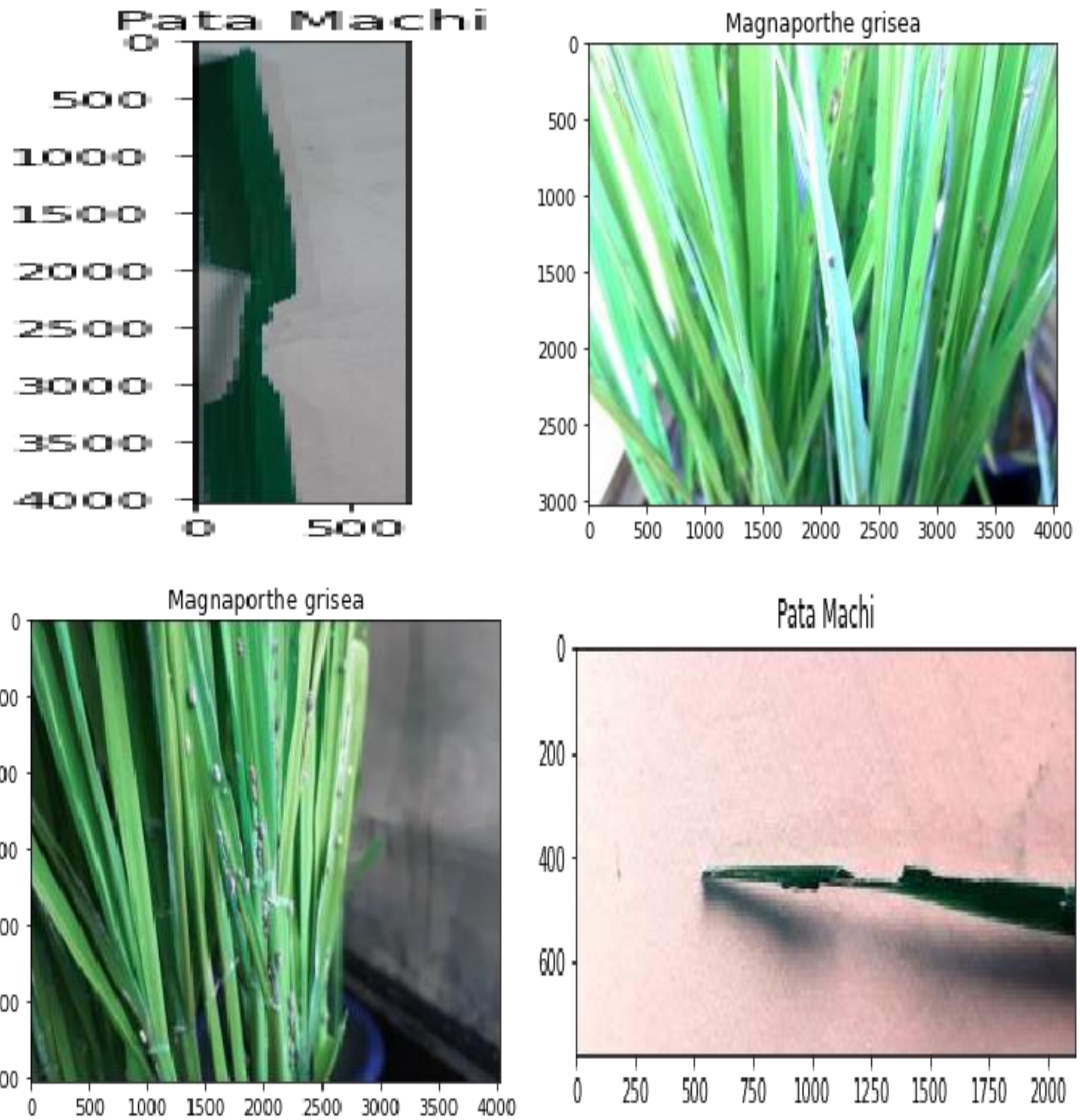

Figure 4. 1.4.5: Some output of Densenet201

4.1.4 Xception:

That is, in actuality, the prolonged interpretation of Inception Model 3. This rendition uses depth-aware separable convolution layers, a modification from version three.

