AN APPROACH EMPLOYING DEEP LEARNING TO ASSESS THE EXTERNAL CONDITION OF RICE SEEDS

BY

MD. ISMAIL HOSSEN SARKER
ID: 192-15-2864

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

Supervised By

Mohammad Jahangir Alam
Sr. Lecturer
Department of CSE
Daffodil International University

Co-Supervised By

Md. Sabab Zulfiker
Sr. Lecturer
Department of CSE
Daffodil International University

DAFFODIL INTERNATIONAL UNIVERSITY

DHAKA, BANGLADESH

AUGUST 2023

APPROVAL

This Project titled **"Classification of an Approach Employing Deep Learning to Assess the External Condition of Rice Seeds"**, submitted by Md. Ismail Hossen Sarkar and Id: 192-15-2864 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 2 August 2023.

BOARD OF EXAMINERS

____

Mr. Narayan Ranjan Chakraborty (NRC) **Chairman**
Associate Professor and Associate Head
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

____

Tania Khatun (TK) **Internal Examiner**
Assistant Professor
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

____

Md. Sabab Zulfiker (MSZ) **Internal Examiner**
Senior Lecturer
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

____

Dr. Md. Zulfiker Mahmud (ZM) **External Examiner**
Associate Professor
Department of Computer Science and Engineering
Jagannath University

DECLARATION

I hereby declare that, this project has been done by me under the supervision of **Mohammad Jahangir Alam, Senior Lecturer, Department of CSE,** Daffodil International University. I also declare that neither this project nor any part of this project has been submitted elsewhere for award of any degree or diploma.

Supervised by:

____

Mohammad Jahangir Alam
Senior Lecturer
Department of CSE
Daffodil International University

Co-Supervised by:

____

Md. Sabab Zulfiker
Senior Lecturer
Department of CSE
Daffodil International University

Submitted by:

____

Md. Ismail Hossen Sarkar
ID: 192-15-2864
Department of CSE
Daffodil International University

ACKNOWLEDGEMENT

First, I express my heartiest thanks and gratefulness to almighty God for His divine blessing makes me possible to complete the final year project successfully.

I really grateful and wish my profound indebtedness to **Mohammad Jahangir Alam, Senior Lecturer**, Department of CSE Daffodil International University, Dhaka. Deep Knowledge & keen interest of my supervisor in the field of "*Machine Learning*" to carry out this project. His endless patience, scholarly guidance, continual encouragement, constant and energetic supervision, constructive criticism, valuable advice, reading many inferior drafts and correcting them at all stage have made it possible to complete this project.

I would like to express my heartiest gratitude to **Dr. Touhid Bhuiyan**, Professor and Head**,** Department of CSE, for his kind help to finish my project and also to other faculty member and the staff of CSE department of Daffodil International University.

I would like to thank my entire course mate in Daffodil International University, who took part in this discuss while completing the course work.

Finally, I must acknowledge with due respect the constant support and patients of my parents.

ABSTRACT

Rice is one of the most important foods in the world, and in some places, it is the main food. It is the biggest food on the planet, and it is now also the biggest growing food in Bangladesh. Most of the time, rice is infected by the fungus. Finding the right diseases and finding solutions is a big part of how our farmers make money, how much food we need, and how safe our food supply is. At the moment, image processing is mostly used to find out what diseases people have. So, in the deep learning field, we used six different designs of convolutional neural networks (CNN). MobileNetV2, VGG16, ResNet50, VGG19, Efficient NET V2L, and ConvNeXtXLarge are the designs. From these architectures, MobileNetV2 is one of the most famous CNN architectures and gave the best results from what we saw. We used the rice image collection for this method. Here, we put our data into three different groups. Two of the groups are made up of sick rice, and the third is made up of fresh rice. This work can help find out what's wrong with rice. Farmers and researchers will find it easy to find rice diseases with this system. With this method, it is much easier to find and classify rice diseases than to find and group them by hand, and it doesn't take long to find rice diseases. This will help people who are going to work for CNN.

TABLE OF CONTENTS

CONTENTS	PAGE NO

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1
Introduction

1.1 Introduction

My mission is to develop a deep learning model that can be used to help farmers and create a platform on which farmers can simply upload images of rice seeds. In order to determine whether or not a given seed will grow into a rice tree, my Deep Learning Model will extract information and provide us with that knowledge. I already have classes that are predefined, and these classes are Healthy and Unhealthy conditions. Diseases not only result in a loss of productivity but also in contamination of the natural environment. Rice is susceptible to a wide variety of illnesses, many of which have the potential to severely reduce crop productivity. Both the Windows operating system and the programming language known as Python were necessary components for the successful completion of my mission. In addition to that, it makes use of a variety of packages, such as Keras, TensorFlow, NumPy, and a great deal of others. I was successful in finishing all of the programming chores with the assistance of Google Collaboratory, which is an online environment hosted in the cloud. In order to develop the classification model, I relied on a Tesla P100 GPU. I have made advantage of the architecture's internal layers, which are known as the Pooling Layer, the Fully Linked Layer, the SoftMax Layer, and the Output Layers. VGG16, VGG19, Resnet50, and MobiNet V2 are the models that I have used in my work. I achieved an accuracy of 85.71% on the validation dataset while using VGG16 with a conventional architecture for up to 150 epochs, and when I pre-trained the model, it performed even better.

By using transfer learning, I am able to achieve an accuracy of one hundred percent for the training dataset. In addition to this, the confusion matrix that I received was 95.24%. The conventional model made inaccurate predictions for ten frames, but transfer learning was able to bring that number down to six frames. I was able to get an accuracy of 80.42% for the validation dataset while running VGG19 on the standard architecture for up to 150 iterations. In addition, there were just six epochs used in the development of this exam. In addition, the accuracy of the validation was 95.24 percent. When I take a look at the confusion matrix, I can see that the standard model made an inaccurate prediction for 23 photographs, but transfer learning brought that number down to 5.

The ultimate accuracy of the tests for the conventional model was found to be 81.74%, while the accuracy for transfer learning was found to be 96.03%. I found that after 20 iterations of the conventional MobileNet V2 design, the validation data set yielded an accuracy that was decreasingly closer to 50.79 percent. Yet, you were able to achieve an accuracy of one hundred percent with the train data set using just five epochs of transfer learning. By transfer learning, I was able to achieve a validation accuracy of 92.59%. The accuracy of the exam increased from 50.79% to 94.44% because to transfer learning. The ReseNet50 model is currently my most accurate one. In this case, by using the conventional architectural approach, I was able to get an accuracy of 90.48 percent for the validation data set throughout the course of the first 150 iterations. On the other hand, if I make use of pre-trained transfer learning, I am able to attain an accuracy rate of one hundred percent using the training data set. In terms of accurate validation, I received a score of 97%. When I take a look at the confusion matrix, I can see that the conventional model produced 9 incorrect predictions, but thanks to transfer learning, we've been able to bring that number down to 5. Listed as a dangerous class with a hundred percent correction. The accuracy of tests based on the conventional model was found to be 92.85%, whereas the accuracy of tests based on transfer learning was found to be 96.03%. The agriculture sector in Bangladesh may benefit from the implementation of a project of this kind. Since the vast majority of Bangladesh's farmers lack education, they are unable to tell the difference between good and harmful seeds. As a direct consequence of this, they are unable to harvest their crops. I anticipate that after farmers have adopted my technology, they will be able to accurately evaluate their current condition of health. Engineers may also create devices for use with the Internet of Things. It takes care of removing damaged seeds from the fields by itself. The idea that I have will act as the basis for the Internet of Things devices. Eventually, a health app will be developed that, by using my database and model, will be able to ascertain an individual's current state of health.

1.2 Motivation

The overall process of my approach for rice disease detection is presented as follows: initially, rice disease image samples are collected and labeled based on the knowledge of field experts; Then, image-processing techniques, including image resizing, image sharpening and image edge filling, etc., are performed on the acquired images, and new sample images are generated to enrich the dataset using data augmentation methods, for example, rotation and translation are used to generate augmented datasets. ; After that, sample images are input to the proposed method for model training, and then the trained model is applied for class prediction of rice disease images. Thus, the final detection results are obtained and can be used to update the sample knowledge library. The details of these phases are illustrated in the next-quant section.

1.3 Rationale of the Study

- To identify rice seeds that are unhealthy.
- Separate rice seeds that are harmful from rice seeds that are good.
- Image processing of the health of rice seedlings in real-time
- Possible inclusion in the IoT Robot in order for it to separate harmful rice seeds from good rice seeds.

1.4 Research Questions

- Is the purpose of this research to educate farmers who may not know the difference between safe and dangerous seeds?
- Would it be beneficial to Engineers and enable the creation of Internet of Things devices that can automatically remove diseased seeds from cultivable lands?
- How will the Deep Learning system make use of this model?
- Which algorithm did you end up using?
- How is precision guaranteed and maintained?
- Where did the images come from in the data sets?
- What would be used in order to capture photographs?

Do I anticipate a higher degree of accuracy?

1.5 Research Objectives

The goal of this research is to determine which kind of rice seeds are harmful. Separation is made between unhealthy paddy and healthy paddy. imaging of the rice seedlings' state of health in real-time The Internet of Things (IoT) robot that is used to distinguish potentially dangerous rice seeds from healthy rice seeds might use this.

1.6 Expected Outcome

If I carry out my study with a high degree of precision, I will be able to identify rice seeds that are not healthy. Separate the rice seeds that are harmful from the rice seeds that are healthy. I was able to carry out image processing of the health of rice seedlings in real time.

1.7 Report Layout Chapter

The paper's organization is provided below:

i) Background
ii) Research Methodology
iii) Experimental Results and Discussion
iv) Summary, conclusion, Recommendation, and implication for future Research
v) Reference

CHAPTER 2

Background Study

Related Works:

In the past, various researches were conducted in the field of image processing Analyzed in this section. Different methods and methods for identifying rice leaves Diseases using image processing are reviewed in this literature review. Considering Multiple algorithmic models, this paper also considers the implications of machine learning and deep learning methods.

Yang Lu et.al. Committed to automated documentation or analysis of rice diseases. Deep learning is hot current pattern recognition and machine learning research topics. You can solve these problems effectively in Plant Pathology; I suggest a different strategy for vascular disease based on convolution neural network technology. More than 500 photographs and extracts have been taken from pink field leaves from the field, CNN was trained to recognize 10 common diseases. Under the single-track correction technique, The proposed CNN model achieves 95.48% accuracy. This approach is far superior to the traditional education model. The results of rice price index analysis show the feasibility or usefulness of this method [10].

S. Ramesh et al. In agriculture, one of the most novel study topics is disease recognition or classification plant leaflets. Agricultural disease detection will be reduced through the use of image processing technology to save the lives of farmers their agricultural produce. A neural network based on Jaya algorithm is done leaf disease notice or order proposed. For photography purposes, light quantity images, rice bacteria, wool and blast are taken directly from the farm. In preprocessing, to remove their origin, RGB images are converted to HSV images and binary images are extracted based on hue and saturation components. Divide patients and non-infected [16].

In this article, I propose a system for image capture plant gram dataset preparation. The second is a convolutional Neural Networks, which are used to classify tensor flow diseases and pesticide determination technologies.

The two steps I use in the system are Java web services and Android applications with deep learning. I used Convolutional Neural Networks at 5th, 4th and 3rd levels to train my Android models and applications become user-friendly with JWS [18].

Wanjie Liang1 was to study the distribution of affected areas, their proportions and their origins using Deep Neural Network with Jaya Optimization Algorithm for advanced Diarrhea and Disease Control for Disease Classification, Probable management and Suggested Field Pest Monitoring according to data from the Wheat Growth Institute, farmers lose up to 37% of their crops to pests and diseases. every year [17].

