Agricultural Rice Leaf Diseases Detection Using Convolutional Neural Networks

Jophet Hernandez
College of Computer Science and Engineering
Jose Rizal University
Mandaluyong, Philippines
jophet.hernandez@my.jru.edu

Johnmark Guillero
College of Computer Science and Engineering
Jose Rizal University
Mandaluyong, Philippines
johnmark.gullero@my.jru.edu

Glen William Perez
College of Computer Science and Engineering
Jose Rizal University
Mandaluyong, Philippines
glenwilliam.perez@my.jru.edu

Abstract:

In this study, we delved into the learning process and performance of various VGG16 models, aiming to optimize their accuracy in classifying rice leaf diseases. Our analysis highlighted the impact of epochs, test sizes, and validation splits on model accuracy. We observed that more epochs generally yield higher accuracy, and finding the right balance between test size and epochs is crucial. However, some combinations of parameters worked unexpectedly, indicating the need for further investigation.

Our experiments encountered challenges, notably the significant time investment for comprehensive model testing and fine-tuning hyperparameters. Balancing predictive accuracy with operational efficiency, influenced by data volume and processing time. Navigating these complexities informed our efforts to refine our modeling approach for more effective disease detection in rice leaves. This study contributes insights into optimizing model performance and challenges in image classification.

Keywords: (Convolutional Nueral Network, Hyperparameters, Fine-tuning, hold out, VGG16)

I. Introduction

In the Philippines, rice is one of the important parts of the economy. Around 11.5 million people depend on growing rice for a living. Rice significantly affects the economy like Consumer Price Index, Gross Value Added, and Gross Domestic Product. [1] Many Filipinos make a living in farming. However, farmers face many challenges because rice plants often get diseases, especially on their leaves. It reduces the amount of rice that can be grown, which impacts the needs of a growing population. Identifying rice leaf diseases can help farmers take early action so that they can prevent the worst possible outcome and reduce crop loss. [2]

A study from India entitled "Custom Convolutional Neural Network for Detection and Classification of Rice Plant Diseases." Helps the farmers to identify rice leaf diseases. The researchers built their custom dataset about different images of rice leaf diseases, providing an accuracy of 99.83%. The dataset has 1400 pictures of other conditions of rice leaves. Using a custom build dataset, allows the researchers to gain advantages like tailoring the specific needs of the project. And have overall control over the data. However, based on their research, there are some gray areas, like it only depends on the project scope and is only limited to the specific country. It is also time-consuming to collect various data to be used in the project. [3]

A study focusing on the detection of rice leaf diseases utilized a pre-existing dataset. The researchers encountered challenges with unbalanced raw samples within the dataset they collected. They highlighted the time-saving aspect of using a pre-built dataset, as it was readily available for their study. However, they also faced drawbacks, including issues related to image quality and the dataset's lack of customization, especially in addressing specific areas or aspects relevant to their research. [4]

Similar to India and other developing nations, rice is a vital crop in the Philippines, essential for food security and the agricultural economy. Diseases affecting rice plants can significantly reduce crop yield and quality, directly impacting the country's food supply. By utilizing a CNN model trained on a dataset of images about healthy and diseased rice leaves, the Philippines can improve its ability to detect diseases in rice plants. This technology can assist farmers in identifying and categorizing various diseases so that they can take immediate action to reduce crop loss.

The performance of the hold-out method serves the purpose of understanding their distinct advantages and limitations in assessing machine learning model performance. The hold-out method's appeal lies in its simplicity and speed, dividing data into training and testing subsets for quick assessments. However, its reliance on random splitting can lead to variance in results and potential bias if the split isn't representative. [5]

In this study the researchers use a pre-trained CNN called VGG16 it is a really good computer vision model known for its design using small 3x3 filters. This change made a big improvement compared to earlier models. They made VGG16 deeper, having around 16 to 19 layers, which added up to about 138 parts that could be adjusted to improve its performance. [8] This pre-built model reduces the training time and can be fine-tuned on smaller datasets.