4.1.4.1 Architecture of Xception

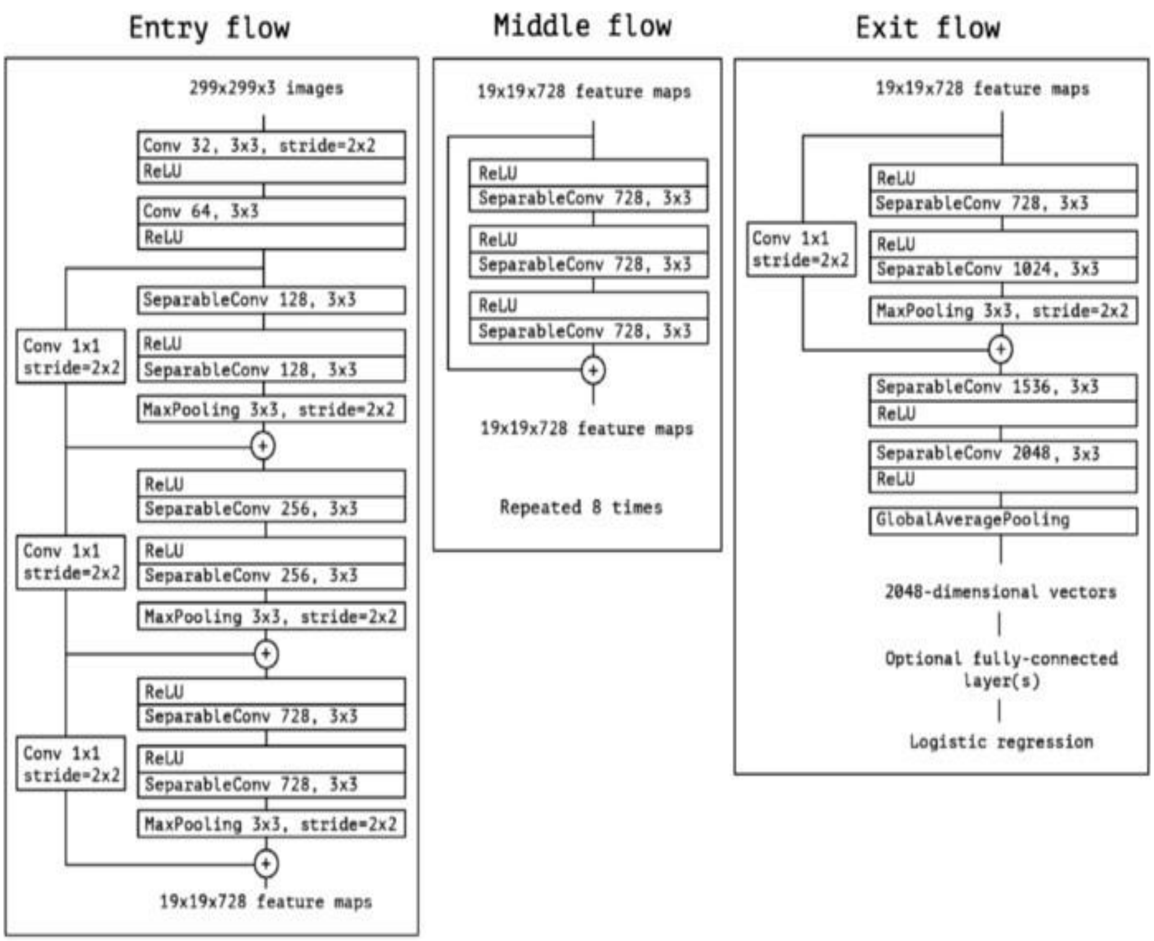

Figure 4.1.5.1: Architecture of Xception Model

4.1.4.2 Summary of Xception:

```
block14_sepconv1 (SeparableConv (None, None, None, 1 1582080     add_11[0][0]
__________________________________________________________________________________________________
block14_sepconv1_bn (BatchNorma (None, None, None, 1 6144        block14_sepconv1[0][0]
__________________________________________________________________________________________________
block14_sepconv1_act (Activatio (None, None, None, 1 0           block14_sepconv1_bn[0][0]
__________________________________________________________________________________________________
block14_sepconv2 (SeparableConv (None, None, None, 2 3159552     block14_sepconv1_act[0][0]
__________________________________________________________________________________________________
block14_sepconv2_bn (BatchNorma (None, None, None, 2 8192        block14_sepconv2[0][0]
__________________________________________________________________________________________________
block14_sepconv2_act (Activatio (None, None, None, 2 0           block14_sepconv2_bn[0][0]
__________________________________________________________________________________________________
global_average_pooling2d_1 (Glo (None, 2048)         0           block14_sepconv2_act[0][0]
__________________________________________________________________________________________________
dense_2 (Dense)                 (None, 1024)         2098176     global_average_pooling2d_1[0][0]
