Rice and Corn Leaf Disease Recognition E-Learning Mobile Application using MobileNet Machine Learning Algorithm

by

Wee, Adzhar K.
Algupera, Aizzy Dianne T.

A Thesis Proposal Submitted in Partial Fulfillment of the Requirements for the Degree of Bachelor of Science in Computer Science

Western Mindanao State University
Institute of Computer Studies
Philippines
July 2021

Table of Contents

List of Tables

List of Figures

Chapter 1 – Introduction

1.1 Background of the Study

Ever since the Neolithic era, farming has been one of the wide-scale transition of many human cultures under Agricultural Revolution. As the normal status quo of every living thing, crops also have what one calls ‘crop health’. Crop health determines what aspects affect a crop on its growing stage (S. Savary *et al.*, 2017). Along with it is the importance in studying the pathogens that always come with the effect on food shortages especially on famines. Doing different tests on what crops best suit the climate in areas around the world, farmers have cultivated crops since then (Toshichika & Ramankutty, 2015).

Crop health can be broken down into three (3) factors which mainly affect a plant physiology; (1) *biological, (2) chemical* and *(3) physical.* Depending on whatever factor affects a crop growth, its yield can be greatly affected (J. Grunert). By assuring the best possibility for higher yield, farmers in other countries developed approaches in checking crop health issues. Traditionally, crop health is assessed via visual inspection of the physical features of the crop, however, as modern technology has its advantages, agriculture technology advanced to using data from machines or satellite imagery. The farmers analyze the data relating natural resources (water, soil, fuel and even the brightness of the sun) and other inputs such as fertilizers more accurately and efficiently (Shaping Agriculture). Similarly, in the Philippines, most large farms now have integrated technology in managing the crop health for higher yield. With this, farms started cutting costs on labor and are now more cost efficient. (Mogato, 2018).

Particularly, branching from the crop health assessment is the crop disease management and prevention. Like any other living things, crops can also be affected by diseases, as no living thing is unsusceptible to diseases. (Shark Keeper). From temperature to humidity, pest infestation, light, atmospheric carbon dioxide, soil moisture and properties. These factors can result in the appearance of diseases that will affect the harvesting yield of crops (J. Plant Pathol Microbiol, 2021).

Zamboanga City is a small industrial city found in Mindanao Region IX. Though the city is mainly on industrial development, the city still provides its citizens with agricultural crops, where two of the main source of grain crops are rice and corn. Rice and Corn have always been a staple crop for Filipinos and is national-wide demand food source (The Philippines). Since the city cannot accommodate too many large farms with its topography, it sometimes depends on neighboring regions and provinces regarding crop production (Cedeño, 2021). One region in particular is where the city is located, with an annual harvest percentage of 6.4% for corn and 5.5% for rice (Palay and Corn Quarterly Bulletin, July 2020). New technologies have surfaced and have aided farmers on different aspects of crop management (M. Terri, October 2019). With this, a focus on detecting crop diseases, mainly on rice and corn is the system's core for development. As there are newly inspired agriculture students and farmers, there are those that do not know much on crop management, especially in regards to its diseases. In connection to this, the application will aid the users on their endeavors on understanding the types of leaf diseases that rice and corn crops sustain. Leaf diseases are more noticeable and treatable, thus, became the focus of the application development. It also includes a list of system registered diseases to provide users a detailed-view of information regarding diseases and their prevention. Overall, the system's detection accuracy reaches a 90% percentile upon testing its dataset accuracy.

Figure 1. Rice (a) and *Corn Plant (b)*

1.2 Statement of the Problem

Grain crops have a percentage of health deterioration in which it is required to be maintained. Traditionally, farmers use their acquired knowledge throughout the years in examining the crops through the way they look and sometimes how the crop leaves feel when being held. In this matter, these diseases can be detected on the crops as they affect the plant's physical appearance. Examples are shown on Figures 2 and 3, where the disease can be seen through the leaves' appearance.

Figure 2. Healthy Rice Plant (a)

Blast Disease (b)

Crops that are affected by a disease are mostly required to be removed or disposed of immediately, as the infection can spread to nearby crops (Crop Genebank Knowledge Base). But, for the majority, farmers will only help prevent the spreading of such diseases to other crops, by using fertilizers, insecticides and pesticides. However, there are those who aspire to enter the field of cultivation without much knowledge on this front. Students, businessmen and young farmers tend to make mistakes when making theories on what type of chemical to use for the disease-stricken crop grains.

Figure 3. Healthy Corn (a)

Gray Leaf Spot (b)

1.3 Objectives

1.3.1 General Objective

The aim of the study is to build an e-learning app that will help agricultural students and aspiring farmers in assisting them with comprehending the types of diseases that strike the crops.

1.3.2 Specific Objectives

To achieve the general objective, it is necessary to:

- Detect the rice and corn disease with a possibility of over 80% accuracy using a sensor (camera);
- Predict the common leaf diseases found locally in the Philippine rice and corn farm along with their cause/s;
- Use a mobile to minimize workload on aspiring knowledge-seekers;
- Provide knowledge on diseases and registered crop diseases;
- Use percentage accuracy in validating crop disease detection; and
- Prove that the detection of crop disease is highly accurate using MobileNet algorithm.

1.4 Significance of the Study

The results of this research would be significant to:

- **People.** For citizens to use for educational or personal researches needed on crop disease mainly rice and corn.

- **Students.** The e-learning mobile app, can help students in identifying the crop disease and understand better how these diseases look like and what causes each.

- **Farmers.** For them to be assisted on checking whether the crops are bearing diseases or not when cultivating them.

- **Department of Agriculture.** This system would serve as an additional input on agricultural-related topic. This would aid them in verifying the diseases of rice and corn.

- **Researchers.** The study would assist and guide them for future research-related on crop health of rice and corn. This study would also provide important things on the matter at hand to further support their studies.

1.5 Scope and Limitations

Scope of the study is as follows:

Users are mostly farmers who will benefit on the study and also citizens who will need it for researches. Since it is an android application, the system can only identify one image at a time. Using the camera of the phone, the user can capture and upload an image of the crop and register it to the application for recognition process. Detection of diseases is based on the physical look of the crops, as the dataset consists of images on crop diseases. A pre-captured image of a crop can also be used for identification. Using a custom trained model for detection, the app can run even when the device is not connected to the internet.

Limitations of the study are as follow:

Disease detection will only cover those that are caused by *fungi* which are commonly found within the Philippine farms of rice and corn as these diseases are the ones that show the most change in the crops. Our team also included the label of a healthy crop for the users to fully utilize the system and its functionality. The table is justified at Chapter 4.

Rice	Corn
Leaf Blight	Philippine Mildew
Brown Spot	Blight
Leaf Blast	Common Rust
Tungro	Gray Leaf Spot

Table 1. Rice and Corn Diseases

The system accepts captured images of the rice and corn plant. Stored images are also accepted. Cameras need to have a high clarity of pixels; 20mp or higher to assure the recognition process proceeds with its high accuracy rate. The application does not support voice-command. It is also expected to run on Android platform version 6.0 (Marshmallow) and higher.

1.7 Operational Definition

Sensor – A **sensor** is a device that produces an output signal for the purpose of sensing a physical phenomenon (Wikipedia).

MobileNet Algorithm – MobileNet model from TensorFlow for image classification of the system. It is Tensor Flow's first mobile computer vision model, and it is designed for use in mobile application. This model uses depth wise separable convolution layers to build low-latency deep neural networks for mobile and embedded devices (Image Classification with MobileNet).

Convolutional Neural Network – A convolutional neural network is a special kind of feedforward neural network with fewer weights than a fully-connected network (T. Wood).

VGG16 – is a convolutional neural network model proposed by K. Simonyan and A. Zisserman from the University of Oxford in the paper "Very Deep Convolutional Networks for Large-Scale Image Recognition". The model achieves 92.7% top-5 test accuracy in ImageNet, which is a dataset of over 14 million images belonging to 1000 classes (Popular Networks).

NasNet – Neural Architecture Search Network or NasNet is a machine learning model which automates network architecture engineering (D. Nair).

SqueezeNet – SqueezeNet is a deep neural network that was released in 2016. SqueezeNet was developed by researchers at Deepscale, University of California, Berkeley and Stanford University. In designing SqueezeNet, the authors' goal was to create a smaller neural network with fewer parameters that can more easily fit into computer memory and can more easily be transmitted over a computer network (Wikipedia).

Chapter 2 – Review of Related Literature

2.1 Related Studies

This chapter will discuss the different studies and researches on crop health recognition both on local and foreign approach. As there are numerous methods on crop health recognition, this chapter will focus on taking into account some of the closely-related studies.

2.1.1 Foreign Studies

2.1.1.1 Research Paper on Mobile aRCee Checker an Application of Rice and Corn Checker for Nutrient Deficiency through Leaf Coloration

It is a research paper on developing rice and corn checker through leaf coloration. Within the research, it shows an application that can quantify the Nitrogen, Potassium and Phosphorus deficiency through the image processing of the leaves. It can diagnose rice and corn nutrient deficiency through the leaf coloration and patterns being recognized from the image (M. Eder, 2016).