Research Objectives:

1. Conduct an in-depth analysis of the Rice Leaf Diseases Detection domain to identify key features and challenges,
and collect relevant data for the development of a specialized CNN-based model for single-object image classification.
2. To develop a specialized CNN-based model for single-object image classification in the Agricultural domain, with a focus on improving Rice Leaf disease detection.
3. To examine the practical utility of the developed model by assessing its efficiency.

II. Literature Review

The limited experience and knowledge of farmers make manual disease recognition difficult and time-consuming. That is why a group of researchers conducted a study entitled "Rice leaf diseases classification using CNN with transfer learning" about identifying different diseases that affect rice plants in India. The study is also about the effectiveness of automated image recognition systems, specifically utilizing convolutional neural network models to help fix the problem of rice farmers. The researchers built their dataset, which would be used for learning the model since there are very few, if any, existing datasets for diseases of rice leaves. The developed deep learning model utilizes transfer learning, based on the VGG-16 architecture, and undergoes training and testing using datasets sourced from paddy fields and the internet. The researchers trained the model with 1509 images of rice leaves and tested on 647 distinct images, resulting in a good classification rate of 92.46%. The application of transfer learning, particularly the configuring of the predefined VGGNet, significantly enhances the model's performance, which is good considering that the dataset that was utilized in training and testing the model was small. To further improve the accuracy of the algorithm, the researchers say they plan to collect more photos from research facilities and agricultural fields. [6]

Lots of rice in the country's food makes more than half of the people sick with these diseases. Bacteria, fungi, and viruses can make rice plants sick, hurting their leaves and limiting their ability to grow and produce food. The research project entitled "Rice Leaf Disease Detection using Machine Learning" aims to help farmers in India manage diseases that harm rice plants. It was possible to find these diseases in this study by looking at shots of the leaves using a special kind of technology called CNN. Their model was trained with a huge collection of smartphone pictures—about 10,800. In 98–99% of cases, their method correctly identified diseases like Brown Spot, Rice Blast, and Hispa. This technology might help farmers find these diseases early and make their crops healthier. Additionally, they believe their system can be improved even better by mixing their pictures with other available images. Nevertheless, they say that more study is needed to improve this method even further, especially when separating the infected leaf parts. Using innovative technology like the Internet and cell phones, they think, will make it much easier to keep an eye on crops and make farming better in the future. [7]

A study titled "Custom Convolutional Neural Network for Detection and Classification of Rice Plant Diseases." Explore the importance of rice cultivation in India and other developing nations. Rice is a big part of Farming in India and is really important for making sure there's enough food for everyone. However, rice plants get diseases, and it affects how much rice can get as well as the quality of the rice. This research suggests utilizing a custom CNN to find and sort out common diseases in rice plants will help detect diseases on rice plants. The researchers trained the custom CNN using a dataset that has 1400 pictures of healthy and unhealthy rice leaves; the unhealthy rice leaves have one of four common diseases in rice plants. The researchers checked how well the program worked using two ways to make it better: Adam: Adaptive Moment Estimation and Stochastic Gradient Descent with Momentum (SGDM). The test findings demonstrated that when utilizing the Adam method, the program was super accurate, getting up to 99.83% right in the 7th round. But when they included the pictures of healthy leaves, the Adam way worked even better than the SGDM way, with a top accuracy of 99.66% and 97.61% in the 7th round, one after the other. This research helps make sure we can find and handle diseases in rice plants better. [3]

A. Summarize and Discuss the Highlights of the related papers.

The related research papers aim to tackle the issue of diseases affecting rice plants by applying machine learning techniques. The first study emphasizes using transfer learning-based CNN models, specifically the VGG-16 architecture, to identify rice leaf diseases. Despite a relatively small dataset, the model achieved a commendable 92.46% classification rate, showcasing the effectiveness of transfer learning even with limited data. The second research project, utilizing a vast dataset of smartphone images, demonstrated an impressive 98-99% accuracy in identifying common rice plant diseases, foreseeing its potential to aid farmers through early disease detection and healthier crop management. The third study focused on a custom CNN, achieving exceptional accuracies up to 99.83% using the Adam method and stressing the importance of accurate disease identification for improved disease management in rice plants. These studies underscore the potential of machine learning in disease identification and its significant role in advancing agricultural practices for better crop health and yield in rice cultivation.