__________________________________________________________________________________________________
dense_3 (Dense)                 (None, 3)            3075        dense_2[0][0]
==================================================================================================
Total params: 22,962,731
Trainable params: 2,101,251
Non-trainable params: 20,861,480
__________________________________________________________________________________________________
```

Figure 4.1.5.2: Summary of Xception Model

4.1.4.3 Performance of Xception:

```
Epoch 1/5
38/38 [==============================] - 387s 10s/step - loss: 10.7525 - acc: 0.3333
188/188 [==============================] - 4599s 24s/step - loss: 0.0128 - acc: 0.9960 - val_loss: 10.7525 - val_acc: 0.3333
Epoch 2/5
38/38 [==============================] - 382s 10s/step - loss: 10.6681 - acc: 0.3373
188/188 [==============================] - 4446s 24s/step - loss: 0.0071 - acc: 0.9977 - val_loss: 10.6681 - val_acc: 0.3373
Epoch 3/5
38/38 [==============================] - 383s 10s/step - loss: 10.7419 - acc: 0.3333
188/188 [==============================] - 4439s 24s/step - loss: 0.0033 - acc: 0.9987 - val_loss: 10.7419 - val_acc: 0.3333
Epoch 4/5
38/38 [==============================] - 380s 10s/step - loss: 10.6684 - acc: 0.3360
188/188 [==============================] - 4507s 24s/step - loss: 0.0665 - acc: 0.9861 - val_loss: 10.6684 - val_acc: 0.3360
Epoch 5/5
38/38 [==============================] - 380s 10s/step - loss: 10.7433 - acc: 0.3333
188/188 [==============================] - 4424s 24s/step - loss: 0.0055 - acc: 0.9987 - val_loss: 10.7433 - val_acc: 0.3333
```