Sladojevic and colleagues [8] aimed to detect plant diseases using deep learning techniques that will help farmers quick and easy detection of diseases which will consequently enable farmers should take appropriate steps at the initial stage. They used 2589 original images and 30880 images for testing train their models using the caffe deep learning framework [9]. A predictive evaluation is to achieve a high accuracy model, the authors used a 10-fold cross validation technique in their dataset. The prediction accuracy of this model 96.77%.

Depending on the extracted percentage of RGB value of damaged area of rice leaves using image processing, a the model was developed in [10] to classify this disease. RGB the percentages were finally fed to the naïve bayes classifier divide diseases into three disease classes: bacteria leaf blight, rice blast and brown spot. Its accuracy the disease classification model is over 89%.

A high accuracy paper was found [11] where a plant disease detection model was developed using CNN. This The model can detect 13 different types of plant diseases. the final accuracy achieved from this model is 96.3%.

In another study [12], the affected parts were isolated from rice leaf surface area using K-means clustering and models then SVM was trained using color, texture and shape hierarchical features.

Maniath et al. Used random forest, an ensemble learning method, classifying into healthy and diseased leaves [13]. For to extract the features of an image, the author used histogram of Oriented Gradient (HOG). Their work made a demand accuracy is 92.33%.

There were image processing and machine learning techniques also used for rice identification and classification [14]. Plant Disease The authors of this paper used K-means clustering for division of diseased areas of rice leaves and Support Vector Machine (SVM) for classification. They are achieved a final accuracy in training, 93.33% and 73.33% test dataset respectively. The same dataset was also used in my work but my method resulted in higher accuracy of bot on training and testing datasets.

CHAPTER 3

Research Methodology

3.1 Methodology

In this section of the report, I will discuss the processes and methods that I followed in order to complete this research as well as the methodology that I employed for my own study. In the first place, I have collected real-time data on rice seeds from Kaggle, which I also acquired by scraping other websites, and I have taken pictures in rice fields. This dataset contains information on both healthy seeds and diseased seeds as separate categories of seed classes. Following that, I established certain protocols for preparing the data, and then I divided it into test and train sets. As I was training my datasets, I made use of CNN-based architectures such as MobileNetV2, ResNet50, VGG16, and VGG19.

3.2 Instruments

For us to be successful in accomplishing this work, I needed to use the Windows operating system and the Python programming language. In addition, numerous other packages are used, including as Keras, Tensorflow, NumPy, and a great deal of others. I was able to complete all of the programming tasks with the support of Google Colaboratory, an online platform that operates in the cloud. A Tesla P100 GPU was used throughout the process of the categorization model's development.

3.3 Data Collection

I have collected photographs depicting the two unique characteristics of rice seeds. The majority of the photos came from the Kaggle repository; however, I also included images that were scraped from the internet and pictures that were taken using a phone. In all, there are 1268 photographs included inside this collection. The total number of photos in the healthy category is 646, while the number of pictures in the unhealthy category is 622. Seventy-five percent of the photographs from each categorization were used for the purpose of training, 15% for validation, and 10% for testing. The following infographic offers a visual illustration of the sample data set pertaining to rice seeds:

Figure 3.1: Sample Dataset Of Rice Seeds

Figure 3.2: Dataset Classe

3.4 Data Analysis

Classes	Quantity
Healthy (train)	486
Unhealthy (train)	467
Healthy (test)	64
Unhealthy (test)	62
Healthy (validation)	96
Unhealthy (validation)	93

Table (3.1) Dataset Splitting Ratio

3.5 Proposed Methodology

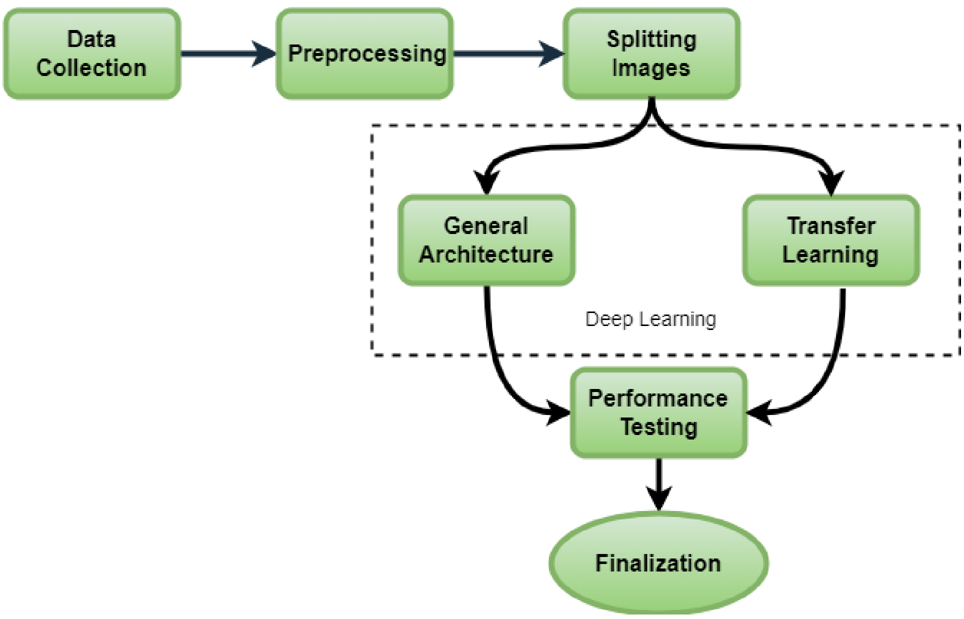

Figure 3.3: Proposed methodology

3.6 Data Preprocessing

Due to the fact that not all of the picture data resolutions in my collection are the same, I had to restrict the size of the input images to 224 by 224 pixels and utilise VGG16 preprocess input in order to construct the batches. Following that, the images were processed using the CNN architectures that were tailored specifically for this activity and made use of the RGB colour space. The use of CNN can become more accurate as a result of the contribution that this colour makes to the identification of features.

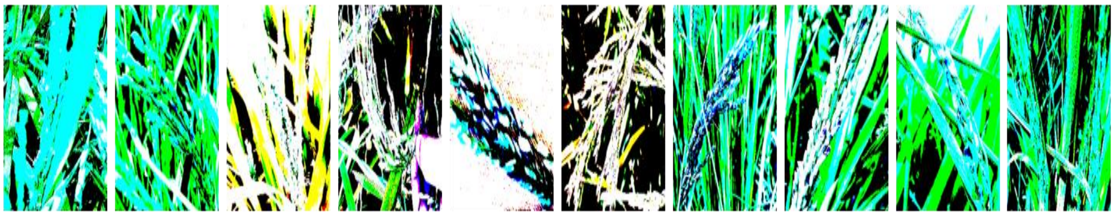

Figure 3.4: Preprocessed images

3.7 Internal layers of the architecture

Pooling Layer

The pooling layer has a diminishing effect on the total spatial volume of the picture. It is placed in between two convolution layers, and maximum pooling is the sole method that can be utilized to reduce the amount of space that is used up by the input image. One possible location for its use is in the middle of two convolution layers. The fact that the pooling layer does not have any parameters is fairly well known; nevertheless, it does contain two hyperparameters that go by the names filter and stride.

Layer That Is Entirely Interconnected

Weight, bias, and neurons are the components that make up the totally linked layer. Neurons on one layer, as well as neuronal connections on another layer, are linked to this layer through this layer. In addition to that, it may be used in a wide range of different ways to the process of training photographs.

SoftMax Layer

The last layer of the CNN algorithm is known as the Softmax layer or the logistic layer. Moreover, binary logistic classification and Softmax multi-classification are also used in this analysis.

Layer For Output

The output level incorporates labels that are present in the dataset we're working with.

3.8 Model Training

I have used CNN-based architecture for my dataset. Nevertheless, I did not use the pre-trained layers in this project. There are no weights included in any of the models. The following list displays my previously owned models:

1. ResNet50
2. MobileNetV2
3. VGG19
4. VGG 16
5. Efficient NET V2L
6. ConvNeXtXLarge
7. General CNN

3.8.1 CNN

Image classification, object recognition, and picture segmentation are examples of the types of computer vision tasks that often make use of CNN, which is an acronym that stands for "Convolutional Neural Network," which is a sort of deep learning algorithm.

A CNN works by feeding the picture it receives as input through a sequence of convolutional layers. These layers each apply a set of filters to the image in order to extract information from it. These filters are taught to detect various patterns and forms included within the picture, such as edges, corners, and textures, and then use this knowledge to further refine their results.

When the output of the convolutional layers has been processed by one or more pooling layers, the spatial dimensions of the feature maps are shrunk but the most essential information is kept intact.

After all is said and done, the output of the pooling layers is sent via one or more fully connected layers, which map the collected features to the appropriate output class or regression result.

CNNs have a number of benefits, one of which is the capacity to automatically learn features from raw data. This eliminates the need for hand-crafted feature engineering, which is one of the advantages of CNNs. They have reached the pinnacle of performance in a variety of computer vision tasks, and their applications may be found in both commercial and academic settings.

3.8.2 ResNet 50

Microsoft first presented their version of a certain kind of neural network known as a Residual Network in the year 2015. ResNet is an abbreviation for the term "Residual Network." The ResNet50 network serves as an illustration of a potential component of the ResNet network architecture. The technique that ResNet proposes as a way to address the problem of deterioration is referred to as "residual mapping," and its name comes from the word "residual mapping." It has around twenty-three million trainable parameters and is composed of fifty layers in total. In contrast to the traditional construction of a DCNN, this one uses a method called global average pooling rather than layers that are eventually connected. Its weight is much lowered despite the fact that it is considerably more comprehensive than the various designs that were used in this investigation.

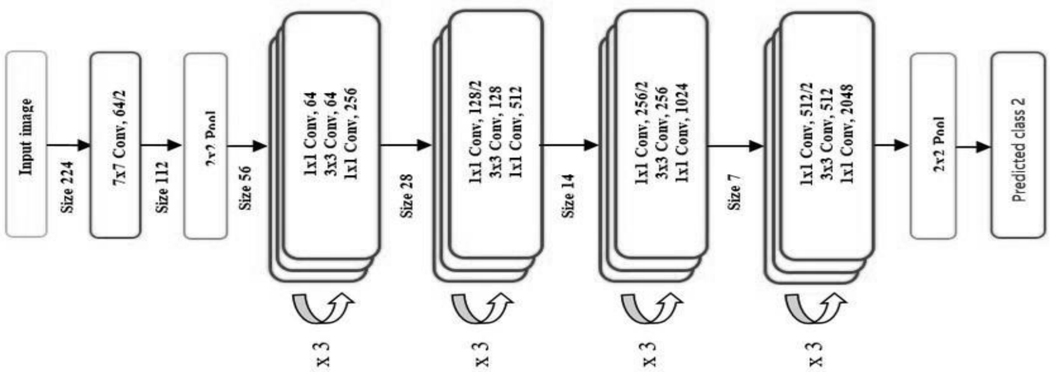

Figure 3.5: Architecture of ResNet50

3.8.3 MobileNetV2

MobileNetV2 is a neural network architecture that is constantly developing and is based on an inverted residual structure. Its primary goal is to perform well on mobile devices. An initial entire convolution level with 32 filters is included into the architecture of MobileNetV2, which is then followed by 19 barrier layers that are now in use.