III. METHODOLOGY

A. Present and discuss the dataset used in this experiment and its characteristics

The dataset utilized for this study encompasses five distinct categories of rice leaf diseases: Bacterial Leaf Blight, Brown Spot, Leaf Blast, Rice Hispa, and Sheath Blight. Researchers custom-built this dataset by gathering various images and compiling them to represent each disease category accurately. Each categories contains of 200 images with a total of 1000 images in all.

B. Explain the preprocessing steps applied to clean and prepare the dataset for image classification modeling.

Preparing the dataset for a VGG16-based image classification model involves a systematic series of steps to ensure effective training. Initially, images are loaded using Keras' image.load_img() function, maintaining a consistent size of 224x224 pixels for uniformity. Subsequently, these images undergo conversion into numerical arrays via image.img_to_array(). To align with VGG16's requirements, specific preprocessing is applied, ensuring proper formatting and normalization. Once the images are appropriately prepared, both the preprocessed data and corresponding class labels are collected. To simplify model training and evaluation, a 70-30 split ratio is employed for dataset partitioning, strategically balancing class distributions in training and holdout sets.

The pre-trained VGG16 model, initially trained on a diverse dataset called ImageNet, is incorporated into the workflow. To make it suitable for the specific task of classifying rice leaf diseases, important classification layers such as GlobalAveragePooling2D and Dense softmax layers are integrated. These additional layers help the model understand and categorize features relevant to identifying different types of rice leaf diseases. To tailor it for rice leaf disease classification, key classification layers like GlobalAveragePooling2D and Dense softmax layers are added. Importantly, the convolutional layers are frozen, preserving their pre-trained weights. This deliberate strategy allows the model to focus on training the newly added classification layers while benefiting from the valuable features acquired during pre-training.

The preprocessing steps involve loading, resizing, and transforming images to align with VGG16 requirements. The dataset is strategically split, and the VGG16 model is adapted for rice leaf disease classification, striking a balance between leveraging pre-trained knowledge and training task-specific layers for accurate classification.

C. Discuss the details of the initial CNN architecture that will be used for the image classifier model as well as the performance metrics that will be applied

The initial Convolutional Neural Network (CNN) architecture used in the image classifier model is built upon the well-established VGG16 model, known for its effectiveness in image classification tasks. To enhance its capabilities for classifying rice leaf diseases, the VGG16 model is pre-trained on the ImageNet dataset, leveraging its ability to capture hierarchical features and patterns within images.

The architecture extends beyond VGG16 by incorporating additional layers tailored to the specific task of rice leaf disease classification. These include a GlobalAveragePooling2D layer, which reduces spatial dimensions and aids in extracting essential features, and a Dense layer with 128 neurons employing Rectified Linear Unit (ReLU) activation. The final Dense layer introduces softmax activation to generate class probabilities, facilitating multi-class classification of rice leaf diseases.

In terms of performance metrics, the model is assessed using the categorical cross-entropy loss function, designed for the effective handling of multi-class classification problems. The Adam optimizer is selected for training, and recognized for its adaptability and efficiency in optimizing deep neural networks. Evaluation of the model's classification accuracy is carried out using the accuracy metric during both the training and validation phases. Accuracy, a commonly used metric for classification tasks, provides insights into the overall correctness of the model's predictions.

The training process happens over multiple rounds, with each round using a group of 32 images at a time (batch size). This setup helps the model learn from the dataset step by step. We closely watch how well the model is doing on both the training and validation sets by looking at accuracy and loss. This careful monitoring is crucial to make sure the model is learning well without fitting too closely to the training data (overfitting). After training, we put the model to the test on a different set of images (holdout or test set) to thoroughly check how well it can handle new, unseen data.

The initial CNN architecture builds on the VGG16 model, adding extra layers for the specific job of identifying diseases in rice leaves. The metrics we use, like categorical cross-entropy loss and accuracy, are chosen to match the needs of dealing with multiple classes. The Adam optimizer is in play to make sure the model learns efficiently. The whole process ensures we end up with a strong and well-checked image classifier for rice leaf diseases.

D. Discuss the strategy formulated by the group to adjust the different hyperparameters of the CNN to get the best model. Note: There is no limit to the hyperparameters that will be covered in this exercise.

