International Journal for Multidisciplinary Research (IJFMR)

E-ISSN: 2582-2160 ● Website: www.ijfmr.com ● Email: editor@ijfmr.com

A Web Application that Classifies Rice Crop Stress with Recommendation System

Karren Vitalicio De Lara[1], Maksuda Sultana[2]

[1]Asst. Professor II, AMA University
[2]Professor, AMA University

Abstract

It provides for the development and evaluation of a web-based application meant to aid in the identification of rice crop stress and possible mitigation measures. Traditional methods have been used, but this approach should replace the manual classification that is time-consuming, tedious, and complex, as it leverages advanced machine learning algorithms to analyze images of rice crops, resulting in the accurate detection of various stress factors such as nutrient deficiencies, pest' infestations, and water stress. In order to ensure the system has high accuracy and reliability for real-world situations, training and testing were conducted using an inclusive dataset that contained labeled images that featured different kinds of stressed rice crops. Also, besides the classification of stress, the app recommends specific advice for particular stress types, such as fertilization, pest control, and irrigation techniques. The recommendations are generated from a massive database with expert agricultural tips and current research to ensure that they give accurate and practical solutions. The usability and effectiveness of the application were assessed through field trials with local farmers and agricultural experts. Results indicated a significant improvement in the early detection and management of crop stress, leading to increased yields and resource efficiency. This web application aims to empower farmers with timely and accurate information, foster sustainable agricultural practices, and enhance food security.

Keywords: rice crop stress, machine learning, image analysis, recommendation system, sustainable agriculture, web application.

1. INTRODUCTION

The current challenges faced by the Department of Agriculture Region IV-B (MIMAROPA) is related to the manual input and slow process of photo submissions by farmers to detect the rice crop stress and possible pest infections. The response time is delayed to around one to two weeks after photo submission, and thus rice crop stress and disease detection is also delayed, resulting in even later action and mitigation for the farmers. This leads to crop losses, lower agricultural productivity, and possible financial difficulties for farmers. Hence, the client said that a web application which detects the rice crop stress can be beneficial to the farmers and agronomists of Lubang Island, Mindoro. Furthermore, there are multiple client requirements for the project. The initial requirement is that the web application must accurately identify both biotic and abiotic stressors affecting a rice crop. The second requirement is that the web app should also provide recommendations to farmers and agronomists regarding how to address and alleviate these stresses in the rice crop. Lastly, the final requirement is that the web app must significantly reduce the time it takes to detect rice crop stress.

2. LITERATURE REVIEW

2.1 Deep Learning for Agricultural Visual Perception Crop Pest and Disease Detection

A significant trend in modern smart agriculture is the increased accuracy of crop detection in real-world applications, as opposed to only image categorization in simplistic or laboratory environments, thanks to advancements in computer vision technology. The ultimate objective of disease detection is to measure the degree of disease occurrence through the assessment of disease severity (damaged area) and disease incidence (diseased leaf/spike/plant ratio). Determining the incidence of sick leaves, ears, or plants requires accurate detection. Determining the exact area of crops destroyed is useful in determining the severity of the illness. Therefore, in order to measure illness occurrence levels, precise disease detection and more condensed disease range indication are necessary. However, in the field scene, wheat diseases occur on wheat leaves with arbitrary leaf orientation, and applying the horizontal bounding boxes to oriented object detection would lead to excessive redundant background regions in the horizontal bounding box. This is not conducive to the accurate calculation of the damaged leaf area in the future to determine the severity of wheat disease. Additionally, the use of horizontal detectors tends to produce missed detection as areas of wheat disease in any direction usually have a large aspect ratio. This is not conducive to accurate calculations of wheat disease incidence in future work. Lastly, the usage of horizontal bounding boxes in high-incidence areas may result in several bounding boxes overlapping, which may make it challenging for farmers or agricultural specialists to determine the precise location of the disease. Consequently, the identification of wheat illnesses cannot be done using standard horizontal detection techniques. This section suggests a convolutional neural network-based approach for identifying wheat illnesses in any direction in order to get over the aforementioned issues (Wang R., et al., 2023).