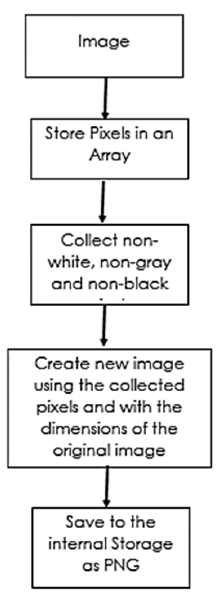


Figure 4. Algorithm for capturing image of the corn leaf

The above Figure 4 shows the algorithm source code of capturing the image of a corn leaf. Its process starts with storing pixels in an array, then the application collects the non-white, non-gray and non-black pixels. From that, it then creates a new image using the collected pixels following the dimensions of the original image, storing it in the internal storage as a PNG.

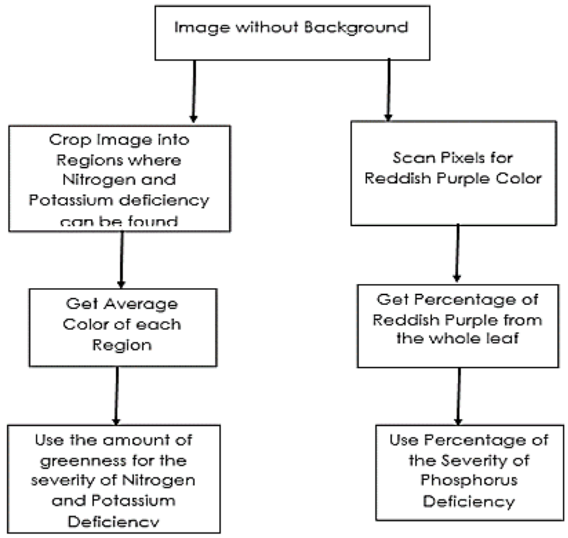


Figure 5.Algorithm for Detecting Nutrient Deficiency

Figure 5 shows the algorithm for detecting nutrient deficiency from captured crop leaf image. This algorithm relies on images without background, which greatly helps the application on determining and scanning images.

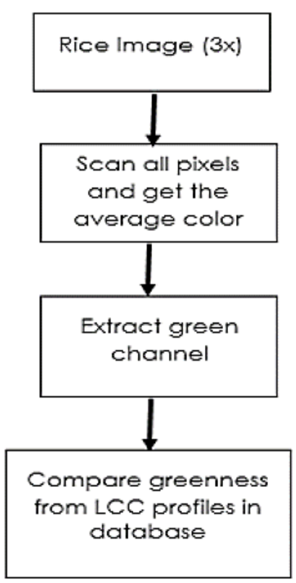


Figure 6. Algorithm for Diagnosing Nitrogen Deficiency for Rice

From the experimental process, results have shown that the application greatly improved the diagnosing of nutrient deficiency on rice and corn images. It can accurately calculate the nutrient deficiency thus, receiving positive feedbacks from respondents.

2.1.1.2 Identification and Recognition of Rice Diseases and Pests using Deep Convolutional Neural Networks

This research paper focuses on rice crops through accurate and timely detection of its diseases and pests. It allows the farmers to apply timely treatment that will greatly reduce economic losses. By utilizing convolutional neural networks (CNN), the research paper approached deep learning-based networks for detecting rice diseases and pests (R. Rhaman *et al.,* 2018).

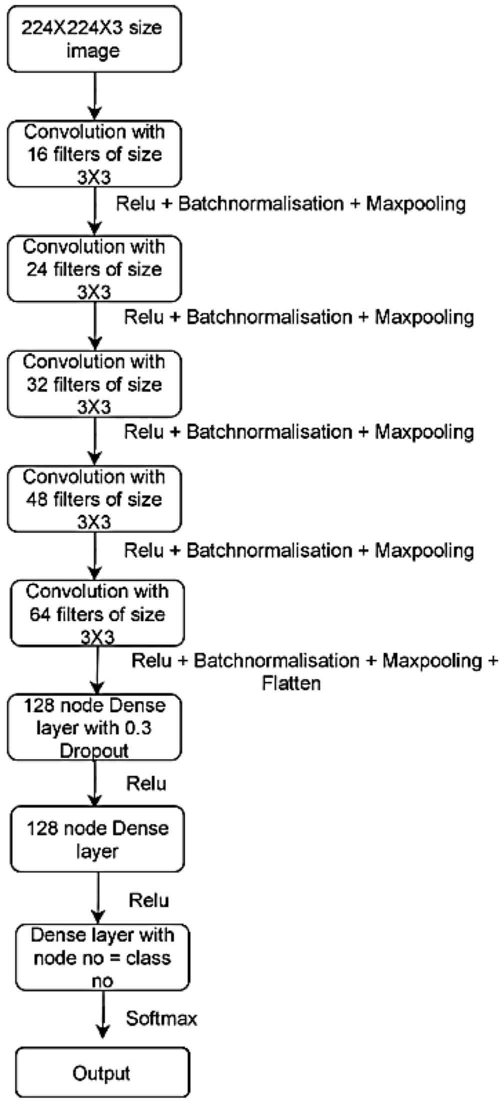


Figure 7. CNN Architecture

The above shown figure is the utilized CNN architecture for the research. It shows the process of how it breaks down the image data into pieces. Constructed from scratch, this CNN architecture was inspired by the sequential nature of VGG16. This architecture only has a 0.8 million parameters compared to a 138 million parameters of VGG16.

In summary, the experimental results show the effectiveness of using two-stage small CNN rather than the state-of-the-art CNN architectures such as MobileNet, NasNet Mobile and SqueezeNet, delivering a desired accuracy of 93.3%, while being a greatly reduced model size (99% smaller than VGG16).

2.1.1.3 Deep Convolutional Neural Network based Detection System for Real-time Corn Plant Disease Recognition

Corn is considered to be the most popular grain in India and having high percentage of crop loss due to diseases threatens the food availability and storage. By researching deep convolutional neural network, smart devices are used to utilize and provide automatic diagnosis of corn diseases and sever crop losses (S. Mishra *et al.*, 2020).

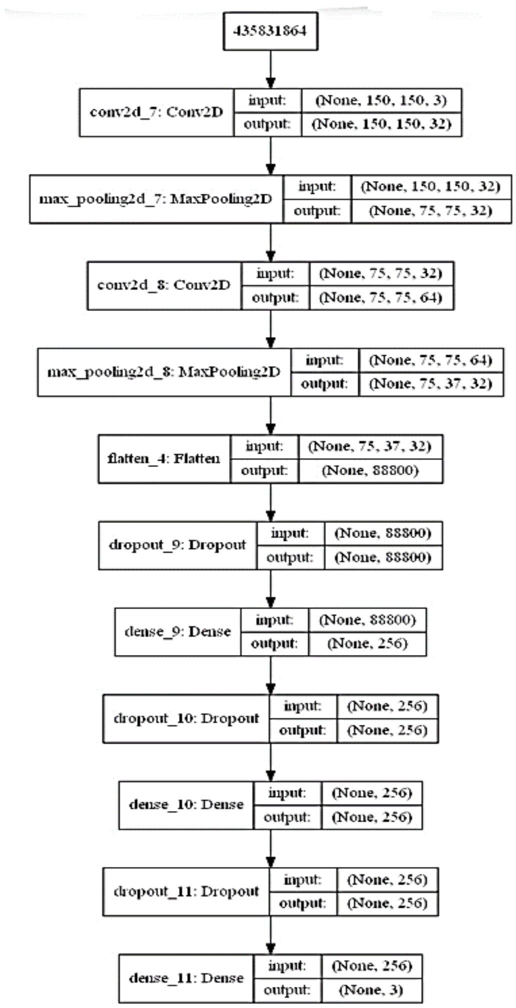


Figure 8. Architecture of DNN trained on GPU

As shown on Figure 8, Convolutional neural network is created by stacking a sequence of layers, namely: Convolutional Layers, Max-Pooling Layers, Activation Layers, and Dropout Layers. The paper presents real-time method, based on the research deep convolutional neural networks. By tuning hyper-parameters and adjusting the pooling combinations on a system with GPU, the deep neural network performance is improved. It has utilized the use of a pre-trained deep CNN model, which was deployed onto a raspberry 3 using Intel Movidius Neural Compute Stick consisting dedicated CNN hardware blocks. During the experimental process, the recognition of corn leaf diseases achieves an accuracy of 88.46%, demonstrating the feasibility of the method.

2.1.2 Local Studies

2.1.2.1 Identification of Diseases in Rice Plant (Oryza Sativa) using Back Propagation Artificial Neural Network

This research focuses on digital image processing to eliminate the subjectiveness of manual inspection of diseases in rice plant. As it is programmed, the research hopes to accurately identify three (3) common diseases of rice plant that Philippine’s farmlands are affected. Image processing is built using MATLAB functions and it compromises techniques such as image enhancement, image segmentation and feature extraction as shown in the image below. The Back Propagation Neural Network is used to enhance the accuracy and performance of image processing (Orillo & Valenzuel, 2014).