To enhance the accuracy of a model, researchers often encounter various hyperparameters, including epochs, learning rates, and the number of hidden layers when analyzing Convolutional Neural Networks (CNNs) for image classification. Adjusting these hyperparameters manually requires considerable time and effort, as we have experienced. Despite the time-consuming nature of this process, it proves worthwhile because it allows us to observe how each hyperparameter influences the model's accuracy.

Our approach involves a method commonly referred to as "babysitting," where we carefully tweak individual hyperparameters and observe the resulting impact on the model's performance. This manual adjustment process is time-intensive but valuable, providing insights into which hyperparameters significantly affect the model's accuracy. By isolating a specific hyperparameter and incrementally adjusting it, we can identify the optimal values that maximize accuracy.

While automatic hyperparameter tuning methods exist, we opted against their use due to time constraints and concerns about potential disturbances to the existing code. The manual "babysitting" approach allows us to maintain control over the adjustments and ensures a thorough understanding of each hyperparameter's influence on the model's performance. Once we reach a point where further adjustments do not cause significant accuracy improvements, After tweaking one hyperparameter and seeing how it affects accuracy, we move on to the next, systematically improving the model step by step.

E. Discuss the Excel file format you will use to record the results of each run in your experiment.

Table 1: Experimentation Logs

model	Epoch	Test size	validation split	Accuracy
VGG16_1	5	0.3	0.3	81.00%
VGG16_2	5	0.3	0.3	82.67%
VGG16_3	10	0.3	0.3	83.67%
VGG16_4	5	0.3	0.5	76.33%
VGG16_5	5	0.5	0.5	77.60%
VGG16_6	15	0.3	0.3	84.33%
VGG16_7	20	0.3	0.3	83.67%
VGG16_8	10	0.3	0.3	87.67%

The Excel file format that is used in this experiment is columns that include the model name or identifier, the number of epochs used for training, the test size indicating the proportion of the dataset allocated for testing, the validation split denoting the ratio of data reserved for validation during training, and a column for accuracy metrics achieved by each model. Each row corresponds to a specific model run, detailing its configuration and performance metrics. By employing clear headers and concise labels for columns, this organized layout facilitates easy comparison between different model setups and their corresponding accuracies. This structured format simplifies the analysis process, enabling quick identification of trends or patterns across varying model configurations and aiding in informed decision-making for subsequent experimentation or model selection.

IV. RESULTS

A. Present and discuss the results of your experiments.

In our series of experiments with the VGG16 model, we systematically varied key hyperparameters to assess their impact on model performance. Notably, the number of training epochs appeared to positively correlate with accuracy, with models VGG16_7 and VGG16_8 achieving the highest accuracies at 84.33% and 87.67%, respectively, after 10 epochs for model VGG16_8 and 20 epochs for model VGG16_7. This suggests that prolonged training contributes to improved model learning and predictive capabilities. Surprisingly, adjusting the test size and validation split did not exhibit a consistent trend in accuracy improvement. For instance, while VGG16_3 achieved 83.67% accuracy with a validation split of 0.3, models like VGG16_5, despite having a larger test size and validation split, yielded slightly lower accuracies at 77.60%. These findings suggest that, in our experimental setup, the number of epochs played a more decisive role in model performance than variations in test size or validation split. Further exploration and fine-tuning may be warranted to identify the most optimal configuration for achieving even higher accuracies.

B. *Display and analyze the performance charts of the top-performing model.*

The performance charts for the top-performing model, provide important insights into its learning process. The validation loss exhibits fluctuations across epochs but shows an overall improvement, with a significant drop in the tenth epoch indicating a notable performance boost. In contrast, the training loss consistently decreases, reflecting continuous improvement and learning throughout training.

Examining the accuracy charts, the training accuracy steadily climbs to an impressive 90%. However, the validation accuracy displays a different learning pattern. Each epoch contributes to bumps in accuracy, suggesting the model is adapting and gaining knowledge from the dataset in a non-subt yet progressive manner. This subtle learning process is particularly noticeable in the validation set, highlighting the model's ability to grasp complicated details in the data.