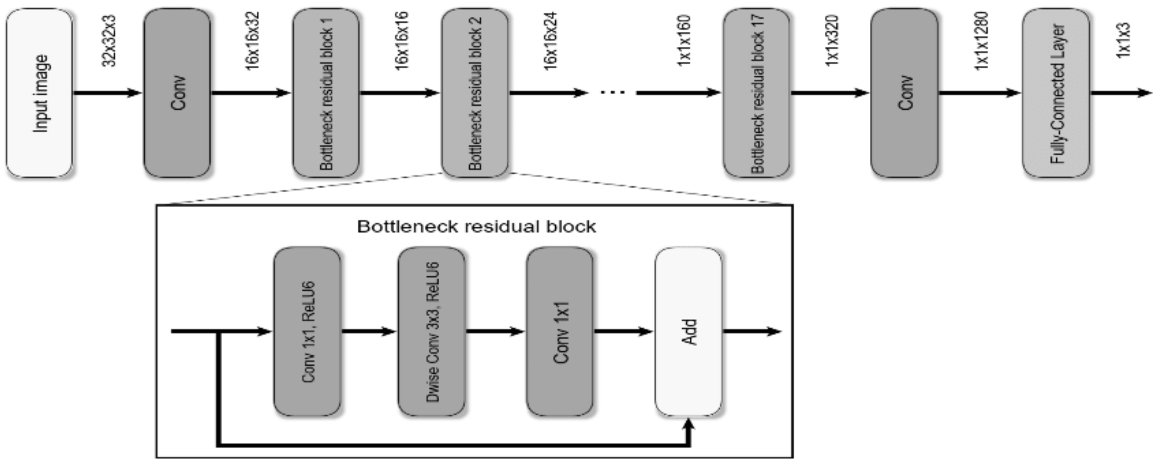

Figure 3.6: Architecture of Mobilenet V2

3.8.4 VGG16

The Visual Geometry Group (VGG) network includes VGG16 as one of its nodes. The VGG16 network is comprised of a total of 17 layers: 16 convolutional layers, 3 fully connected layers, and 5 max-pooling layers. At the competition held by the ILSVRC, it earned a second-place finish. It offers more than 138 million distinct settings for each parameter. By using the dropout and ReLU activation functions, it helps to reduce the amount of generalization error that occurs across all fully linked layers. Also, the output of the model makes use of the softmax function in certain instances. This approach instead makes use of a number of filters with a dimension of 3 by 3, which eliminates the need for large-scale kernels or filters. Also, the little size of these kernels makes it possible to do sophisticated feature extraction at a lower cost.

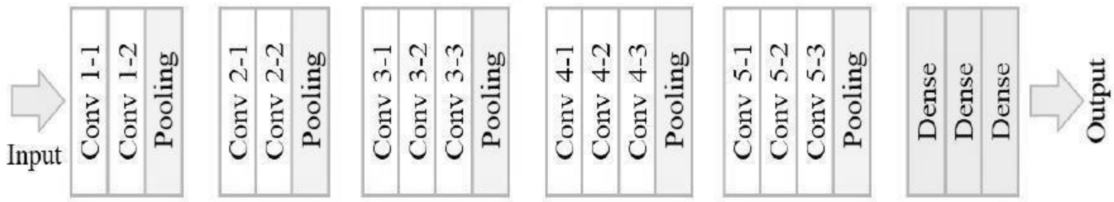

Figure 3.7: Architecture of VGG16

3.8.5 VGG19

VGG19 is the name of one of the many variations of the VGG16 gene. It contains 19 layers, which is an increase from the usual 16 layers (1 SoftMax layer, three fully connected layers, 16 convolution layers, and 5 MaxPool layers). The input for this network was an RGB image that had a constant size of 224 pixels by 224 pixels. Once each pixel had been deleted at the beginning of the preprocessing phase, the mean RGB value of each pixel was calculated. This was carried out for the whole of the training session. Kernels that are three pixels by three pixels in size and have a stride size of one pixel may be used to convey the overall notion of the image. The objective of the method that is known as "spatial padding" is to preserve the image's spatial resolution. The max pooling algorithm was applied to a pixel window that was two by two, and the stride parameter was set at 2. Following this step, a Rectified linear unit, also known as a ReLu, was applied to the model in order to add non-linearity into it. This allowed the model to be categorized with more precision and sped up the computing process. Implementation of three ultimately connected layers, the first two of which had a size of 4096, a layer with 1000 channels for 1000-way classification, and a layer with a softmax function as the last layer in the chain of implementation. Each of the levels has a complete connection to the next.

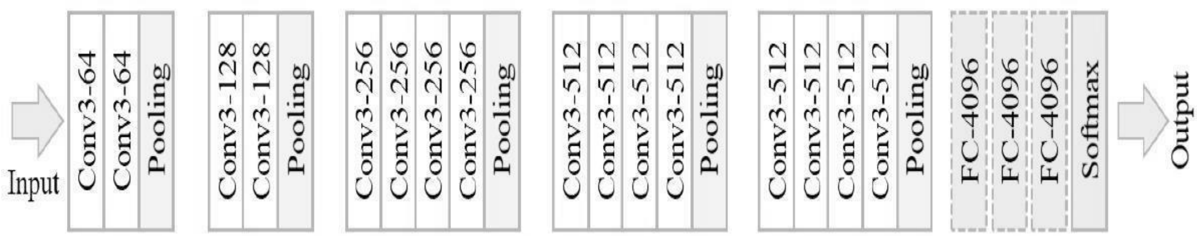

Figure 3.8: Architecture of VGG19

I utilized the weight of each architecture, which was either 'imagenet' or 'none,' and any non-trainable parameters were changed to false at the end of the process. All of the models have been used both for transfer learning and the conventional technique.

3.8.6 EfficientNetV2-L

EfficientNetV2-L is a convolutional neural network (CNN) model that is a member of the EfficientNetV2 family. The EfficientNetV2 family is a series of convolutional neural network (CNN) models that are recognized for their efficiency and good performance on a variety of computer vision tasks. Within the EfficientNetV2 series, the version known as "EfficientNetV2-L" is precisely one of the bigger available options.

A compound scaling strategy is used in the construction of the EfficientNetV2 models. This method concurrently improves the network's depth, breadth, and resolution. This strategy makes it possible for the models to reach a better level of accuracy while still preserving efficiency in terms of the computing resources that are needed for training and inference.

Due to the fact that it is one of the bigger versions, the EfficientNetV2-L has a wider depth and breadth in comparison to the other varieties. Because of this expanded capacity, it is now able to extract more intricate patterns and characteristics from pictures, which may contribute to an improvement in its overall performance on demanding tasks such as object identification, image categorization, and object detection.

3.8.7 ConvNeXt-XLarge

It is Yet Another CNN Architecture suited for Computer Vision jobs ConvNeXt-XLarge is yet another CNN architecture suited for computer vision jobs. It is an extension of the architecture known as ConvNeXt, which tries to improve the representational capability of conventional CNNs by using the idea of grouped convolutions. This modification was developed by Google.

ConvNeXt-XLarge applies the grouped convolutions in parallel, which enables the model to capture different and complementing characteristics by evaluating numerous transformation routes inside each layer. This is accomplished by applying the grouped convolutions in ConvNeXt-XLarge. Because of this parallelism, the model's expressive capacity is increased, and it also improves the model's capability to learn discriminative features based on input pictures.

The term "XLarge" denotes that the ConvNeXt-XLarge architecture is an expanded version of the original ConvNeXt design. It often comprises of a greater number of layers and parameters, which enables it to capture patterns that are more detailed and to deliver better accuracy while working on difficult computer vision jobs.

Both EfficientNetV2-L and ConvNeXt-XLarge are examples of recent developments in deep learning architectures for computer vision. These designs seek to achieve an equilibrium between the levels of model efficiency and performance that they provide. Because of their bigger sizes, they are able to accomplish more complicated jobs; nevertheless, it is crucial to note that the particulars of these models may change depending on the implementation and research developments that occur after September 2021, which is the point at which my expertise will cease to be relevant.

3.9 Performance Testing

I have taken the help of different parameters to test the performance of the models. Such as Accuracy Precision Recall F1 Score etc.

Confusion Matrix; A confusion matrix is a table that summarizes the performance of a classification model by displaying the number of true positives, true negatives, false positives, and false negatives. In other words, it counts the number of cases in which the model got something right and got something wrong. It is a practical instrument for evaluating the efficacy of a classification model and identifying weak spots that need to be strengthened.

F1 Score: The F1 score is a statistic that is often used for the purpose of assessing the effectiveness of a binary classification model. It gives a single measure of the model's performance that takes into account both false positives and false negatives, and it is the harmonic mean of accuracy and recall.

Precision: Precision is a statistic that represents the fraction of true positives, or samples that have been accurately recognized, among all of the positive predictions that a classification model has produced. It is a metric that assesses how precise the model's positive predictions are.

Recall: The term "recall" refers to a statistic that calculates the percentage of "true positives," or samples that have been properly recognized, relative to the total number of "real positives" in a dataset. It is a measurement of how effectively the model can recognize positive samples inside the dataset. The higher it is, the better.

Validation: The term "validation" refers to the process of testing the effectiveness of a machine learning model by applying it to a dataset that is distinct from the one that was used for training purposes. This is done to prevent the model from being too accurate by checking to see whether it has learnt any generalizable patterns from the data.

Accuracy: Accuracy is a statistic that quantifies the percentage of true predictions produced by a classification model relative to the total number of predictions made. Accuracy is also known as predictive accuracy. It is a measurement of how effectively the model can accurately categorize the samples that are included in the dataset. On the other hand, accuracy might be deceiving in datasets that are unbalanced, meaning that one type is much more common than the others.

CHAPTER 4

Experimental Results and Discussion

4.1 Results

At this point, I have discussed the evaluation of the performance of my models, as well as a number of parameters and the degree of accuracy; in addition, I have shown plot diagrams for each model. The results were confirmed with the help of ResNet50, MobileNetV4, VGG19, and VGG16 by applying the relevant parameters. I have put each model through its paces in order to discover which of the models performs the best in both the training and the testing stages. In order to do this, I have given each model its own unique set of challenges.

4.1.1 Experimental Result & Analysis for VGG16 Architecture

Using the validation dataset in VGG16 with the conventional architecture and up to 150 epochs, I achieved an accuracy of 85.71 Figure (4.1), however with the pre-trained transfer learning, I achieved an accuracy of 100% with the training dataset. In addition, there were just seven epochs used in order to accomplish that test. In terms of the correctness of the validation, I obtained 95.24% Figure (4.2). When I take a look at the confusion matrix, I can see that the conventional model made incorrect predictions for 10 photos, but after transfer learning, that number dropped to six. In addition, there is an obvious disparity between the normal model and the transfer learning table in terms of the F1 score, the accuracy, and the recall (4.1). In conclusion, the accuracy of the testing was 92.06% when using the conventional model, while it was 95.24% when utilizing transfer learning.