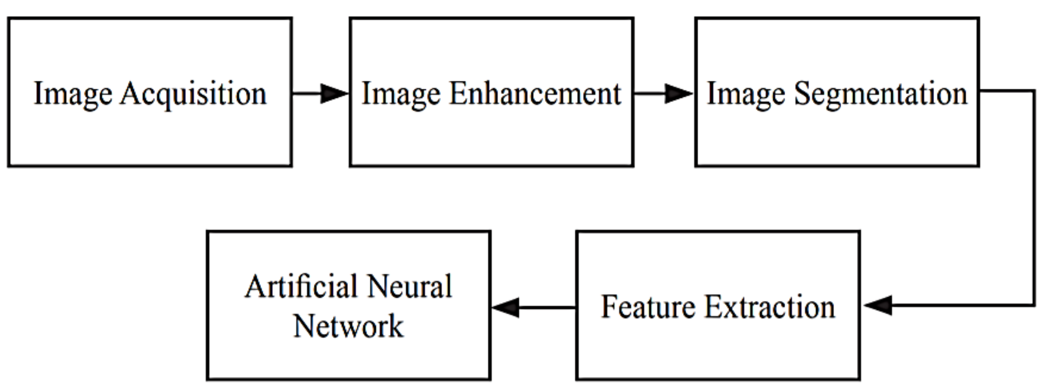


Figure 9. Block Diagram

Involving 134 images of diseases, the database of the network consisted of percentages where, the 70% was used for training, 15% for validation and another 15% for testing. After the processing, the program was expected to give corresponding strategic options to consider when the disease is detected. Overall, it was proved to be 100% accurate.

2.1.2.2 Assessment of Lettuce (Lactuta Sativa) Crop Health using Back Propagation Neural Network

This research paper uses the crop lettuce as its study model. The objective of the study is to develop a simple color recognition algorithm using digital image processing and pattern recognition. Using an Artificial Neural Network (ANN) and Back Propagation Algorithm, both provide a high accuracy and versatility in recognizing the defects in the lettuce based on their colors. ANN is commonly used as a computational model, where it is patterned in the biological neural system of a mammal but in a smaller scale (I. Valenzuela *et al.*, 2018).

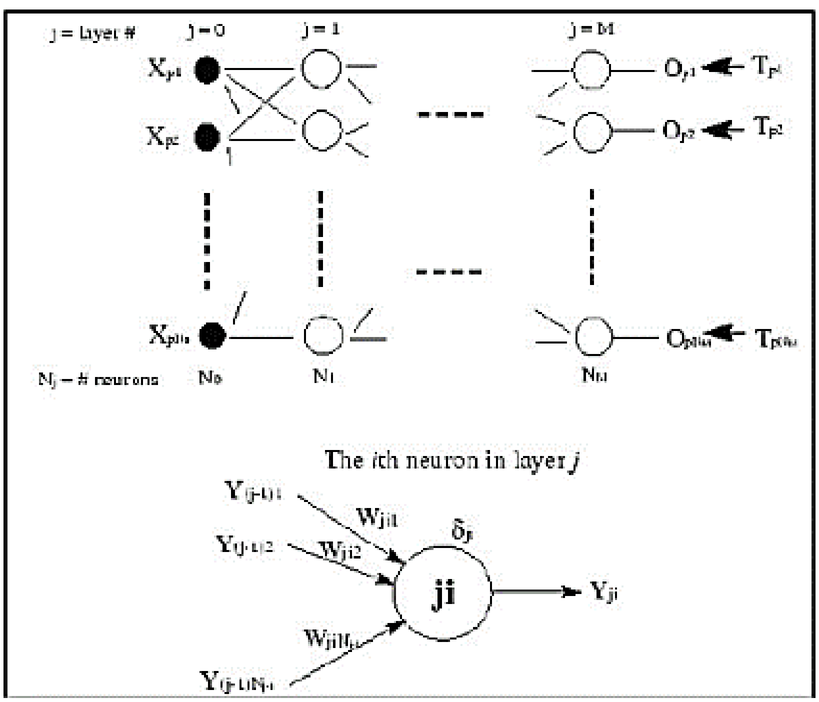


Figure 10. Multilayer Network of Backpropagation ANN

The image shown on Figure 10 is the network algorithm which is composed of layers, namely: input, hidden and output. The organization of these layers makes the network highly capable of predicting the outcomes with high accuracy. Adding the back propagation algorithm to this provides a high accuracy and versatility in recognizing defects in the lettuce crop, based on their colors. This neural network serves as the medium for identifying the quality of the lettuce crop.

Summarizing this, the back propagation neural network was successfully applied on the assessment of lettuce crop health based on its color. The RGB components of each image were extracted by a color feature extraction of LabVIEW. With the given results, the system is marked capable of assessing the health of a lettuce with a minimum square error of 3.2484e-07.

2.2 Related System

2.2.1 Rice Doctor Application powered by Lucid Mobile

It is an application which is for identification of the crop health of the rice itself only. Using the sensors on capturing images, the app will identify what the issue of the captured image of the rice crop is (Lucidcentral).

2.2.2 riceXpert

This application is a system like the Rice Doctor which is built for health recognition for the rice itself. Since it was made for farmers especially from India, since it only discussed the rice issue and not the other grain crops (Google Play).

2.2.3 Features

Features compared are stated below.

Features	Our System	Rice Doctor	riceXpert
Rice	✓	✓	✓
Corn	✓	×	×
Image Recognition	✓	✓	✓
Internet Connection	×	✓	✓
Prevention Method	✓	✓	✓

Table 2. Feature Comparison

2.3 Algorithm

The algorithm used for developing the application, is under the MobileNET algorithm for mobile-first computer vision models for TensorFlow. Designed to effectively maximize accuracy while being mindful of restricted resources for an on-device or embedded application. TensorFlow offers a wide-variety of pre-trained models such as Inception model datasets. However, MobileNet works better with latency, size and accuracy. In terms of output and performance, there is a significant amount of lag with a full-fledged model.

Chapter 3 – Methodology

3.1 Target Users/Stakeholders

The proposed study will aim its research towards the grain farms in Zamboanga City as the system focuses on both rice and corn disease recognition.

3.2 Research Instruments

Research instrument covers the qualitative method, where the researchers went to farms, surveyed and asked questions about farmers' traditional way of gathering information on grain crop disease. Farmers showed different states of rice and corn health from healthy to disease-stricken ones.

3.3 Research Design

This proposed study is an Applied Research Design where it focuses on assisting the target users about grain crop disease recognition. Further information is at Phases of Development.

3.4 Statistical Tools

To cut operational cost and have time efficiency, MS Excel will be utilized for data visualization and simple statistics for the study.

3.4.1 Technical Aspect

The algorithm that the team adapted to use is the MobileNet algorithm from TensorFlow for image classification. MobileNet is a class of CNN that was open-sourced by Google which gives developers an excellent starting point for training classifiers that are insanely small and insanely fast. The reason why it was decided to adapt this algorithm is with respect to the fact that the system relies on gathered images for data classification. This is TensorFlow's first vision model which is designed for mobile application use. It uses depth wise separable convolution layers to build low-latency deep neural networks for mobile and alike devices. Using the depthwise convolution, it can significantly reduce the number of parameters when compared to networks with same depth but regular convolutions. Resulting in lightweight deep neural networks (A. Pujara, 2020).