The charts illustrate that VGG16_8 is a top-performing model with both steady and slight learning patterns. The fluctuating validation loss, especially the significant improvement in the tenth epoch, coupled with the continuous decline in training loss, demonstrates the model's capacity to learn and generalize. The accuracy charts further emphasize the model's adaptability, with noticeable bumps in accuracy throughout training, showcasing its ability to comprehend the complexness of the Rice Life dataset.

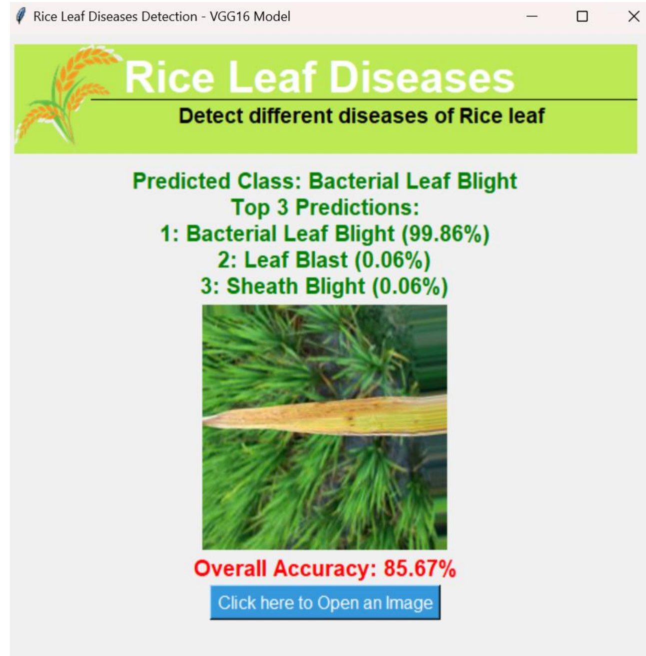

Figure 1 Rice leaf diseases detection GUI

The GUI presented reveals the top 3 predictions derived from the uploaded photo, utilizing a technique known as one-hot coding. This method displays the top 3 categories associated with rice leaf diseases and their probabilities calculated through the softmax function. This tool gives rice farmers an accessible and accurate method of detecting different rice leaf diseases. Farmers can make informed decisions using this technology, promptly addressing concerns and protecting their harvests.

V. ANALYSIS

A. *Interpret the results based on your observation.*

After analyzing Experiment Logs that shows the different models for the VGG16, a few things stand out. First off, more epochs generally mean better accuracy—VGG16_6 with 15 epochs has a higher accuracy (84.33%) compared to VGG16_1 with just 5 epochs (81.00%). The size of the test group matters too; for instance, VGG16_2 with a smaller test size has better accuracy (82.67%) than VGG16_5 with a larger one (77.60%). The impact of the validation split is a bit trickier to pin down. Sometimes it helps (like in VGG16_3 with 10 epochs and an accuracy of 83.67%), but not always (VGG16_4 with similar settings but an accuracy of 76.33%).

Taking a closer look, it seems that some parameters work better together. The highest accuracy happens with VGG16_8, which has 10 epochs and a test size of 0.3 (87.67%). This suggests that finding the right balance between the test size and the number of epochs could be key. But, there are a few curveballs, like VGG16_7 with 20 epochs having a similar accuracy to VGG16_3 with 10 epochs. So, while more epochs usually mean better accuracy, figuring out the perfect combo of test size and validation split requires more tweaking and investigation to really nail down the best settings for the model.

B. *Discuss any challenges or limitations you encountered during the experiments.*

During our experimentation, we encountered a range of challenges that offered valuable insights into the complexities of our image classification project. One remarkable challenge revolved around the significant time required to thoroughly test our model's performance. The combined duration of both training and testing phases averaged around 20-30 minutes, primarily due to the substantial amount of data our model had to handle during the experimentation process.

Another noteworthy challenge originated from the subtle task of fine-tuning hyperparameters. While the careful adjustment of these parameters played a pivotal role in optimizing our model's performance and achieving optimal outcomes, it also posed a challenge by extending the overall running time. Striking the right balance became a delicate effort, where achieving the highest level of predictive accuracy through hyperparameter optimization needed to be carefully considered against the essential need for maintaining operational efficiency.