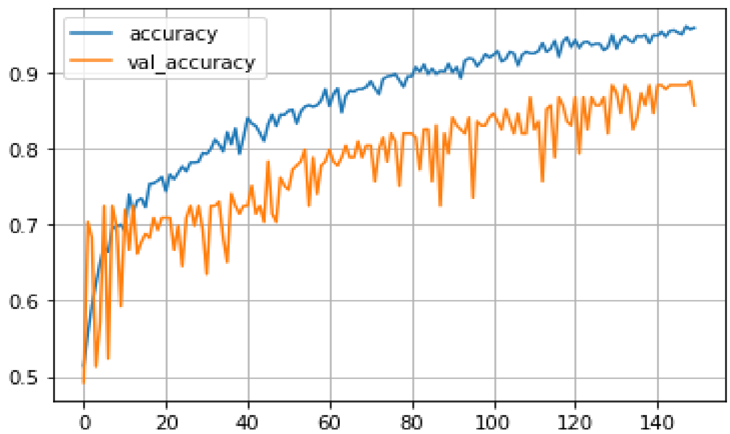

Figure 4.1: Accuracy of Train and validation data(VGG16) (Normal)

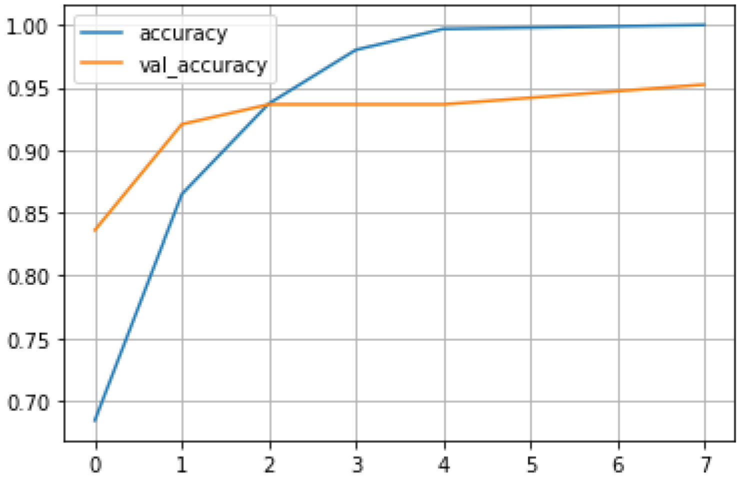

Figure 4.2: Accuracy for Train and validation(VGG16) (Transfer Learning)

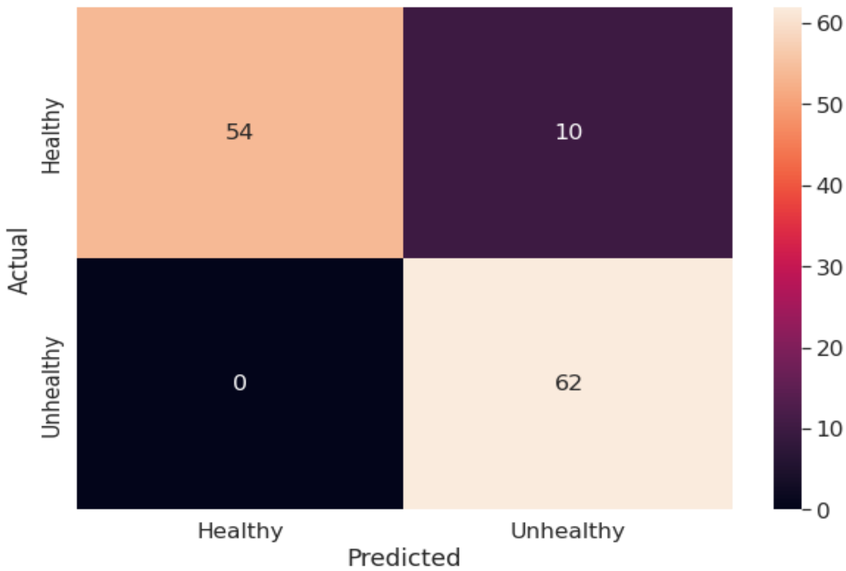

Figure 4.3: Confusion Matrix of VGG16 (Normal)

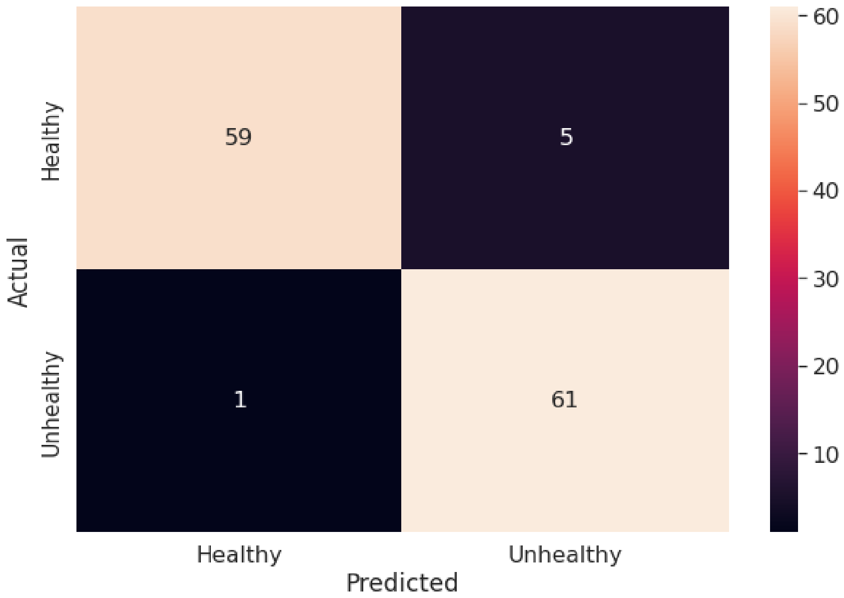

Figure 4.4: Confusion Matrix for VGG16 (Transfer Learning)

Tabel 4.1: Classification Report (VGG16)

Class	Weight	F1 Score	Precision	Recall
Healthy	None	0.92	1	0.84
Unhealthy	None	0.93	0.86	1
Healthy	ImageNet	0.95	0.98	0.92
Unhealthy	ImageNet	0.95	0.92	0.98

4.1.2 Experimental Result & Analysis for VGG19 Architecture

For the validation dataset, I achieved an accuracy of 80.42 percent using VGG19 for standard architecture up to 150 epochs. The figure 4.5 represents this percentage. And a perfect score of one hundred percent accuracy utilizing pre-trained transfer learning on the dataset used for training. In addition, there were just six epochs used in the development of this exam. The reliability of the validation was 95.24 percent, according to the figure (4.6). If I take a look at the confusion matrix, I can see that the conventional model made inaccurate predictions for 23 photographs, but transfer learning brought that number down to five pictures. In addition to this, the F1 score, accuracy, and recall are all significantly different between the regular model and the transfer learning table (4.2). Transfer learning resulted in a final accuracy on tests of 96.03 percent, compared to the traditional model's final accuracy of 81.74 percent.

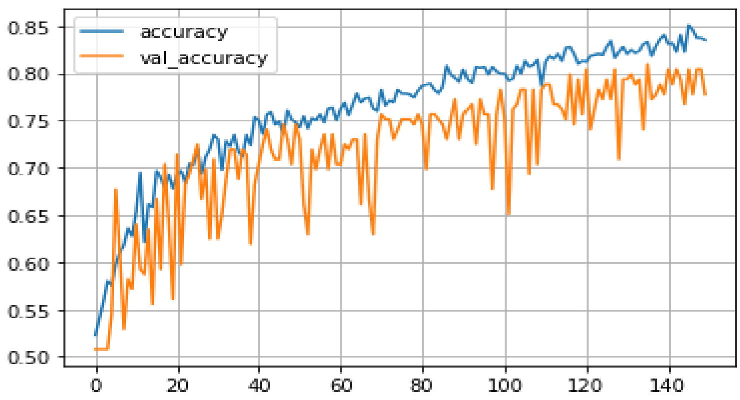

Figure 4.5: Accuracy for Train and validation data (VGG19) (Normal)

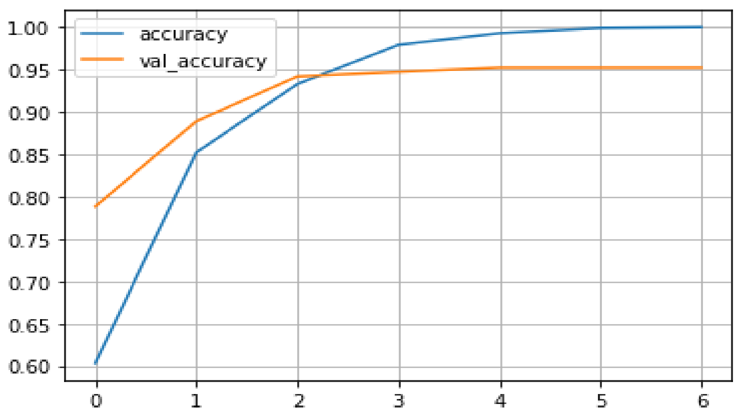

Figure 4.6: Accuracy for Train and validation (VGG19) (Transfer Learning)

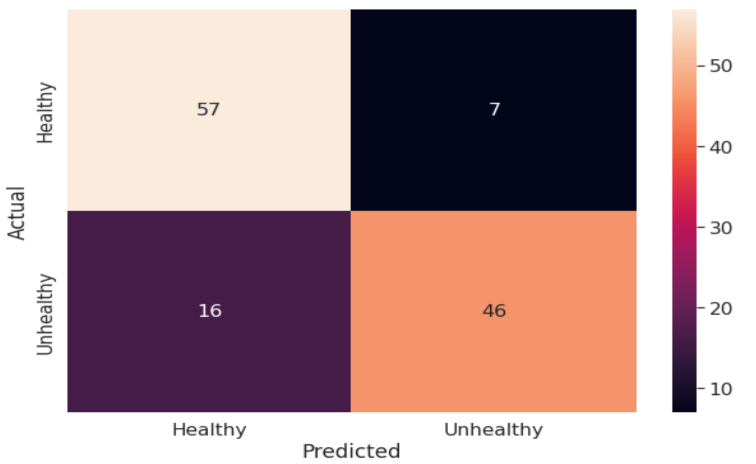

Figure 4.7: Confusion Matrix of VGG19 (Normal)

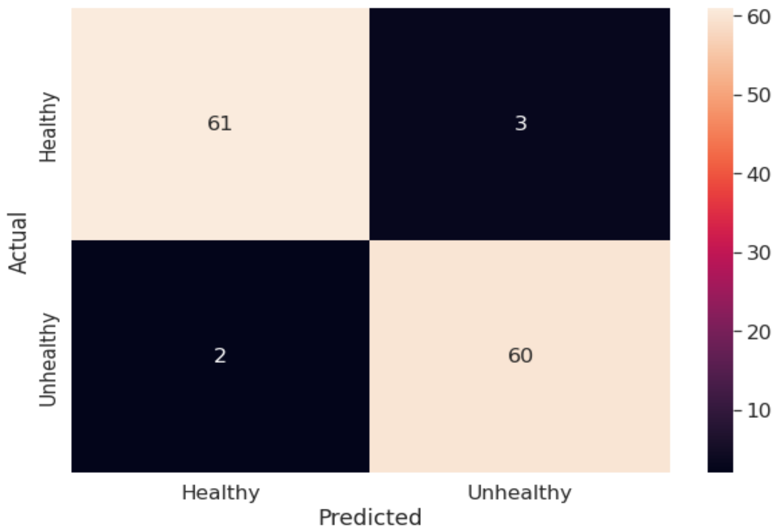

Figure 4.8: Confusion Matrix for VGG19 (Transfer Learning)

Tabel 4.2: Classification Report (VGG19)

Class	**Weight**	**F1 Score**	**Precision**	**Recall**
Healthy	None	0.83	0.78	0.89
Unhealthy	None	0.80	0.87	0.74
Healthy	ImageNet	0.96	0.97	0.95
Unhealthy	ImageNet	0.96	0.95	0.97

4.1.3 Experimental Result & Analysis for MobileNet V2 Architecture

By using the MobileNet V2 standard architecture across all 20 epochs, I was able to achieve an accuracy for the validation dataset that ranged from 50.79% to 4.9. Yet, using transfer learning I was able to achieve 100% accuracy with the training dataset after just five iterations. Figure (4.10) demonstrates a validation accuracy of 92.59 percent, which was obtained by transfer learning. The conventional model made incorrect predictions for 62 photos, which resulted in a whole class being unable to predict this model projected in the confusion matrix; however, transfer learning brought this number down to seven. This model has undergone a significant transformation. Both the normal model and transfer learning provide distinct F1 scores, and they also produce different accuracy and recall tables (4.3). The accuracy of the tests increased from 50.79 to 94.44 percent because to transfer learning.