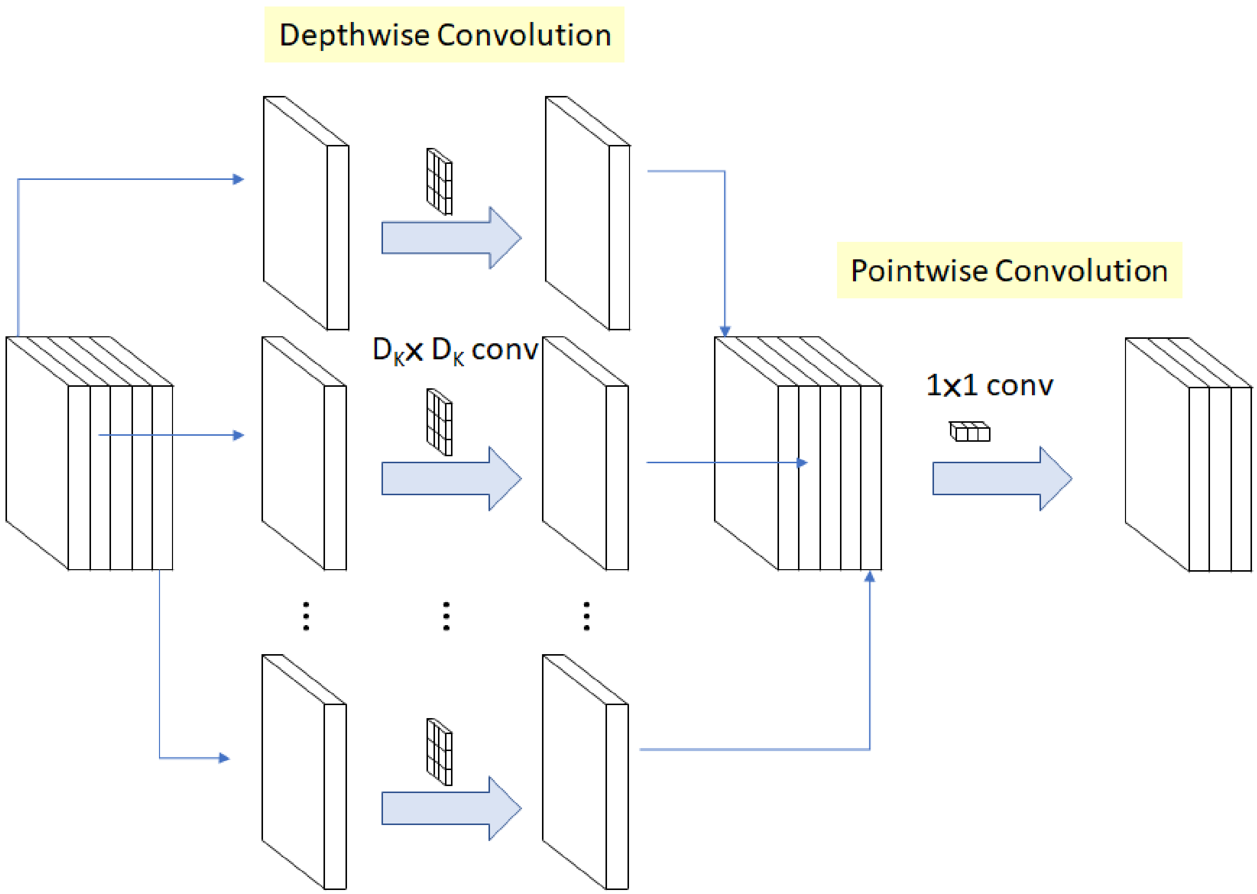


Figure 11. Depthwise Separable Convolution

Depthwise separable convolution is made from two operations: **Depthwise convolution** and **Pointwise Convolution.**

Depthwise Convolution is the channel-wise DK × DK spatial convolution. For an example from Figure 3, we have five (5) channels; then we will have 5 DK × DK spatial convolutions. This is a map of a single convolution on each input channel separately. Using its computational cost **Df * M * Dk**^{2}, its number of output channels are the same as the number of its input channels.

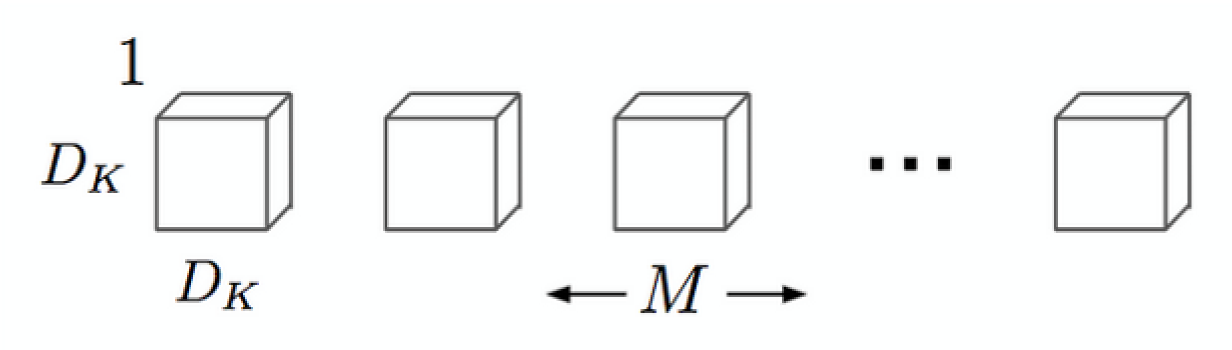


Figure 12. Depthwise Convolution

Pointwise Convolution is the 1 × 1 convolution to alter the dimension. The pointwise simply combines the features of kernel size of 1 × 1 that was created by the depthwise convolution. Computational cost is **M * N * Df²**.

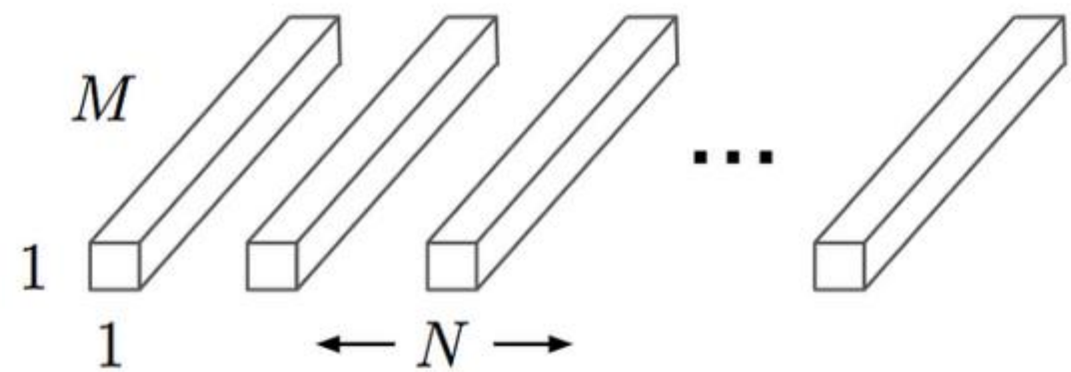


Figure 13. Pointwise Convolution

By adapting this algorithm, we can further analyze with precision from the use of the 1 × 1 pointwise convolution.

3.4.2 Phases of Development

Planning

The system consists of these stages as shown below in Figure 14, where (1) Research and Preliminary Information is gathered from farmers regarding their traditional way of monitoring crop health. Along with research on the algorithm to be adapted and the collection of the dataset. (2) Planning of how to adapt the algorithm and how to utilize the gathered information and the dataset. (3) Software

Development where our team starts on developing the system for the study. (4) Testing of the system and can return to System Development if it requires and lastly, (5) Implementation of the system.

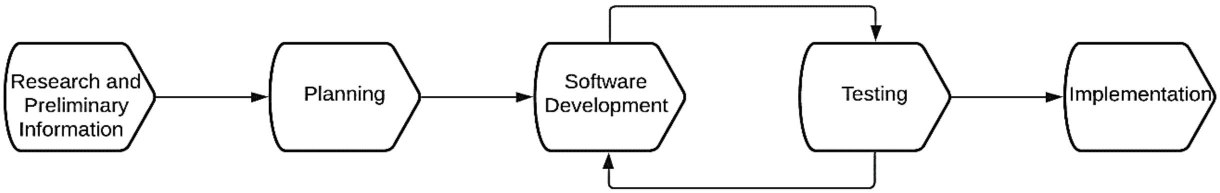
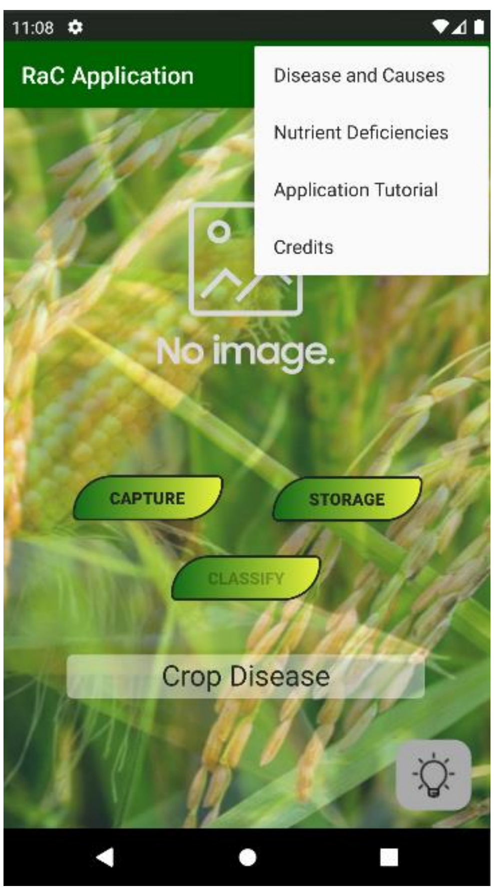
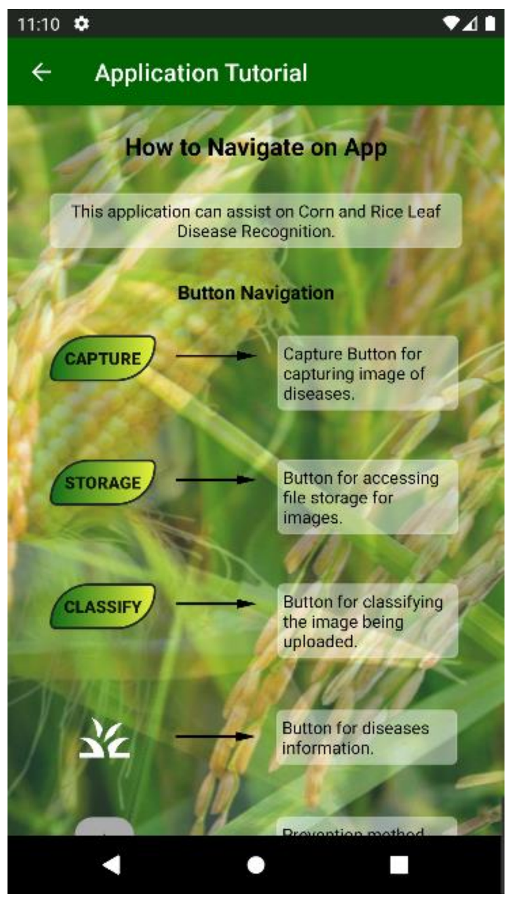