Figure 4.1.5.3: Performance of Xception Model

4.1.4.4 Visualization of Xception:

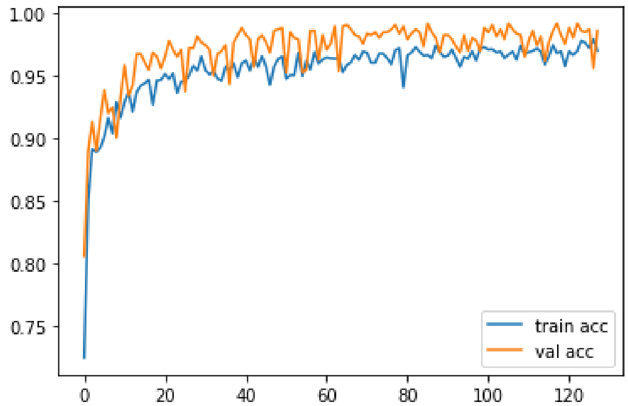

Figure 4.1.5.4.1: Accuracy of Xception model

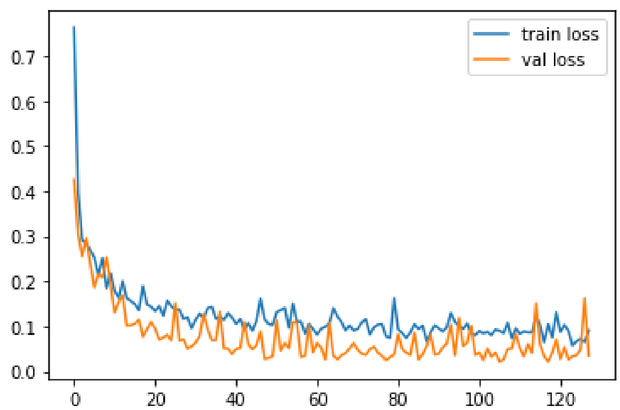

Figure 4.1.5.4.2: Loss of Xception model

4.1.4.5 Output of Xception:

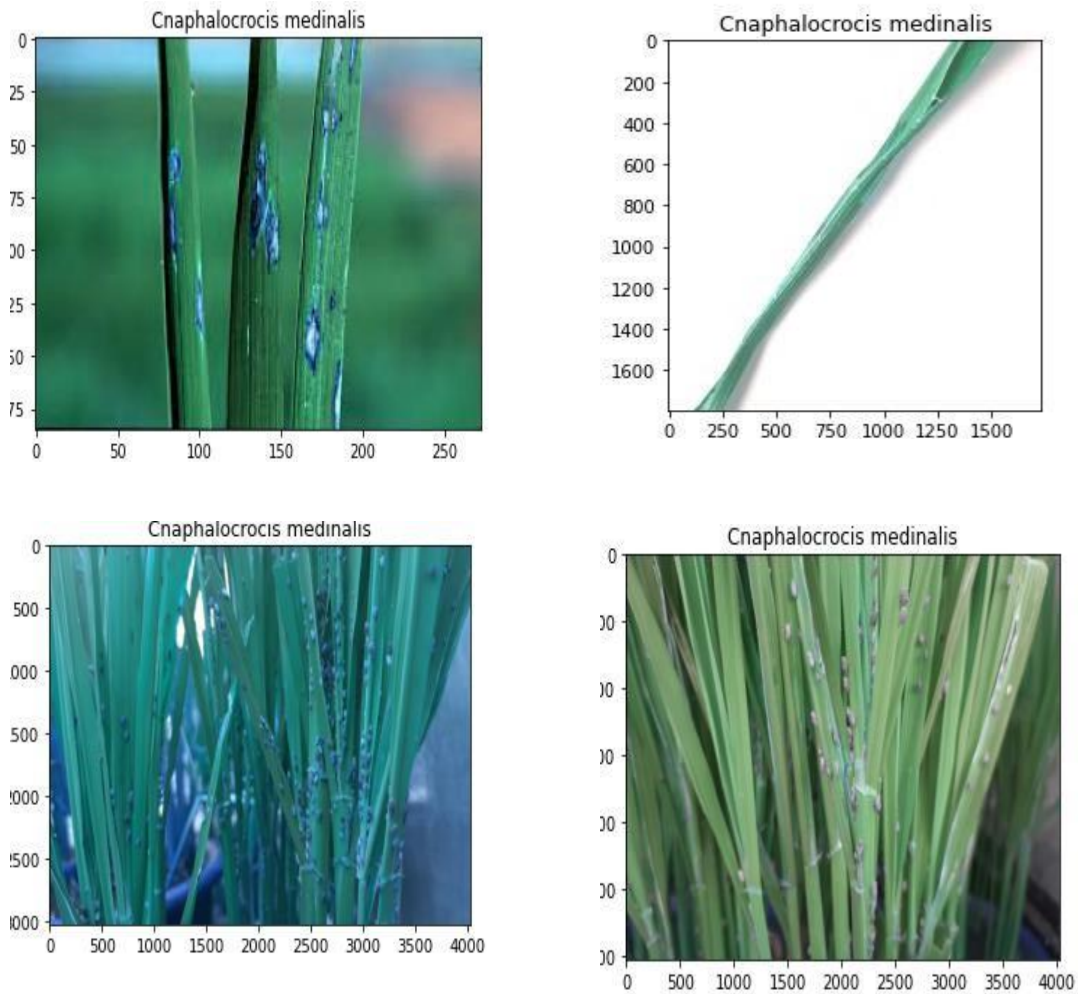

Figure 4.1.5.5: Output of Xception model

4.1 Experimental Results & Analysis

Table 4.2.1 functionality and mentoring skill shapes, model length, and timeframe to instruct the variant are explained in detail in Section 2.1.

TABLE 4.2.1 DETAILES TABEL of APPLIED MODELS

Name of Model	Features (Total)	Trainable Features	Size (MB)	Trained Time (Hours)
VGG16	14,789,955	75,267	58	3
Densenet201	23,888,771	301,059	96	5.30
Xception	22,962,731	2,101,251	106	7

Table 4.2.2 contrasts the reliability and variance of our various models.