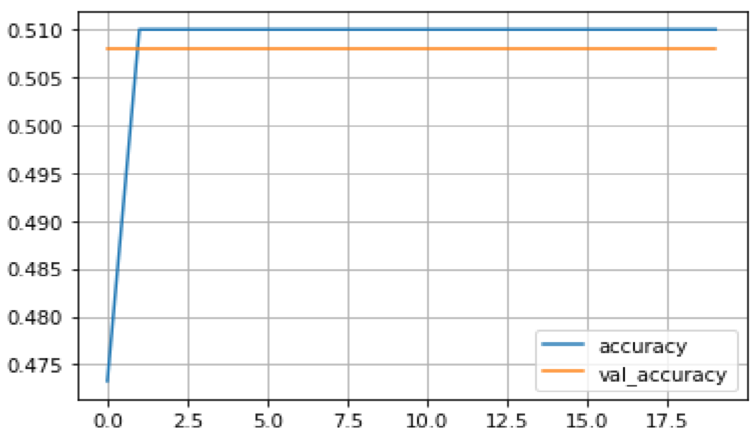

Figure 4.9: Accuracy for Train and validation data (MobileNet V2) (Normal)

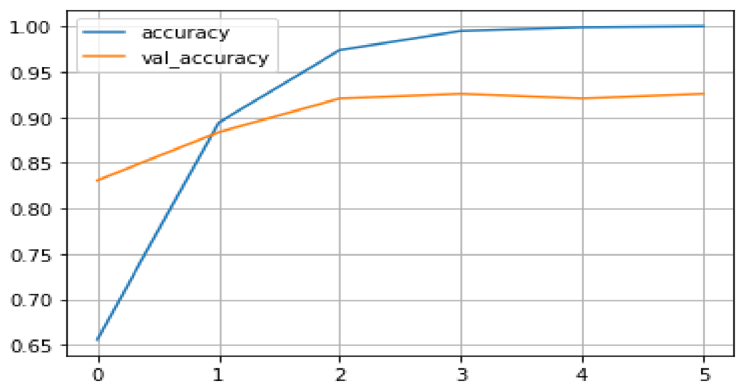

Figure 4.10: Accuracy for Train and validation (MobileNet V2) (Transfer Learning)

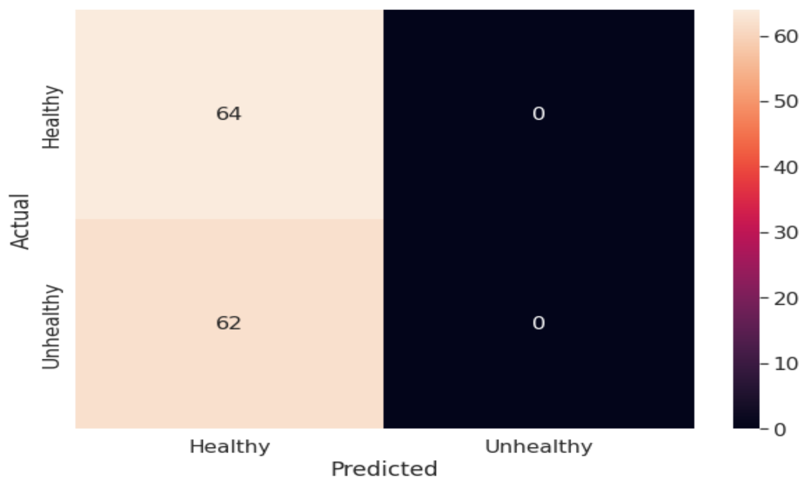

Figure 4.11: Confusion Matrix of MobileNet V2 (Normal)

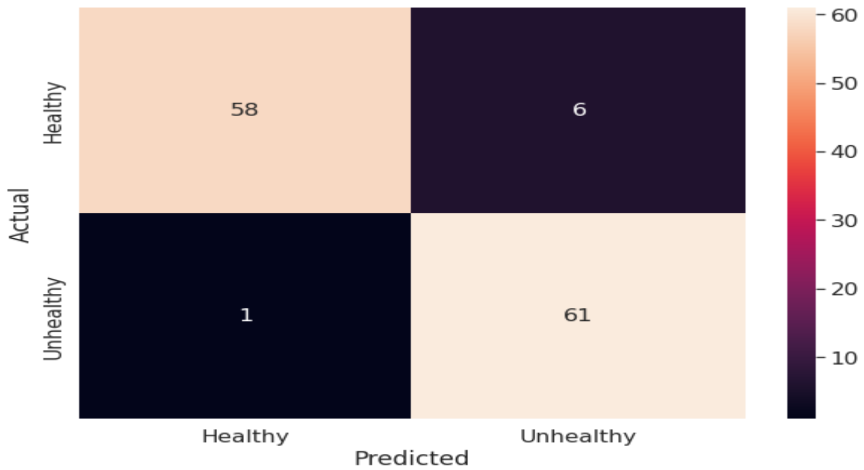

Figure 4.12: Confusion Matrix for **MobileNet V2** (Transfer Learning)

Tabel 4.3: Classification Report (MobileNet V2)

Class	Weight	F1 Score	Precision	Recall
Healthy	None	0.67	0.51	1
Unhealthy	None	0	0	0
Healthy	ImageNet	0.94	0.98	0.91
Unhealthy	ImageNet	0.95	0.91	0.98

4.1.4 Experimental Result & Analysis for ResNet50 Architecture

The ReseNet50 model provides the most accurate results for us. In this case, the accuracy for the validation dataset that I was able to attain using the traditional architecture after the first 150 iterations was 90.48 Figure (4.13). On the other hand, I was able to attain an accuracy of one hundred (100%) with the training dataset when I used pre-trained transfer learning. Considering how accurately the validation was performed, I got a score of 97.35 percent figure (4.14). When I take a look at the confusion matrix, I can see that the traditional model produced nine inaccurate predictions, but when I used transfer learning, that number decreased down to five. Listed as the most unhealthy category with a perfect score. The accuracy of the testing carried out using the traditional model was 92.85 percent, while the accuracy of the testing carried out utilizing transfer learning was 96.03 percent.

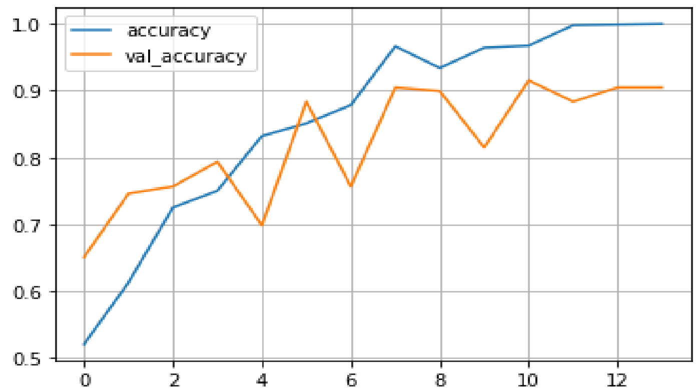

Figure 4.13: Accuracy for Train and validation data (ResNet 50) (Normal)

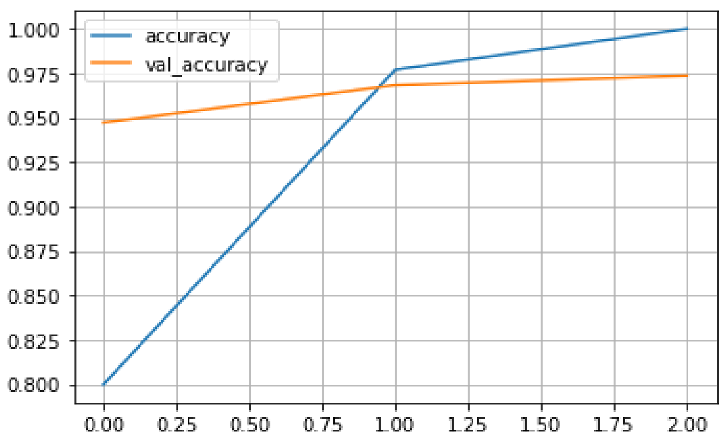

Figure 4.14: Accuracy for Train and validation data (ResNet 50) (Transfer Learning)

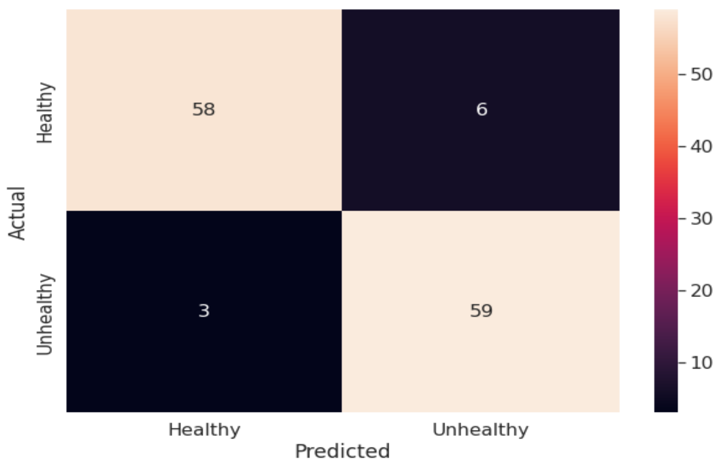

Figure 4.15: Confusion Matrix of ResNet50 (Normal)

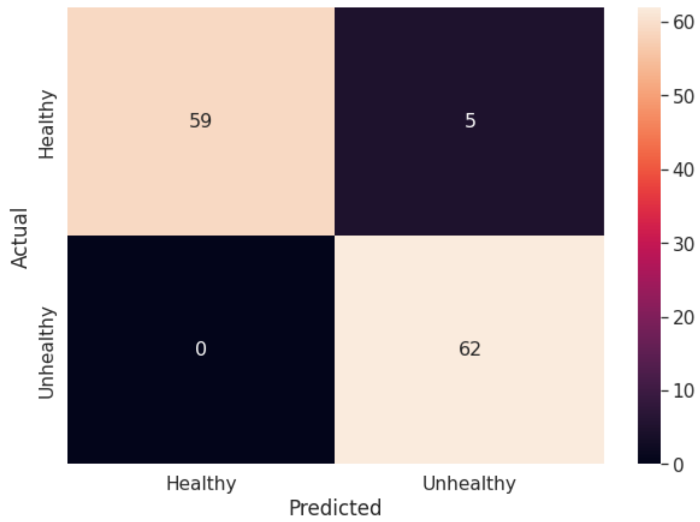

Figure 4.16: Confusion Matrix for ResNet50 (Transfer Learning)

Tabel 4.4: Classification Report (ResNet50)

Class	Weight	F1 Score	Precision	Recall
Healthy	None	0.93	0.95	0.91
Unhealthy	None	0.93	0.91	0.95
Healthy	ImageNet	0.96	1	0.92
Unhealthy	ImageNet	0.96	0.93	1

4.1.5 Experimental Result & Analysis for Efficient NET V2L Architecture

A complete analysis of Efficient NET V2L classification model's ability to differentiate between healthy and unhealthy samples is shown in the table below. The model is evaluated based on two distinct circumstances, the first of which does not make use of ImageNet weights, and the second of which does. The model gets an F1 score of 0.66 when evaluating the "Healthy" class without the use of ImageNet weights, which is indicative of excellent overall accuracy. The recall of 0.94 implies that it has a strong capacity to identify most of the genuine healthy samples, which is consistent with the accuracy of 0.51, which means that it properly detects around half of the true positive healthy samples. Because of the inclusion of ImageNet weights, the performance of the model greatly increases, as seen by the better F1 score, precision, and recall values for both the "Healthy" and "Unhealthy" classes of images. This highlights how successful it is to use pre-trained weights for improved classification accuracy in this specific challenge.