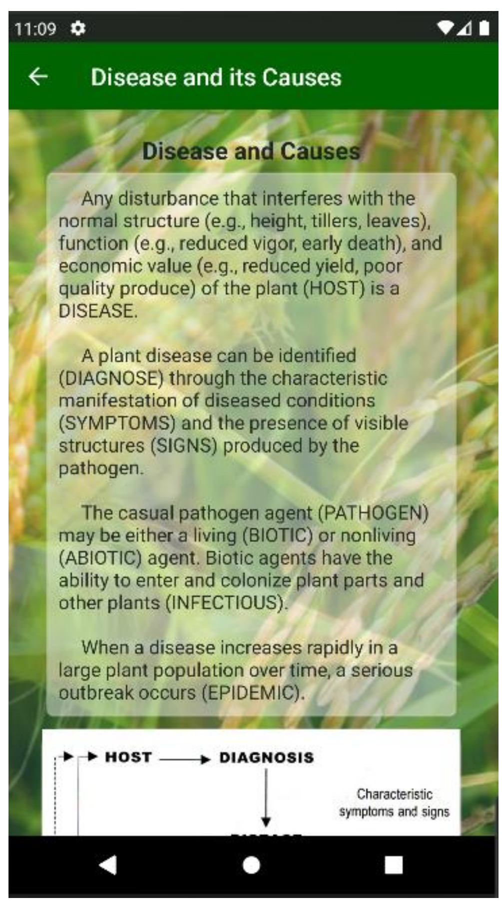
Figure 14. Research Planning






11:09
RaC Application
Nutrient Deficiencies
Nitrogen Deficiency
Symptoms:
• stunted plant
• reduced tiller number
• small, narrow, erect leaves that turn yellowish-red and brown
• old leaves become light straw-colored and wither
Corrective measures:
• Apply nitrogen fertilizer as needed, based on the leaf color chat (LCC)
• Split application of nitrogen to increase efficiency.
Phosphorus Deficiency
Symptoms:
• stunted plant
• reduced tiller number
• erect stem with small, narrow, erect leaves
• young leaves look normal but old ones turn brown and wither
Corrective measures:


11:10
Credits
Developers
Algupera, Aizzy Dianne
Wee, Adzhar
Adviser
Liao, Mara Marie (WMSU)
Instructor
Lines, Marvic (WMSU)
Photos by and from
Aizzy and Wee
kaggle.com
References
http://www.knowledgebank.irri.org/
https://www.gardeningknowhow.com/
https://cals.cornell.edu/
Field Guide by Philippine Rice Research Institute
Special Thanks
Sir Rolando R. Narisma (PhilRice)


11:09
Disease and its Causes
Disease and Causes
Any disturbance that interferes with the normal structure (e.g., height, tillers, leaves), function (e.g., reduced vigor, early death), and economic value (e.g., reduced yield, poor quality produce) of the plant (HOST) is a DISEASE.
A plant disease can be identified (DIAGNOSE) through the characteristic manifestation of diseased conditions (SYMPTOMS) and the presence of visible structures (SIGNS) produced by the pathogen.
The casual pathogen agent (PATHOGEN) may be either a living (BIOTIC) or nonliving (ABIOTIC) agent. Biotic agents have the ability to enter and colonize plant parts and other plants (INFECTIOUS).
When a disease increases rapidly in a large plant population over time, a serious outbreak occurs (EPIDEMIC).
HOST
DIAGNOSIS
Characteristic symptoms and signs

Conceptual Design

The diagram below shows the process of how data is being used and transferred to different stages.

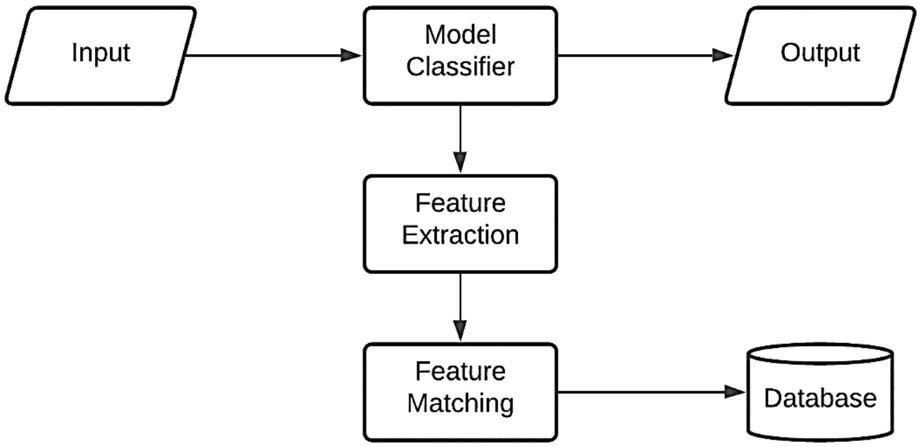


Figure 15. Conceptual Design

Dataset Collection

Images to be used for the dataset was gathered using a mobile phone camera as to not create conflict with image size and pixilation. The rest of the images being used are from kaggle.com. Each crop holds five (5) categories of images including images of healthy crops in which images are ranging from 400-500 per folder. These collected images are then equally converted to provide better feature extraction for the MobileNet algorithm having a dimension of 640px480p each.

- **Hardware, Software and Network Requirements**

  The developed application will work on any mobile android-based device with minimum version of 6.0 (Marshmallow) or higher. The pixels required for better recognition of images must not be lower than 20 megapixels. This system is provided with its own set of images for the dataset, thus leads to the feature of not requiring any internet connection to use the app.

- **Development Tools**

  The system was coded through a laptop that our team provided, running on Intel7 and Ryzen5 for smooth programming. Along with connection to the internet for constant guide in developing the system. The development also included the use of a physical emulator for better testing of the system. The physical emulator used is labeled as a product of POCO under Xiaomi Inc. The usage of this physical emulator was decided by factoring out the system's usability in newly released mobile devices, as most users now own devices with high specifications.

3.4.3 System Architecture

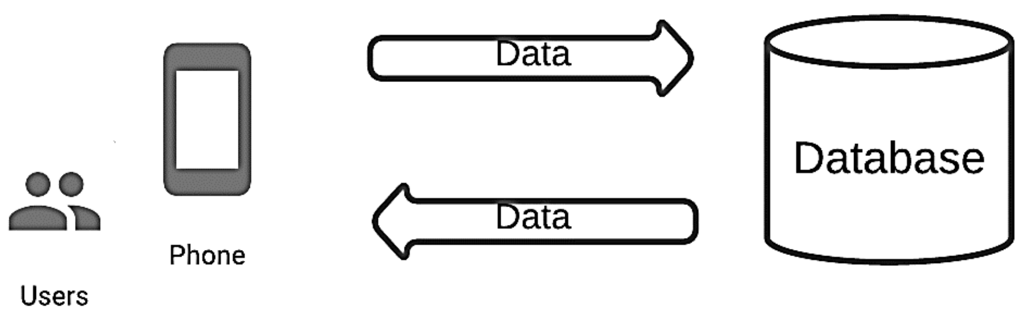


Figure 16. System Architecture

Chapter 4 – Results and Discussions

4.1 Data Gathering

As mentioned in the objectives of this proposal, the images gathered to be used as the main core of the dataset are the diseases commonly found in the Philippines. Rice diseases which are known as being common in the Philippines are the (1) Bacterial Leaf Blight, (2) Brown Spot and (3) Rice Blast (J. Orillo *et al.*, 2014) and on corn crops disease being the 'Peronosclerospora philippinensis' (Philippine downy mildew of maize). This stated crop disease is considered to be the one that majorly affects corn crops in the Philippines (Cueva *et al.*, 2020). Conducting a research and interview to near farms and agriculture offices, the stated Rice Tungro and three (3) more added Corn diseases are what the final research has garnered from the said interview. These interviews were done from the local Agriculture Department of the Municipality of Kabasalan.