Furthermore, the complexity of our project lay in navigating the interaction between processing time, the magnitude of data under consideration, and the careful optimization of hyperparameters. Reaching a balanced state among these factors emerged as a crucial focal point, ensuring the efficacy and reliability of our predictive modeling framework for the detection of diseases in rice leaves. These challenges, albeit challenging, provided invaluable insights that directed subsequent refinements and enhancements to our experimental approach.

VI. CONCLUSIONS

Upon reviewing the performance charts of the VGG16 models and diving into the experiment logs, some key observations emerged. First, more epochs generally lead to higher accuracy. Models with more epochs tended to perform better, showcasing higher accuracy rates. Additionally, the size of the test group showed an impact; smaller test sizes often correlated with better accuracy.

However, the influence of the validation split could have been more straightforward. Sometimes, altering the validation split improved accuracy, but this needed to be more consistent across all models. Finding the right balance between test size and epochs seemed crucial, as seen with the highest accuracy achieved by VGG16_8, which had ten epochs and a specific test size.

Despite these patterns, there were curveballs. Some models with more epochs didn't necessarily outperform those with fewer epochs, adding complexity to the relationship between epochs and accuracy. This indicated that pinpointing the perfect combination of test size and validation split required more exploration and fine-tuning.

Throughout the experiments, challenges surfaced, notably the considerable time required for testing due to the extensive dataset. Fine-tuning hyperparameters was another hurdle, demanding a delicate balance between optimizing predictive accuracy and maintaining operational efficiency. The interplay between processing time, data volume, and hyperparameter optimization proved complex. Yet, it underscored the need for a balanced approach to ensure a reliable predictive model for disease detection in rice leaves. While more epochs generally improved accuracy and smaller test sizes often led to better performance, finding the precise configuration of parameters remains ongoing . The challenges encountered provided valuable insights, guiding refinements for a more effective experimental approach in the future.

REFERENCES

[1] "BRIDGING THE RICE YIELD GAP IN THE PHILIPPINES - Leocadio S. Sebastian,* Pedro A. Alviola,** Sergio R. Francisco***." https://www.fao.org/3/X6905E/x6905e0b.htm

[2] "Detection of rice leaf diseases using image processing," IEEE Conference Publication | IEEE Xplore, Mar. 01, 2020. https://ieeexplore.ieee.org/abstract/document/9076527

[3] S. P. Singh, K. Pritamdas, K. J. Devi, and S. D. Devi, "Custom Convolutional Neural Network for Detection and Classification of Rice Plant Diseases," PDF, 2023. Accessed: Nov. 22, 2023. [Online]. Available: https://www.irjet.net/archives/V9/i8/IRJET-V9I8256.pdf?fbclid=IwAR3dL7QKUsbPZ3Fs7fd3k5VZVgkLZfV1W0RDdbJ0WNPbqZMMkP02uaNIrpM

[4] Haruna, Y., Qin, S., & Mbyamm Kiki, M.J. (2023). An Improved Approach to Detection of Rice Leaf Disease with GAN-Based Data Augmentation Pipeline. *SSRN Electronic Journal*.

[5] Yadav, S., & Shukla, S. (2016). Analysis of k-Fold Cross-Validation over Hold-Out Validation on Colossal Datasets for Quality Classification. *2016 IEEE 6th International Conference on Advanced Computing (IACC)*, 78-83.

[6] P. Haritha, "Rice leaf diseases classification using CNN with transfer learning," PDF, 2022. Accessed: Nov. 22, 2023. [Online]. Available: https://www.irjet.net/archives/V9/i8/IRJET-V9I8256.pdf?fbclid=IwAR3dL7QKUsbPZ3Fs7fd3k5VZVgkLZfV1W0RDdbJ0WNPbqZMMkP02uaNIrpM

[7] T. Chavan, D. B. Lokhande, and D. P. Patil, "Rice Leaf Disease Detection using Machine Learning," International Journal of Scientific Research in Computer Science, Engineering and Information Technology, pp. 575–583, Dec. 2022, doi: 10.32628/cseit228677.

[8] Rohini G. Sep. 23, 2021 "Everything you need to know about VGG16". Available:https://medium.com/@mygreatlearning/everything-you-need-to-know-about-vgg16-7315defb5918.