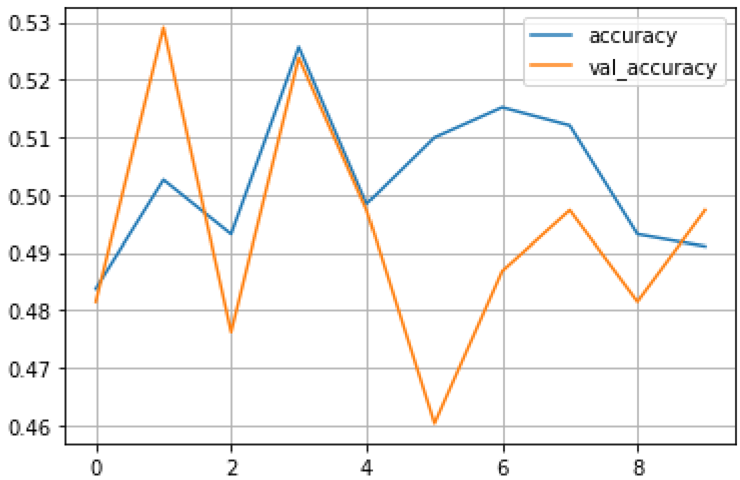

Figure 4.17: Accuracy for Train and validation data (Efficient NET V2L) (Normal)

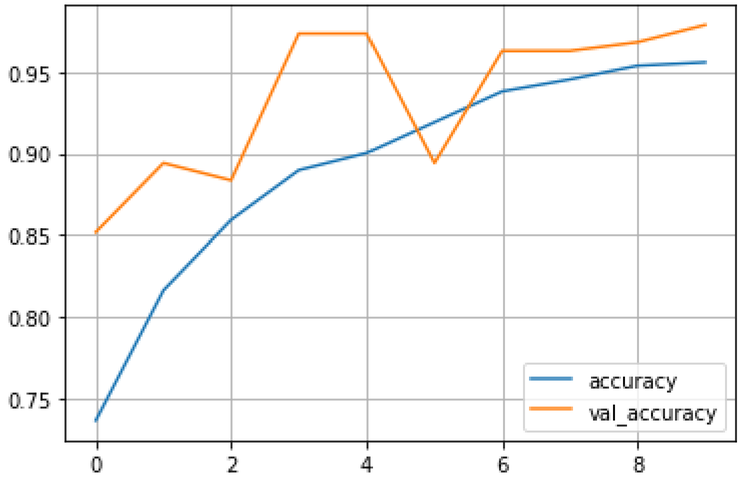

Figure 4.18: Accuracy for Train and validation data (Efficient NET V2L) (Transfer Learning)

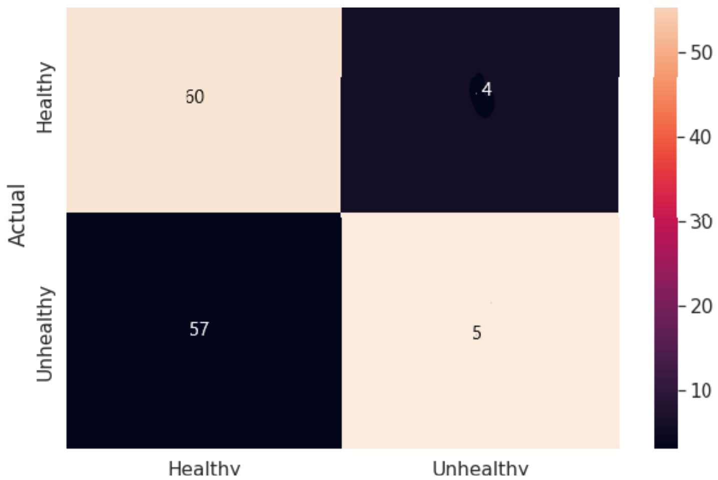

Figure 4.19: Confusion Matrix of (Efficient NET V2L) (Normal)

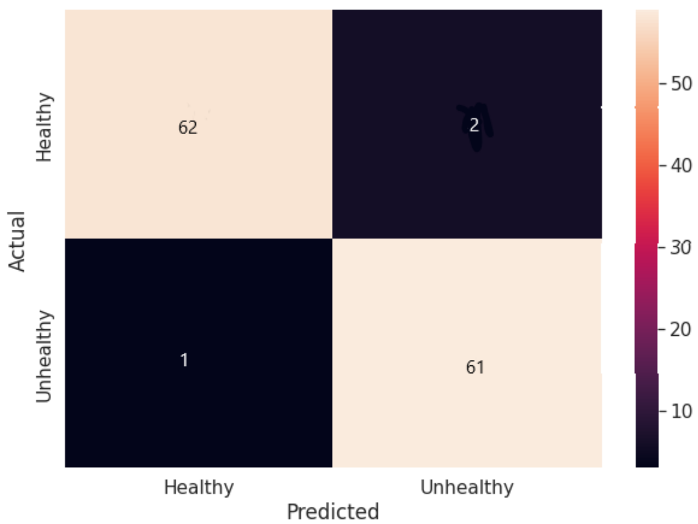

Figure 4.20: Confusion Matrix for (Efficient NET V2L) (Transfer Learning)

Class	Weight	F1 Score	Precision	Recall
Healthy	None	0.66	0.51	0.94
Unhealthy	None	0.14	0.56	0.08
Healthy	ImageNet	0.98	0.98	0.97
Unhealthy	ImageNet	0.98	0.97	0.98

Tabel 4.5: Classification Report (Efficient NET V2L)

4.1.6 Experimental Result & Analysis for ConvNeXtXLarge Architecture

A ConvNeXtXLarge model that was developed to categorize samples into "Healthy" and "Unhealthy" categories is evaluated, and the results of that assessment are shown in the table below. The performance of the model is evaluated both with and without the use of the ImageNet weights. The model earns an F1 score of 0.93 for the "Healthy" class even without the weights from ImageNet, which indicates that it has a good overall accuracy. The precision of 0.88 shows that it properly detects 88% of the genuine positive healthy samples out of all samples classed as healthy, while the recall of 0.98 indicates a strong capacity to identify most of the actual healthy samples. Both of these numbers are impressive. The model also gets an F1 score of 0.91 when evaluating the "Unhealthy" class without the use of ImageNet weights, which indicates that it has an excellent overall accuracy. The precision of 0.98 implies that it properly identifies 98% of the genuine positive unhealthy samples out of all samples classed as unhealthy, while the recall of 0.85 shows that it detects 85% of the actual unhealthy samples. Both of these numbers are much higher than the average recall of 0.7. The addition of ImageNet weights results in a further improvement in the performance of the model, as shown by improved F1 scores, accuracy values, and recall values for the "Healthy" and "Unhealthy" classes, respectively. This demonstrates how successful it is to make use of pre-trained weights in order to achieve increased classification accuracy in this specific challenge..

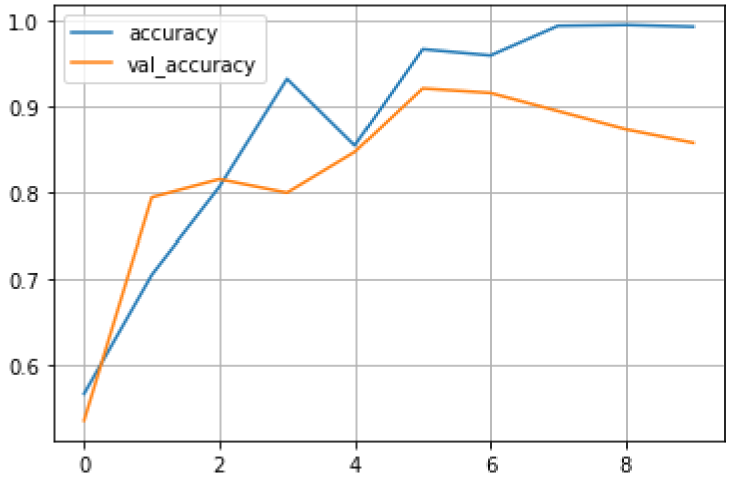

Figure 4.21: Accuracy for Train and validation data (ConvNeXtXLarge) (Normal)

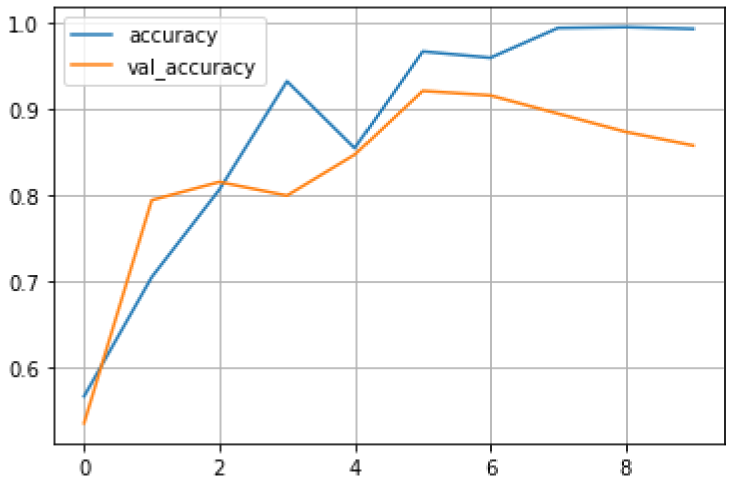

Figure 4.22: Accuracy for Train and validation data (ConvNeXtXLarge) (Transfer Learning)

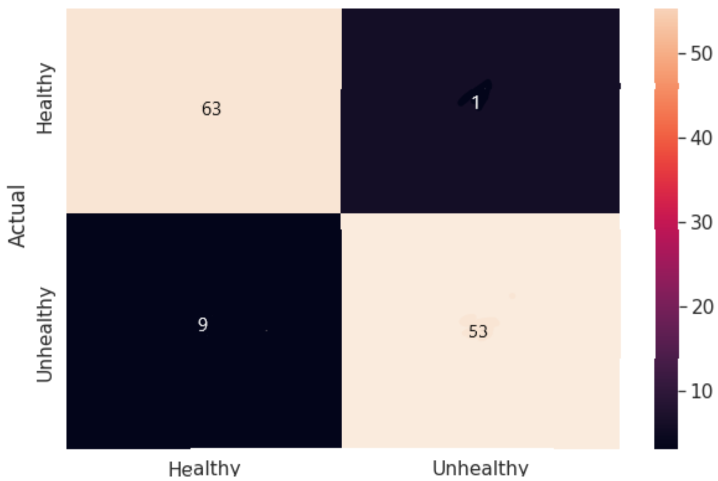

Figure 4.23: Confusion Matrix of ConvNeXtXLarge(Normal)

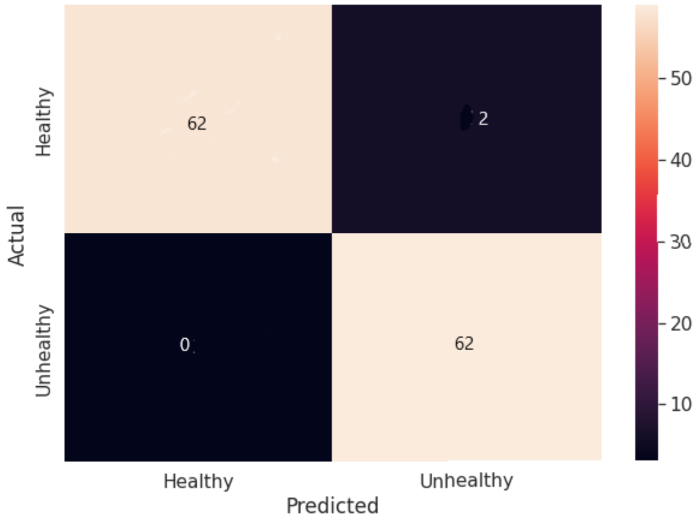

Figure 4.24: Confusion Matrix for ConvNeXtXLarge(Transfer Learning)