Figure 17. Corn Mildew (a)

Corn Leaf Blight (b)

Corn Common Rust (c)

Corn Gray Leaf Spot (d)

Rice Tungro (e)

Rice Brown Spot (f)

Rice Leaf Blast (g)

Rice Leaf Blight (h)

The above images on Figure 17 are the sample images on each disease category within the system model. Some of the corn crop images were taken from a small farm in the Municipality of Kabasalan near a rubber plantation. Rice crop images were from a highway-side farm, stretching from the barangay Lumbayao to barangay Dipala. Each of these images, along with the rest of the image on the dataset have been trained, tested and validated through coding on Google Colab.

Crop Disease	Number of Images within the Dataset	Image Dimension
Corn Common Rust	450	640p x 480p
Corn Gray Leaf Spot	425	640p x 480p
Corn Leaf Blight	461	640p x 480p
Corn Mildew	411	640p x 480p
Healthy Corn	478	640p x 480p
Healthy Rice	400	640p x 480p
Rice Brown Spot	400	640p x 480p
Rice Leaf Blast	433	640p x 480p
Rice Leaf Blight	407	640p x 480p
Rice Tungro	443	640p x 480p

Table 3. Dataset Image Characteristics

The characteristics of the images per category of the dataset are shown in Table 3. These images, both locally-taken and collected from kaggle.com were mass converted to a same pixel dimension.

4.2 Data Analysis

For the data analysis, the starting was to create the model for the basis of the system. As mentioned above in the research paper, the basis for this model was all the images, merged together with different category folders per diseases as stated under Phases of Development. To further give an understanding on how to manage these images and create the model to be used, the research team decided to train different sets of images from the collection, as a form of trial and analyzation. This decision was to give insights on how to handle the image set and how they are to be used inside the model itself. After deliberation, the first set of images that was used were all the images under the 'corn disease'. These images were trained, tested and finalized using Google Colab. After writing the set of codes for creating a model, the result outputted a high prediction accuracy of over 90%. This entails that a model was successfully created but is nowhere near completion, as shown on Figure 18.

```
=================================================================
Total params: 3,418,148
Trainable params: 5,124
Non-trainable params: 3,413,024
_________________________________________________________________
None
/usr/local/lib/python3.7/dist-packages/keras/optimizer_v2/gradient_descent.py:102: UserWarning: The `lr` argument is deprecated, use `learning_rate` instead.
  super(SGD, self).__init__(name, **kwargs)
Epoch 1/5
117/117 [==============================] - 156s 1s/step - loss: 0.6633 - accuracy: 0.8467
Epoch 2/5
117/117 [==============================] - 156s 1s/step - loss: 0.5339 - accuracy: 0.9204
Epoch 3/5
117/117 [==============================] - 151s 1s/step - loss: 0.5177 - accuracy: 0.9274
Epoch 4/5
117/117 [==============================] - 151s 1s/step - loss: 0.5041 - accuracy: 0.9375
Epoch 5/5
117/117 [==============================] - 151s 1s/step - loss: 0.4958 - accuracy: 0.9434
14/14 [==============================] - 34s 1s/step - loss: 0.5036 - accuracy: 0.9332
```

Figure 18. Model Creation

The Figure 19 below shows the process of how the model was created. The model was trained, and evaluated for its accuracy and loss from the images being processed. In this section, inputting into the Google Colab, a set example of prediction accuracy and output of the model. Stating only a hundred images for testing of data prediction accuracy. Showing the right predictions in black label and wrong predictions in red label.

```
[ ] # A helper function that returns 'red'/'black' depending on if its two input
    # parameter matches or not.
    def get_label_color(val1, val2):
      if val1 == val2:
        return 'black'
      else:
        return 'red'

    # Then plot 100 test images and their predicted labels.
    # If a prediction result is different from the label provided label in "test"
    # dataset, we will highlight it in red color.
    plt.figure(figsize=(20, 20))
    predicts = model.predict_top_k(test_data)
    for i, (image, label) in enumerate(test_data.gen_dataset().unbatch().take(100)):
      ax = plt.subplot(10, 10, i+1)
      plt.xticks([])
      plt.yticks([])
      plt.grid(False)
      plt.imshow(image.numpy(), cmap=plt.cm.gray)

      predict_label = predicts[i][0][0]
      color = get_label_color(predict_label,
                              test_data.index_to_label[label.numpy()])
      ax.xaxis.label.set_color(color)
      plt.xlabel('Predicted: %s' % predict_label)
    plt.show()
```

Figure 19. Model Training

Figure 20 on the next page is the set of images used as an example from trained images of the model.

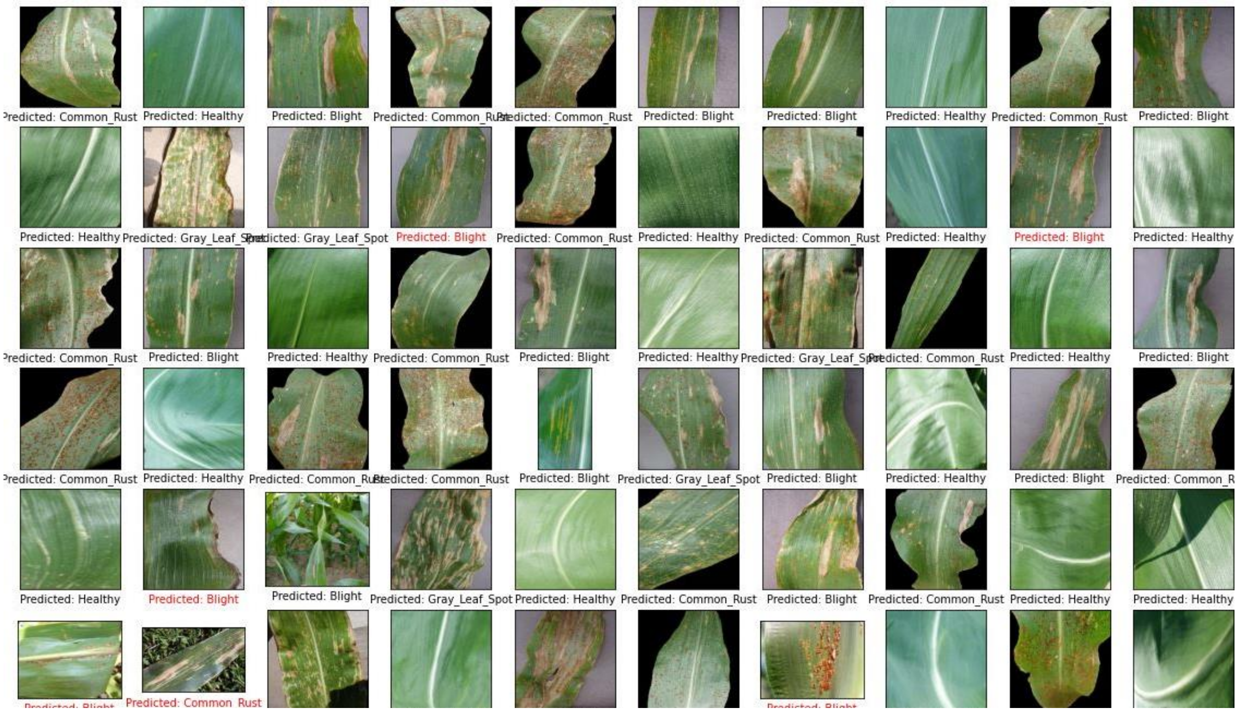


Figure 20. Training Output

From the first trial set namely 'corn diseases', our research moved to another practice step of adding 'Rice' and 'Corn' folder labels where the stated diseases will be moved onto, as shown on Figure 21.

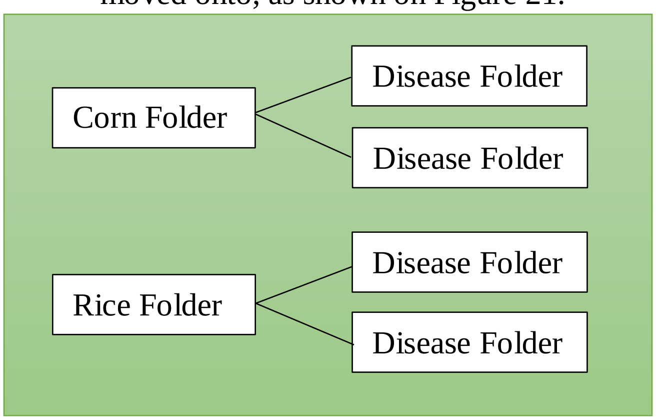


Figure 21. Categorization Example

However, this resulted in an error of classifying of images for the dataset, as shown on Figure 22. It shows that the images inside the disease folders are cannot be read and must be placed directly under the main folder to allow the program to identify the images for classification. This ended with removing the main folder of classification and just directly placing all the disease folders into one main folder.

```
---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)
<ipython-input-3-dfff606fa399> in <module>()
      7
      8 # Customize the TensorFlow model.
----> 9 model = image_classifier.create(train_data)
     10
     11 # Evaluate the model.

                                   1 frames
/usr/local/lib/python3.7/dist-packages/tensorflow_examples/lite/model_maker/core/task/image_classifier.py in train(self, train_data, validation_data, hparams, steps_per_epoch)
    165                         'than batch_size (%d). To solve this problem, set '
    166                         'the batch_size smaller or increase the size of the '
--> 167                         'train_data.' % (len(train_data), hparams.batch_size))
    168
    169     train_ds = train_data.gen_dataset(

ValueError: The size of the train_data (9) couldn't be smaller than batch_size (32). To solve this problem, set the batch_size smaller or increase the size of the train_data.
```