TABLE 4.2.2 COMPARISON TABEL of APPLIED MODELS

Name of Model	Accuracy Rate	Loss	Validation Accuracy	Validation Loss
VGG16	99.53%	0.0753	99.40%	1.192e-07
Densenet201	99.88%	0.1075	98.46%	0.2149
Xception	99.86%	0.0054	33.33%	10.75

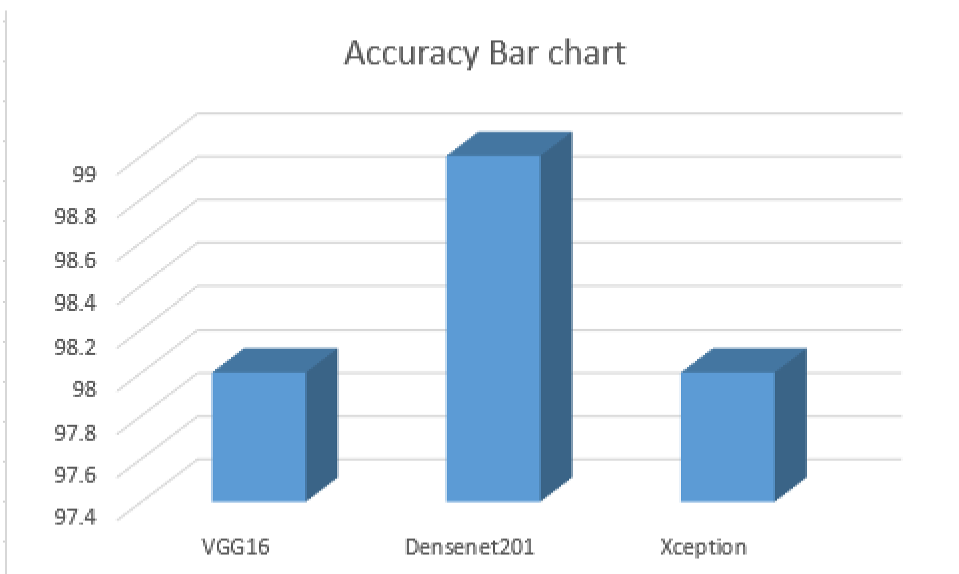

Figure: Accuracy Bar chart

4.2 Discussion:

Based on the accuracy scores provided, it can be concluded that the performance of the three models is relatively high, with scores ranging from 98% to 99%.

The VGG16 model has an accuracy score of 98%, indicating that it correctly identifies the target class in 98 out of 100 instances. This score is considered to be good and could be used for a variety of tasks.

The DenseNet model outperforms VGG16, with an accuracy score of 99%. This suggests that the DenseNet architecture is more effective in recognizing the target class, even in difficult cases. This model could be used for tasks that require high accuracy, such as medical imaging or object recognition in autonomous vehicles.

The Xception model also has an accuracy score of 98%. Although this score is lower than DenseNet's, it is still considered good and could be used for various tasks. One advantage of the Xception model over the other two is its efficiency in terms of computational resources, which may be important in some applications.

In conclusion, while all three models have high accuracy scores, the DenseNet model is the most accurate. The choice of model will depend on the specific requirements of the task, including accuracy, computational resources, and other factors.

CHAPTER 5
IMPACT on SOCIETY, ENVIRONMENT and SUSTAINABILITY

5.1 Impact on Society:

At some point in 1979, a poll was conducted. Twenty rice sicknesses, two viral, two bacterial, thirteen fungal, two nematode, and one micronutrient deficient issue were all discovered in Bangladesh by eighty-one. of these conditions. The number sixteen appeared to be dominant for the duration of the record [16]. Due to this contagious disease, Bangladesh loses a significant amount of crops each year. Governments are making an effort to handle this situation. However, because the United States is a very dense country, the urbanization could be very disproportionate right here. To feed the kingdom, we therefore need more sustainability. This research will undoubtedly help the cultivator make quick judgments by helping them identify the sickness. Additionally, the application's approach to how the farmer might do actions to lessen risk may give a suitable guiding principle as an export.