Tabel 4.6 Classification Report (ConvNeXtXLarge)

Class	Weight	F1 Score	Precision	Recall
Healthy	None	0.93	0.88	0.98
Unhealthy	None	0.91	0.98	0.85
Healthy	ImageNet	0.98	1	0.97
Unhealthy	ImageNet	0.98	0.97	1

4.2 Performance Comparisons

Table 4.7: Result Comparison of Architectures

Architecture	**Weight**	**Accuracy**	**Validation Accuracy**	**F1 Score**	**Precision**	**Recall**	**Sensitivity**	**Specificity**
VGG19	None	81.74	80.42	0.816	0.824	0.8162	0.7808	0.8679
VGG16	None	92.06	85.71	0.9203	0.93	0.921	1	0.861
ResNet50	None	92.85	90.48	0.9285	0.9292	0.9289	0.9508	0.9076
mobileNetV2	None	50.79	50.79	33.64	0.2539	0.5	0.5	0
ConvNeXtX Large	None	92.06	85.71	0.9201	0.9282	0.9196	0.875	0.981
Efficient NET V2L	None	51.58	49.74	0.4019	0.5341	0.5090	0.5128	0.5555
VGG19	Image Net	96.03	95.24	0.9603	0.9603	0.9604	0.9682	0.9523
VGG16	Image Net	95.23	95.24	0.9523	0.9523	0.9528	0.9833	0.9242
ResNet50	Image Net	96.03	97.35	0.9602	0.9626	0.9609	1	0.9253
mobileNetV2	Image Net	94.44	92.59	0.944	0.9467	0.945	0.983	0.9104
ConvNeXtX Large	Image Net	98.41	98.41	0.9841	0.9843	0.9843	1	96.87

Efficient NET V2L	Image Net	97.61	97.88	0.9761	0.9761	0.9763	0.9841	0.9682

VGG19 - None: The performance of the VGG19 deep learning architecture is shown in this row, despite the absence of any pre-trained weights. In the test dataset, the model has an accuracy of 81.74%, while during the validation process, it has an accuracy of 80.42% and receives an F1 score of 0.816. Specificity is 0.8679, sensitivity is 0.7808, precision is 0.824, and recall is 0.8162.

VGG16 - None: In this row, the performance of the VGG16 architecture is shown without the use of pre-trained weights. Using the training dataset, the model obtains an accuracy of 92.06%, while during validation it achieves an accuracy of 85.71% and receives an F1 score of 0.9203. The specificity is 0.861, the recall is 0.921, the sensitivity is 1, and the precision is 0.93.

This row demonstrates the performance of the ResNet50 architecture when pre-trained weights are not used. It is labeled "None." For the test dataset, the model has an accuracy of 92.85%, while during validation it has an accuracy of 90.48% and receives an F1 score of 0.9285. There is a 0.9292 precision, a 0.9289 recall, a 0.9508 sensitivity, and a 0.9076 specificity.

mobileNetV2 - None: The performance of the mobileNetV2 architecture is shown in this row without the use of pre-trained weights. Using the dataset that was used for testing, the model had an accuracy of 50.79%, and it had the same accuracy throughout the validation process, which resulted in an F1 score of 0.2539. There is a half in precision, a half in recall, a zero in sensitivity, and a zero in specificity.

VGG19 - ImageNet: The performance of the VGG19 architecture using pre-trained weights taken from the ImageNet dataset is shown in this row. Using the dataset used for testing, the model has an accuracy of 96.03%, while during validation it obtains an accuracy of 95.24% and receives an F1 score of 0.9603. There is a 0.9603 precision, a 0.9604 recall, a 0.9682 sensitivity, and a 0.9523 specificity.

This row demonstrates the performance of the VGG16 architecture using pre-trained weights derived from the ImageNet dataset. VGG16 - ImageNet The model has an accuracy of 95.23% on the dataset that it is being tested on, and it has an accuracy of 95.24% while it is being validated, earning it an F1 score of 0.9523. There is a 0.9528 precision, a 0.9833 recall, a 0.9242 sensitivity, and a 0.9833 specificity.

This row demonstrates the performance of the ResNet50 architecture when it is pre-trained with weights taken from the ImageNet dataset. The accuracy of the model is 96.03% when it is applied to the test dataset, and it achieves 97.35% accuracy when it is validated, earning an F1 score of 0.9602. Specificity is 0.9253, sensitivity is 1, and precision is 0.9626. The recall is 0.9609.

MobileNetV2 - ImageNet: The performance of the mobileNetV2 architecture using pre-trained weights derived from the ImageNet dataset is shown in this row. Using the training dataset, the model obtains an accuracy of 94.44%, while during validation it achieves an accuracy of 92.59% and receives an F1 score of 0.944. The recall rate is 0.945.

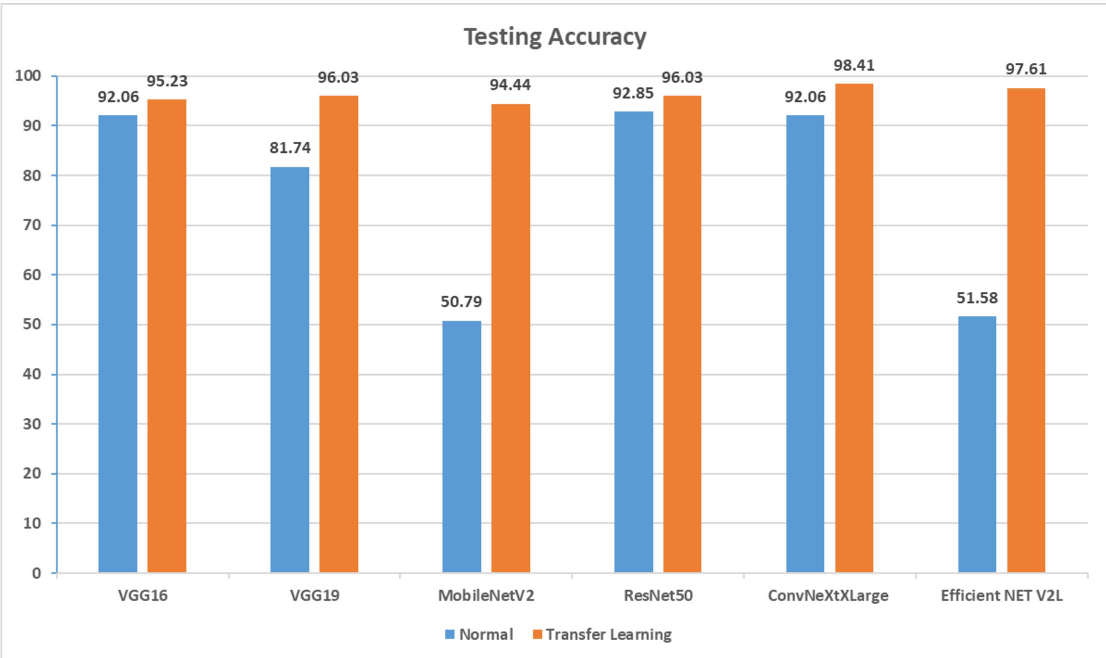

Figure 4.25: Testing Accuracy visualization

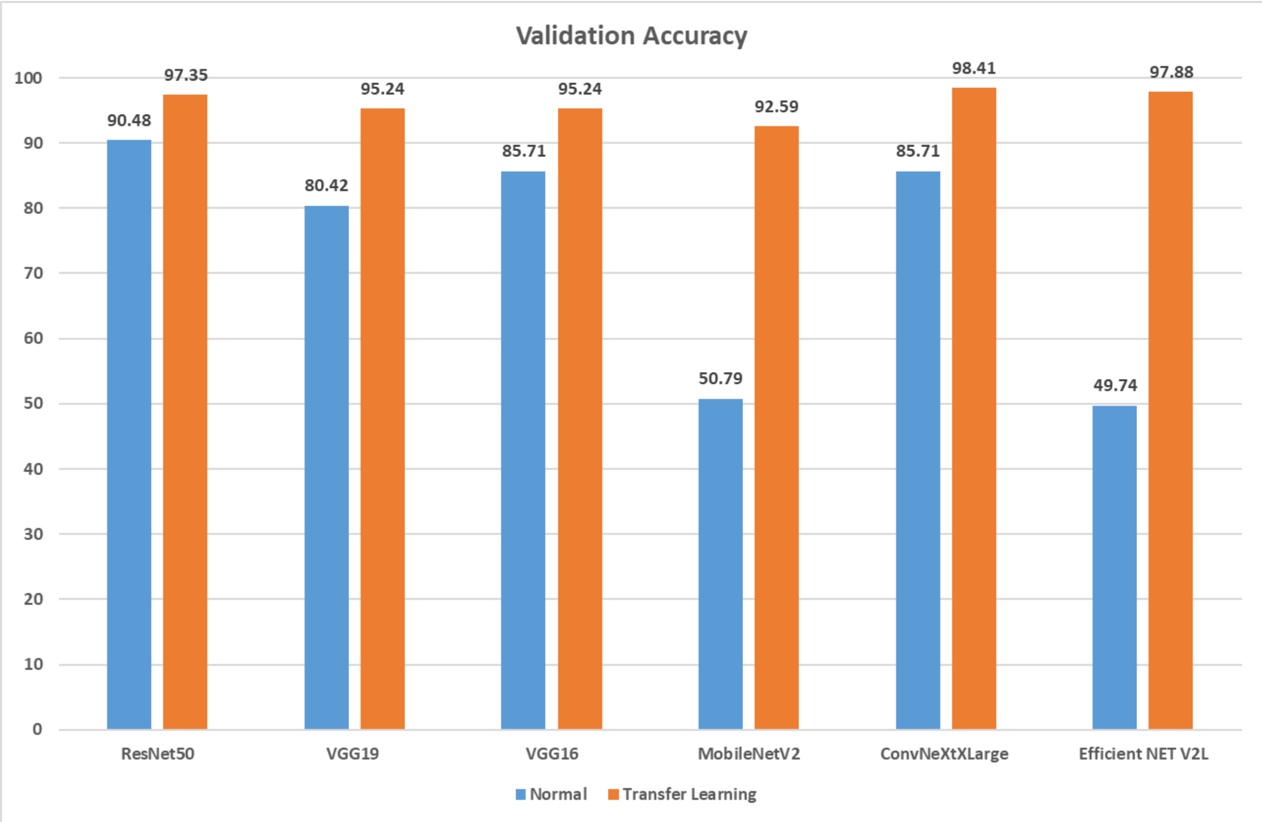

Figure 4.26: Validation Accuracy visualization

It is clear from the data shown in table (4.5) that Transfer Learning performs much better than the conventional methodology. I am able to see from the table and the figure (4.17) that ConvNeXtXLarge transfer learning provides outcomes with an accuracy of 98.41%. Yet, when I look at the validation accuracy comparison figure, I can observe that ConvNeXtXLarge also offers 98.41% accuracy in validation data, but others give less accuracy in validation data. Thus, I suggest using ConvNeXtXLarge as a method for determining the overall health of rice seeds.