Figure 22. Error Runtime

4.3 Data Finalization

After the trials for the different sets of images, the research proceeded to create the model to be used for the system. In this section, all the images were now called to create the model. The process is the same as the trials, however the prediction accuracy of model was at 70%. The reason for this was because before the conversion of all the images into same dimension and pixel, the images gathered were raw, with different pixels, dimension and above all image file size. The differences of these images stem from where and how they were taken.

```
[ ] loss, accuracy = model.evaluate(test_data)

    24/24 [==============================] - 262s 4s/step - loss: 0.9044 - accuracy: 0.7894
```

Figure 23. Model Loss and Accuracy

A solution for this included a code that will quantize the dataset along with converting them into same pixel dimension. Quantizing the dataset is from the fact that the MobileNet algorithm outputs a high prediction accuracy when the images being included in the dataset does not exceed its favorable quantity for prediction accuracy (U. Kukarni *et al.*, 2021).

```
[ ] config = QuantizationConfig.for_float16()
```

Figure 24. Quantization

With this, the research reached the final step for the creation of the model. After inputting all the codes that was required, the percentage accuracy and loss were not that far from each other and the 80% mark was also met. In actuality, the model outputted a prediction accuracy of 95% as shown on Figure 25.

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 hub_keras_layer_v1v2 (HubKe  (None, 1280)             3413024
 rasLayerV1V2)

 dropout (Dropout)           (None, 1280)              0

 dense (Dense)               (None, 10)                12810

=================================================================
Total params: 3,425,834
Trainable params: 12,810
Non-trainable params: 3,413,024
_________________________________________________________________
None
Epoch 1/5
143/143 [==============================] - 599s 4s/step - loss: 1.0849 - accuracy: 0.7712
Epoch 2/5
143/143 [==============================] - 188s 1s/step - loss: 0.7752 - accuracy: 0.9163
Epoch 3/5
143/143 [==============================] - 196s 1s/step - loss: 0.7362 - accuracy: 0.9312
Epoch 4/5
143/143 [==============================] - 192s 1s/step - loss: 0.7103 - accuracy: 0.9432
Epoch 5/5
143/143 [==============================] - 194s 1s/step - loss: 0.6946 - accuracy: 0.9489
16/16 [==============================] - 80s 3s/step - loss: 0.6736 - accuracy: 0.9529
```

Figure 25. Final Training

On Figure 26 below is the final training output of the model. These training images show a 5% prediction accuracy miscalculation within the dataset.

Figure 26. Final Training Output

4.4 System Results

Along with how to prevent these diseases, the application also has information on every registered disease within the system which gives users the knowledge they would need on managing the crop diseases.

```
[11] model.evaluate_tflite('model.tflite', test_data)