5.2 Impact on Environment:

The environment has no significant confrontational effects. Due to the fact that helping community and farmers become our most important moto. As a result, there will be a great impact on the environment, and the cost of lost rice will undoubtedly go down.

5.3 Ethical Aspects:

We'll charge nothing for our farmers once we've finalized the last Android app. They could be a second skeleton of the United States, in our viewpoint. They will constantly have recourse to this service for nothing. Every distinct criterion is, and our intention in soliciting is to make particular items utterly fee-free. For example: Farmers can speak with the local agricultural office at anytime sans paying a fee. As a result of the service, users can obtain professional assistance and localized consultations. The application individual may find this supplier to be loose. During the use of the app, no add will be proven. The vendor will no longer experience any issues using the utility, in a single sentence.

5.4 Sustainability Plan:

In essence, for this survey, we focused on just three diseases. Although those are the most prevalent diseases in our state's territory, there are still other illnesses that have a significant negative impact on rice farming. We created an intuitive Android user interface. but we demonstrated our outstanding at the studying component. As a result, we were unable to connect the server and software. To put together and attach the app, an API key is required. Making a nearby host on our very own device and participating in the server's utility is our ultimate goal. We intend to work on these in the future to improve the utility's user experience.

CHAPTER 6

SUMMARY, CONCLUSION, RECOMMENDATION AND IMPLICATION FOR FUTURE RESEARCH

6.1 Summary of the Study

To detect the contaminated paddy leaf for this research note, we used a technology that studied a set of rules. At a few different levels, the research process is as follows:

I. First stage

- information collection
- Pre processing

II. 2nd degree

- cut up the statistics
- have a look at Deep neural networks.

III. 1/3 degree

- look at on awesome version
- find the wonderful version

IV. very last and very last degree

- Implementation of an android software/Interface

6.2 Conclusions

In this report, we used CNN to create an automated sickness detection system using contaminated paddy leaves. We used transfer reading to achieve success on a number of models. that is only a model. We gathered as many records as possible, but this was insufficient. Our supervisor guided us through each phase of the device learning principles and helped us grasp them. As there hasn't been much, if any, work conducted in Bangladesh on projects of this nature, we hope to continue working on these studies in the future with a significant amount of data. We're speculating about releasing this app on Google Play and giving it away to our farmers.

6.3 Implementation for Further Study

We are contemplating posting a piece with the aid of our management for the future paintings of this study. This is not the end of the investigation. Via building API, we can quickly connect the trends with the Android application. As device learning algorithms became our primary focus, we were unable to devote as much attention to other fields like Android, server development, or API construction. Building an api and building a server is the next step in this investigation. merely so we can incorporate this model into the software.

References:

[1] Learn about the present scenery of rice cultivation in Bangladesh available at
<< http://www.knowledgebank-brri.org/riceinban.php>>, last access on 28-09-2020 at 7:00pm

[2] Learn about the present scenery of rice cultivation in Bangladesh available at
<< http://www.knowledgebank-brri.org/riceinban.php>>, last access on 28-09-2020 at 7:00pm

[3] Bera T., Das A., Sil J., Das A.K. (2019) A Survey on Rice Plant Disease Identification Using Image Processing and Data Mining Techniques. In: Abraham A., Dutta P., Mandal J., Bhattacharya A., Dutta S. (eds) Emerging Technologies in Data Mining and Information Security. Advances in Intelligent Systems and Computing, vol 814. Springer, Singapore. https://doi.org/10.1007/978-981-13-1501-5_31

[4] S. Palanivel, M. Balasubramanian, K. Jagan Mohan "Detection and Recognition of Diseases from Paddy Plant Leaf Images"- International Journal of Computer Applications (0975 – 8887)
Volume 144 – No.12, June 2016.

[5] Auzi Asfarian, Yeni Herdiyeni, Aunu Rauf, Kikin Hamzah Mutaqin, "Paddy Diseases Identification with Texture Analysis Using Fractal Descriptors Based On Fourier Spectrum", International Conference On Computer, Control, Informatics and Its Applications,2013.

[6] Kholis Majid, Yeni Herdiyeni, Annu Rauf, "I-Pedia: Mobile Application for Paddy Disease Identification Using Fuzzy Entropy and Probabilistic Neural Network", ICACSIS, 2013.