CHAPTER 5

Impact on Society, Environment, and Sustainability

5.1 Impact on Society

Rice is the most significant cereal crop in Bangladesh, ranking third behind maize and wheat as the country's most important agricultural export. This dinner has a significant amount of significance for the majority of Bangladeshi households. In recent years, there has been a significant rise in the demand for rice in Bangladesh, although output has remained relatively stagnant. Agricultural resources are often regarded as being among the most significant natural resources that are both renewable and dynamic. The majority of my arable land is farmed by subsistence farmers who live in poverty. Rice and rice seeds are the most in-demand crops in Bangladesh, where the great majority of the population lives in rural areas. Bangladesh's economy is likewise heavily dependent on agriculture. Farmers investigate almost each and every one of them. Individuals who are not informed on agricultural studies outside of what they know are only able to rely on their strategies and previous experiences. There are some scientific experiences among them, but not all of them. Other from that, they study a relatively small percentage of seeds. As a result, corporations are mostly disregarded and rely nearly entirely on chance.

5.2 Impact on the Environment

Diseases not only result in a loss of productivity but also in contamination of the natural environment. Rice is susceptible to a wide variety of illnesses, many of which have the potential to severely reduce crop productivity. Of these factors, rice plant diseases are responsible for a loss in output of 10-15% of rice (Peng et al., 2009). More than 2.5 billion people throughout the globe make their living in some way directly or indirectly from the agricultural sector. Urgent action is necessary as a result of the inherent relationship that the industry has with the surrounding natural environment, the direct dependency that the industry has on natural resources for production, and the significance of the industry for the socioeconomic growth of the country. as well as ambitious plans to construct agricultural systems that are more robust. The information that is said to be dreadful, life-threatening, and has swallowed up billions of dollars in recovery and rebuilding is often what dominates the headlines. But are I seeing an increase in the frequency and severity of natural disasters? Or are I allowing my selves to be influenced by a cognitive bias? If nothing is done to change the situation, I may as well prepare for the end of the world when it comes to the agricultural environment.

5.3 Sustainability Plan

The goal of this research is to determine which kind of rice seeds are harmful. rice that is harmful and instead choose rice that is healthful. imaging of the rice seedlings' state of health in real time maybe Using an Internet of Things robot to sort out the unhealthy rice seeds from the healthy ones. The provision of deep learning models with the intention of pointing farmers in the right direction is my primary objective. Provide some pictures of the rice germination process. my deep-learning model extracts The information that informs us whether or not the rice seed in question will develop into rice wood Class circumstances that were specified as healthy and harmful. Not only does disease result in a loss of output, but it also results in the damage of the ecosystem and contamination of the environment. Rice fields that are infected with disease provide much lower yields. They suffer a loss of 10-15% of their total rice yield as a result of rice disease (Peng et al., 2009). Windows as an operating system and the programming language Python are required for the successful completion of my task.

CHAPTER 6

Summary, Conclusion, Recommendation, and Implication for Future Research

6.1 Summary of the Study

The provision of deep learning models with the intention of pointing farmers in the right direction is my primary objective. Please provide some images of rice as soon as possible. my deep-learning model extracts Information that informs us if the specified sauce contains rice or not, as well as whether or not it contains wood. Conditions that are predefined to fall into the healthy and unhealthy groups. Not only does disease result in a loss of output, but it also results in the damage of the ecosystem and contamination of the environment. Rice fields that are infected with disease provide much lower yields. They suffer a loss of 10-15% of their total rice yield as a result of rice disease (Peng et al., 2009). The Windows operating system and the programming language known as Python are required for the successful completion of my goal. In addition to that, it makes use of a number of other packages, such as Keras, TensorFlow, and NumPy, among others. programming activities with Google Collaborator, an online community based tool that is hosted in the cloud. The categorization model was developed using a GPU device of the type Tesla P100. I made advantage of the architecture's deeper levels, including the pooling layer, the full Connectivity Layer, the SoftMax Layer, and the Output Layers. In this section of the paper, the technique that was used for the study is discussed. as well as the procedures that I followed to finish this investigation. I have, first and foremost, collected real-time data on the Kaguru rice seeds. This also comes from the immediate area. In the rice fields, I scraped and snapped shots with my camera. There are two primary classifications of seeds. Both healthy and sick seeds are included in this dataset as separate classes. Then, you can count on us. Established a number of processes for processing the data, and then partitioned those processes into testing and training sets. For the training of a dataset such as this, I used a CNN-based architecture, namely MobileNetV2, ResNet50, VGG16, and VGG19. The usual accuracy for VGG16 was determined to be 92.06%. The accuracy of VGG19 when compared to Mobinet V2 or ResNet50 is typically 81.74%, 50.79%, or 92.85% respectively. These outcomes are enhanced by transferring learning and improving accuracy correspondingly by 96.03%, 92.59%, and 96.03%. The transfer accuracy is 95.24%. The results of this investigation provide a substitute method. An investigation on the diagnosis of plant diseases using deep learning. my jumpsuit Transfer learning achieves an accuracy of about 97%, which may be further improved. Accuracy even when a huge number of data inputs may be added in a relatively short amount of time. I have high hopes that my approach would assist farmers in precisely determining the condition of their health after deployment. In addition, it is not out of the realm of possibility for engineers to create an Internet of Things device that can automatically remove damaged seeds from fields. IoT

gadgets are built on top of my paradigm as the basis. In the not too distant future, I want to create a health app that, by using various databases and models, will be able to ascertain a person's current state of health.

6.2 Conclusion

Our ability to accurately determine the patient's health status is greatly enhanced by the ConvNeXtXLarge model, which has a validation accuracy of 98.41 percent. While my data set is somewhat limited, I have taken every precaution to guarantee the quality of my findings. The agriculture sector in Bangladesh might gain from such a program. Because of their lack of education, most farmers in Bangladesh are unable to distinguish between healthy and infected seeds. As a result, they can't gather their produce. my methodology was developed with the intention of assisting farmers in more precisely gauging their current state of health. In addition, engineers may design Internet of Things (IoT) devices to automatically eradicate unhealthy seeds from agricultural areas. my framework will be used as the basis for Internet of Things devices.

In the future, my information and model will be used to create a health app that can determine a person's health status.

REFERENCES

[1] Shrestha, G., Deepsikha, Das, M., & Dey, N. (2020). Plant Disease Detection Using CNN. 2020 IEEE Applied Signal Processing Conference (ASPCON). doi:10.1109/aspcon49795.2020.9276722

[2] Polder, G., Blok, P. M., de Villiers, H. A. C., van der Wolf, J. M., & Kamp, J. (2019). Potato Virus Y Detection in Seed Potatoes Using Deep Learning on Hyperspectral Images. Frontiers in Plant Science, 10. doi:10.3389/fpls.2019.00209

[3] Jiang, F., Lu, Y., Chen, Y., Cai, D., & Li, G. (2020). Image recognition of four rice leaf diseases based on deep learning and support vector machine. Computers and Electronics in Agriculture, 179, 105824. doi:10.1016/j.compag.2020.105824

[4] M. K. R. Asif, M. A. Rahman and M. H. Hena, "CNN based Disease Detection Approach on Potato Leaves," *2020 3rd International Conference on Intelligent Sustainable Systems (ICISS)*, Thoothukudi, India, 2020, pp. 428-432, doi: 10.1109/ICISS49785.2020.9316021.

[5] Chung, Chia-Lin & Huang, Kai-Jyun & Chen, Szu-Yu & Lai, Ming-Hsing & Chen, Yu-Chia & Kuo, Yan-Fu. (2016). Detecting Bakanae disease in rice seedlings by machine vision. Computers and Electronics in Agriculture. 121. 404-411. 10.1016/j.compag.2016.01.008.

[6] Aukkapinyo, K., Sawangwong, S., Pooyoi, P., & Kusakunniran, W. (2019). *Localization and Classification of Rice-grain Images Using Region Proposals-based Convolutional Neural Network. International Journal of Automation and Computing.* doi:10.1007/s11633-019-1207-6

[7] Kumar, Raj & Baloch, G. & Pankaj, & Buriro, Abdul Baseer & Ahmed, Junaid. (2021). Fungal Blast Disease Detection in Rice Seed using Machine Learning.

[8] Lu, Yang, Shujuan Yi, Nianyin Zeng, Yurong Liu and Yong Zhang. "Identification of rice diseases using deep convolutional neural networks." *Neurocomputing* 267 (2017): 378-384.

[9] S. Ramesh, D. Vydeki "Recognition And Classification of Paddy Leaf Diseases using Optimized Deep Neural Network with Jaya Algorithm" in "Journal of Information Processing in Agriculture, published by Elsevier", vol. 6, issue 3, Sep. 2019.

[10] Pranali K., Kosamkar K. Thenmozhi, Dr.V.Y.Kulkarni,KrushnaMantri"Leaf Disease Detection and Recommendation of Pesticides Using Convolution Neural Network"in Fourth International Conference on Computing Communication, Control & Automation (ICCUBEA), August 2018, Pune, India, 18617820, IEEE.

[11] Wan-jie Liang, Hong Zhang, Gu-feng Zhang & Hong-xin Cao"Rice Blast Disease Recognition Using a Deep Convolutional Neural Network", "Scientific Reports - a Nature Research Journal", Article no.2869, doi: 10.1038/s41598-019-38966-0, Feb. 2019.

[12] R. Kaur and V. Kaur, "A deterministic approach for disease prediction in plants using deep learning," 2018.

[13] "Caffe, available at<<" https://caffe.berkeleyvision.org/>> Last Accessed on: 2023-06-26.

[14] Islam, T., Sah, M., Baral, S. and Choudhury, R.R. (2018) A Faster Technique on Rice Disease Detection Using Image Processing of Affected Area in Agro-Field. 2018 Second International Conference on Inventive Communication and Computational Technologies (ICICCT), Coimbatore, 20-21 April 2018, 62-66.

[15] Sladojevic, Srdjan & Arsenovic, Marko & Anderla, Andras & Stefanović, Darko. (2016). Deep Neural Networks Based Recognition of Plant Diseases by Leaf Image Classification. Computational Intelligence and Neuroscience. 2016. 1-11. 10.1155/2016/3289801.

[16] F. T. Pinki, N. Khatun and S. M. M. Islam, "Content based paddy leaf disease recognition and remedy prediction using support vector machine," *2017 20th International Conference of Computer and Information Technology (ICCIT)*, Dhaka, Bangladesh, 2017, pp. 1-5, doi: 10.1109/ICCITECHN.2017.8281764.

[17] S. R. Maniyath, P. Vinod, M. Niveditha, R. Pooja, N. Shashank, R. Hebbar, et al., "Plant disease detection using machine learning," in 2018 International Conference on Design Innovations for 3Cs Compute Communicate Control (ICDI3C), pp. 41–45, IEEE, 2018.

[18] Prajapati, H.B., Shah, J.P. and Dabhi, V.K. (2017) Detection and Classification of Rice Plant Diseases. Intelligent Decision Technologies, 11, 357-373. https://doi.org/10.3233/IDT-170301

ismail1

ORIGINALITY REPORT

29%	27%	12%	13%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	dspace.daffodilvarsity.edu.bd:8080 Internet Source	14%
2	Submitted to Daffodil International University Student Paper	3%
3	www.researchgate.net Internet Source	3%
4	www.jcreview.com Internet Source	2%
5	norma.ncirl.ie Internet Source	1%
6	Junde Chen, Defu Zhang, Y.A. Nanehkaran, Dele Li. "Detection of rice plant diseases based on deep transfer learning", Journal of the Science of Food and Agriculture, 2020 Publication	1%
7	"Sustainable Communication Networks and Application", Springer Science and Business Media LLC, 2021 Publication	<1%