     {'accuracy': 0.9529411764705882}
```

Figure 27. Final Accuracy

Test runs were done by experimenting on different types of disease-stricken leaves of rice and corn plants using the custom modifier. All the images within the model, results in an 95% accuracy, giving us only a 5% miscalculation on recognition of diseases. However, taking into note that depending on how the image was taken and how close the specimen to the camera, the results mostly fall under 5% miscalculation. The system also accepts images that are totally unrelated to the trained images and still gives a result of disease classification. The reason for this is because the image classification algorithm recognizes the features that it has extracted from the image being classified, thus giving out results that may have similarities to the features from the dataset.

All these test runs are programmed within Google Colab and run on AMD Ryzen 5 2500U with a clock of 2.00GHz and a RAM of 12GB in a windows 10 environment.

Chapter 5 – Conclusion and Recommendation

5.1 Conclusion

After the testing of selected/captured images, the team concluded that in order to add more diseases that the system can predict, the system must undergo on a dataset modification along with its model. To do this, one must collect and train images all over again before integrating it to the system. The system solely relies on the trained images of the dataset which brings us to the quantization of the model that has an overall accuracy of 82%. It can definitely use a more stable model type and a larger quantization to fully house the images within the dataset the team have used. However, the model still delivered outstanding results, making the system work within the expected outcome.

5.2 Recommendation

This paper presents a system for Rice and Corn Disease Classification based on a deep learning MobileNet algorithm which is a Convolutional Neural Network. Taking into account that this is an E-learning app, the system will be of great use to the general population who wishes to learn more about the diseases that strike the golden crops of the country and to those who are entering the field of agriculture with knowledge or next to no knowledge on crop diseases. Using MobileNet as the algorithm gives us an efficient and quick way of getting great results. As a team, we recommend the use of MobileNet in any projects that require the algorithm of an image classification and that you explore similar mobile application that offers free use of apps to acquire more knowledge on this front.

References

Abhijeet Pujara (July 4, 2020). *Image Classification with MobileNet*

https://medium.com/analytics-vidhya/image-classification-with-mobilenet-cc6fbb2cd470

Anonymous (Not Stated). *Traditional Foods and their Geographical Relevance*

https://owt-thephilippines.weebly.com/staple-foods.html

Cedeño (January 28, 2021). *Zamboanga Folks Urged to Engage in Gardening*

https://www.sunstar.com.ph/article/1884162/Zamboanga/Local-News/Zamboanga-folks-urged-to-engage-in-gardening

Devika Nair (N.A). *NasNet – A Brief Overview*

https://iq.opengenus.org/nasnet/

Google Play

riceXpert - Apps on Google Play

J. Plant Pathol Microbiol (May 24, 2021). *Effects of Environmental Factors on Crop Diseases Development*

https://www.walshmedicalmedia.com/open-access/effects-of-environmental-factors-on-crop-diseases.pdf#:~:text=Environmental%20factors%20are%20triggering%20simultaneous%20effects%20on%20the,properties%20%28pH%20and%20nutrients%29%20and%20atmospheric%20carbon%20dioxide.

Ira Valenzuela, Argel Bandala & Elmer Dadios (August 13, 2018). *Assessment of Lettuce (Lactuta Sativa) Crop Health using Back Propagation Neural Network*

https://scinet.science.ph/union/UploadFiles/download.php?b=Fil-363345_363345.pdf&f=../Downloads/Fil-363345_363345.pdf&t=application/pdf

IRRI, Seed Health Unit (Patria Gonzales, Evangeline Gonzales, Carlos Huelma, Myra Almodiel, Joel Dumlao), Los Banos, Philippines (Not Stated). *Crop Genebank Knowledge Base* https://cropgenebank.sgrp.cgiar.org/index.php/management-mainmenu-434/stogs-mainmenu-238/rice/guidelines/fungi

Jeanne Grunert (N.A). *4 Plant Growth Factors that Affect All Plants* https://garden.lovetoknow.com/garden-basics/plant-growth-factors

John William Orillo & Ira Valenzuela (November 2014). *Identification of Diseases on Rice Plant (Oryza Sativa) using Back Propagation Artificial Neural Network* https://www.researchgate.net/publication/281148745_Identification_of_Diseases_in_Rice_Plant_Oryza_Sativa_using_Back_Propagation_Artificial_Neural_Network

Lucid Central https://www.lucidcentral.org/editors-pick-crop-diagnostic-keys/rice-doctor/

Marylene Eder (December 2016). *Mobile aRCEE Checker an Application of Rice and Corn Checker for Nutrient Deficiency through Leaf Coloration* https://indjst.org/download-article.php?Article_Unique_Id=INDJST10895&Full_Text_Pdf_Download=True

Mogato (July 26, 2018). *In Philippines, Technology is Seeping into Agriculture* https://www.bworldonline.com/in-the-philippines-technology-is-seeping-into-agriculture/

Palay and Corn Quarterly Bulletin https://psa.gov.ph/sites/default/files/Palay%20and%20Corn%20Quarterly%20%20Bulletin%2C%20October%202020.pdf

Popular Networks (N.A). *VGG16 – Convolutional Neural Network for Classification and Detection* https://neurohive.io/en/popular-networks/vgg16/

Rafeed Rahman, Preetom Saha Arko, Mohammed Eunus Ali, Mohammad Ashik Iqbal Khan Khan (December 2018). *Identification and Recognition of Rice Diseases and Pests Using Deep Convolutional Neural Networks*

https://www.researchgate.net/publication/329413922_Identification_and_Recognition_of_Rice_Diseases_and_Pests_Using_Deep_Convolutional_Neural_Networks

Shaping Agriculture

https://www.cropscience.bayer.com/innovations/data-science/a/digital-tools-for-farmers

S. Savary, S. Bregaglio, L. Willocquet, D. Gustafson, D. Mason D'Croz, A. Sparks, N. Castilla, A. Djurle, C. Allinne, Mamta Sharma, V. Rossi, L. Amorim, A. Bergamin, J. Yuen, P. Esker, Neil McRoberts, J. Avelino, E. Duveiller, J. Koo & K. Garrett (March 17, 2017). *Crop Health and its Global Impacts on the Components of Food Security*

https://link.springer.com/article/10.1007/s12571-017-0659-1#:~:text=The%20phrase%20%E2%80%9Ccrop%20health%E2%80%9D%20is,plant%20physiology%20and%20crop%20performances

Wikipedia (N.A). *Sensor*

https://en.wikipedia.org/wiki/Sensor

Shark Keeper

http://sharkkeeper.com/can-sharks-get-sick-or-diseases/?fbclid=IwAR2Ms5eRO0yTvDiYzYsFspk_sk69461ftDvEPDeU9GefYugnbhOzimUAp4o#:~:text=All%20living%20things%20are%20prone%20to%20diseases.%20Compared,make%20sharks%20perfect%20and%20completely%20immune%20to%20diseases

Terri Queck-Matzie (October 9, 2019). *Successful Farming*

https://www.agriculture.com/technology/technology-adds-efficiencies-in-counting-plants-and-assessing-early-stage-crops

Sumita Mishra, Rishabh Sachan, Diksha Rajpal (January 2020). *Deep Convolutional Neural Network based Detection System for Real-time Corn Plant Disease Recognition* https://www.researchgate.net/publication/340711014_Deep_Convolutional_Neural_Network_based_Detection_System_for_Real-time_Corn_Plant_Disease_Recognition

Thomas Wood (N.A). *Convolutional Neural Network* https://deepai.org/machine-learning-glossary-and-terms/convolutional-neural-network#:~:text=A%20convolutional%20neural%20network%20is%20a%20special%20kind,node%20in%20the%20first%20layer%2C%20and%20so%20on.

Toshichika, & N. Ramankutty (March, 2015). *How do Weather and Climate Influence Cropping Area and Intensity?* https://www.sciencedirect.com/science/article/pii/S2211912414000583#!

Uday Kulkarni, Meena S.M., Sunil V. Gurlahosur, Gopal Bhogar (April, 2021). *Quantization Friendly MobileNet (QF-MobileNet) Architecture for Vision Based Application on Embedded Platforms* https://www.sciencedirect.com/science/article/abs/pii/S0893608020304470

Wikipedia (N.A). *SqueezeNet* https://en.wikipedia.org/wiki/SqueezeNet

Appendices

Appendix A – Prototype

Figure 28. Prototype Images

Appendix B – Plagiarism Report

Figure 29. Originality Report

Document Viewer

Turnitin Originality Report

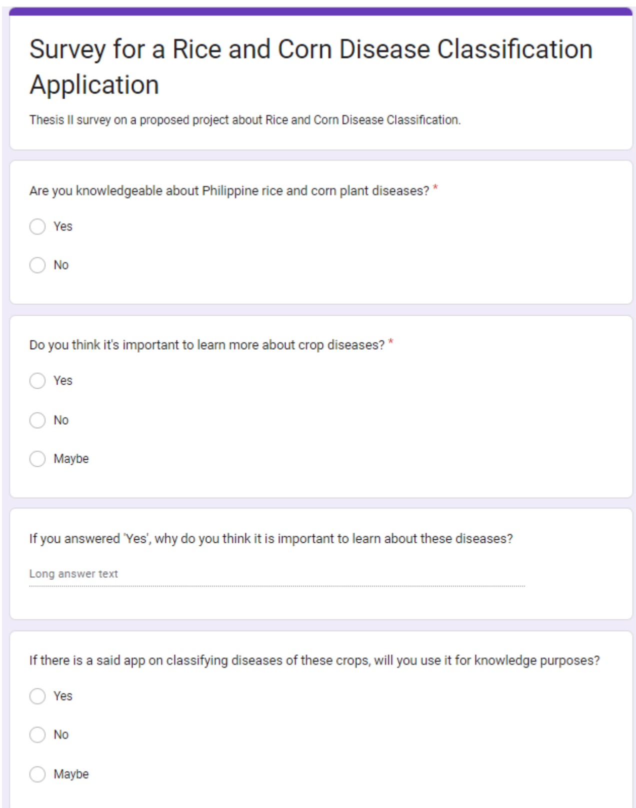
Processed on: 23-Jul-2021 04:08 PST
ID: 1592969163
Word Count: 2290
Submitted: 4

RaC App: A Rice and Corn Health Recognition M... By Aizzy Dianne Algupera

Similarity Index	Similarity by Source	
18%	Internet Sources:	15%
	Publications:	9%
	Student Papers:	12%

Appendix C – Survey Form

Figure 30. Survey Form



Survey for a Rice and Corn Disease Classification Application

Thesis II survey on a proposed project about Rice and Corn Disease Classification.

Are you knowledgeable about Philippine rice and corn plant diseases? *

- Yes
- No

Do you think it's important to learn more about crop diseases? *

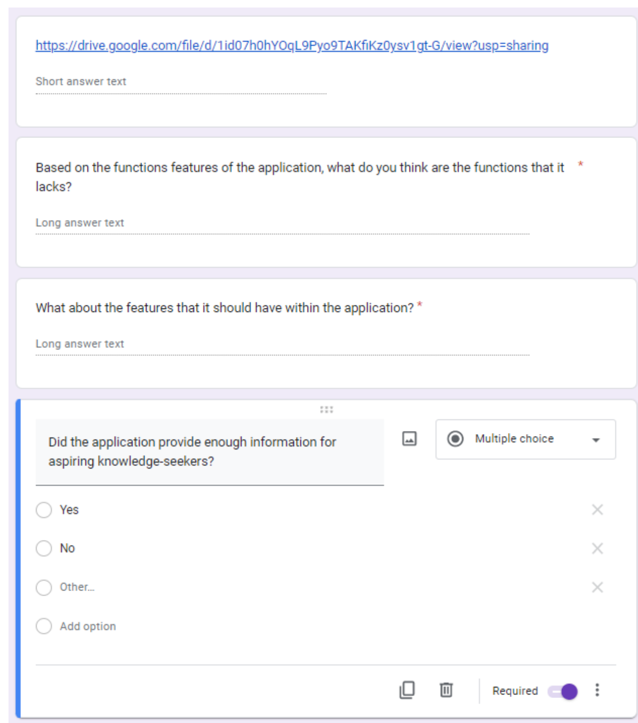
- Yes
- No
- Maybe

If you answered 'Yes', why do you think it is important to learn about these diseases?

Long answer text

If there is a said app on classifying diseases of these crops, will you use it for knowledge purposes?

- Yes
- No
- Maybe



https://drive.google.com/file/d/1id07h0hYOqL9Pyo9TAKfiKz0ysv1gt-G/view?usp=sharing

Short answer text

Based on the functions features of the application, what do you think are the functions that it lacks? *

Long answer text

What about the features that it should have within the application? *

Long answer text

Did the application provide enough information for aspiring knowledge-seekers?

Multiple choice

- Yes
- No
- Other...
- Add option

Required

Appendix D – Proofread and Validation

Republic of the Philippines
Western Mindanao State University
College of Computing Studies
DEPARTMENT OF COMPUTER SCIENCE
Zamboanga City

CERTIFICATE OF VALIDATION

THIS IS TO ACKNOWLEDGE THAT THE THESIS ENTITLED

RICE AND CORN DISEASE RECOGNITION E-LEARNING MOBILE APPLICATION USING MOBILENET MACHINE LEARNING ALGORITHM

WRITTEN BY

AIZZY DIANNE ALGUPERA
and
ADZHAR WEE

HAS BEEN APPROVED THAT THE THESIS APPLICATION PROVES ITS ABILITY TO CLASSIFY IMAGES WITH REGARDS TO CROP DISEASES, CERTIFYING ITS BENEFITS BY THE UNDERSIGNED, ON THE DATE OF

8/3/2022

ROLANDO R. NARISMA
Validator

Republic of the Philippines
Western Mindanao State University
College of Computing Studies
DEPARTMENT OF COMPUTER SCIENCE
Zamboanga City

CERTIFICATE OF PROOFREADING

THIS IS TO ACKNOWLEDGE THAT THE THESIS ENTITLED

RICE AND CORN DISEASE RECOGNITION E-LEARNING MOBILE APPLICATION USING MOBILENET MACHINE LEARNING ALGORITHM

WRITTEN BY

AIZZY DIANNE ALGUPERA
and
ADZHAR WEE

HAS BEEN PROOFREAD FOR APPROPRIATE ENGLISH LANGUAGE USAGE, GRAMMAR, PUNCTUATION, AND SPELLING BY THE UNDERSIGNED AND RETURNED TO THE CUSTOMER ON

23 SEPTEMBER 2022

ROSALIE Y. WEE
Proofreader