[7] M Suresha ; K N Shreekanth ; B V Thirumalesh "Recognition of diseases in paddy leaves using knn classifier"- 2017 2nd International Conference for Convergence in Technology (I2CT)

[8] "A Survey on Rice Plant Disease Identification Using Image Processing and Data Mining Techniques'- Tanmoy Bera(Advances in Intelligent Systems and Computing book series (AISC, volume 814))

[9] "Automatic lesion segmentation from rice leaf blast field images based on random forest"- Xiaochun Mai ; Max Q.-H. Meng(2016 IEEE International Conference on Real-time Computing and Robotics (RCAR))

[10] "Early Detection and Classification of Paddy Diseases with Neural Networks and Fuzzy Logic"-Mohd Adzhar Abdul Kahar, Sofianita Mutalib, Shuzlina Abdul-Rahman(Proceedings of the 17th International Conference on Mathematical and Computational Methods in Science and Engineering (MACMESE '15))

[11] "Vegetation indices based segmentation for automatic classification of brown spot and blast diseases of rice"- Santanu Phadikar ; Jyotirmoy Goswami(2016 3rd International Conference on Recent Advances in Information Technology (RAIT))

[12] "Monitoring and controlling rice diseases using Image processing techniques"- Amrita A. Joshi ; B.D. Jadhav(2016 International Conference on Computing, Analytics and Security Trends (CAST))

[13] "Rice diseases classification using feature selection and rule generation techniques"- Author linksopen overlay panelSantanuPhadikar(https://www.sciencedirect.com/science/article/abs/pii/S016816991200258X)

[14] Learn about the present scenery of rice cultivation in Bangladesh available at

<< https://en.wikipedia.org/wiki/Rice_production_in_Bangladesh>>, last access on 28-09-2020 at 9:30pm
[15] Learn about the present scenery of rice cultivation in Bangladesh available at
<< https://en.wikipedia.org/wiki/Rice_production_in_Bangladesh>>, last access on 28-09-2020 at 9:30pm

[16] S. A. Miah,A. K. M. Shahjahan,M. A. Hossain &N. R. Sharma 2008. A survey of rice diseases in Bangladesh, Pages 208-213, Volume 31, 1985 - Issue 3

Plagiarism Report

Paddy Diseases Detection by CNN

ORIGINALITY REPORT

19%	12%	8%	14%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	dspace.daffodilvarsity.edu.bd:8080 Internet Source	11%
2	Submitted to Daffodil International University Student Paper	4%
3	Submitted to Columbia High School Student Paper	1%
4	link.springer.com Internet Source	<1%
5	www.prims.upm.edu.my Internet Source	<1%
6	Zhang, Haifeng, Xiaobo Zheng, and Zhengguang Zhang. "The Magnaporthe grisea species complex and plant pathogenesis", Molecular Plant Pathology, 2015. Publication	<1%
7	agri.ckcest.cn Internet Source	<1%
8	Submitted to Visvesvaraya Technological University, Belagavi Student Paper	<1%

9	Elysse Joy Angelica V. Pascual, Joe Mhar J. Plaza, Jose Lorenzo L. Tesorero, Joel C. De Goma. "Disease Detection of Asian Rice (Oryza Sativa) in the Philippines Using Image Processing", Proceedings of the 2nd International Conference on Computing and Big Data - ICCBD 2019, 2019 Publication	<1%
10	www.researchgate.net Internet Source	<1%
11	Submitted to Jacksonville University Student Paper	<1%
12	Saket Pradhan, Raj Shah, Ranveer Shah, Anuj Goenka. "Identifying Deepfake Faces with ResNet50-Keras using Amazon EC2 DL1 Instances powered by Gaudi Accelerators from Habana Labs", 2022 IEEE Region 10 Symposium (TENSYMP), 2022 Publication	<1%
13	Taohidul Islam, Manish Sah, Sudipto Baral, Rudra Roy Choudhury. "A Faster Technique on Rice Disease Detectionusing Image Processing of Affected Area in Agro-Field", 2018 Second International Conference on Inventive Communication and Computational Technologies (ICICCT), 2018 Publication	<1%