2.2 Machine Learning and Deep Learning for Smart Agriculture and Applications

With a focus on precision agriculture, digital farming, and emerging concepts, this study illuminates the significance of sustainable food production and resource management in the face of evolving digital hardware and software technologies. Geospatial technology, robotics, the Internet of Things (IoT), and data analytics converge with machine learning and big data to unlock new possibilities in agricultural management. In addition to providing state-of-the-art insights into data-intensive processes within operational agricultural environments, this study analyzes the synergies between various disciplines. Machine learning has several uses, ranging from precision agriculture and crop monitoring to automated irrigation systems and drones for field study. These cutting-edge methods are also beneficial for tracking the health and identification of animals. One of the study's key focuses is the critical role of health monitoring for plants and fruits in achieving sustainable agriculture. Plant diseases pose significant financial challenges in the farming industry worldwide. By leveraging sophisticated image processing and advanced computer vision techniques, automated detection and identification of plant diseases are revolutionized, enabling precise and rapid identification while minimizing human effort and labor costs. For researchers involved in image processing and computer vision for smart agriculture, this study offers invaluable insights. It covers the most important fields of image processing in the agricultural domain, encompassing computer vision applications, machine learning, and deep learning approaches. (Hashmi M. F. et al., 2023).

2.3 Computer Vision and Machine Learning in Agriculture

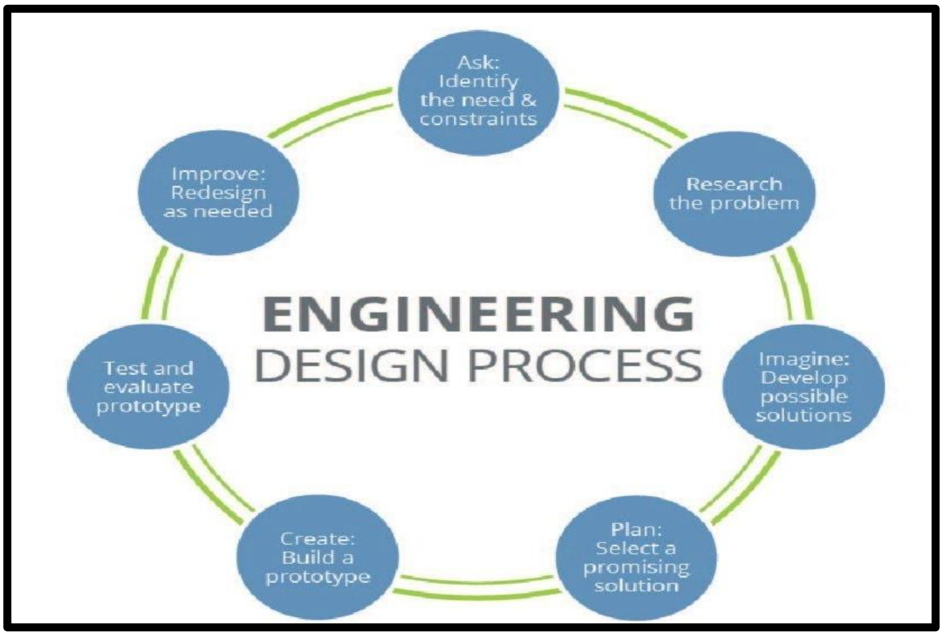
The literature highlighted that the majority of people on the planet get their energy mostly from rice. The

illness in the rice plant must be identified as soon as possible in order to plan for the growing population's need for food. Deep learning has proven to be more effective than standard machine learning, especially in the field of computer vision, when it comes to recognizing intricate structures in high-dimensional data. The proposed work focuses on creating rice leaf disease detection (RLDD) system that identifies many rice leaf diseases such as brown spot, hispa, and leaf blast using rice plant images obtained from the Kaggle API and trained on a deep learning model. Data fusion integrates the information from multiple sources to improve the efficiency. The motivation of the proposed system is to apply the late fusion method to classify the rice leaf diseases. The proposed model yields an accuracy of 98.85% compared to the other models (Rajathi N. et al., 2023).
The synthesis of literature pertaining to stress detection and recognition of Philippine rice varieties underscores a multifaceted exploration at the intersection of agricultural science, technological innovation, and environmental sustainability. The Philippines, as an agrarian nation heavily reliant on rice production, manifests a significant interest in optimizing cultivation practices, mitigating crop stressors, and enhancing varietal resilience. Through an extensive review of relevant studies, several key themes emerge, encapsulating both the challenges faced and the strides made in this domain. Furthermore, a crucial area of research focused on enhancing crop resilience and production stability is the identification and characterization of stress-responsive characteristics within Philippine rice varieties. Various researchers have discovered a variety of morphological, physiological, and molecular markers connected to stress tolerance through thorough phenotypic and genotypic analysis. These markers help in stress-resilient cultivar selection and breeding. They also provide information for targeted genetic engineering efforts that strengthen current cultivars against common stresses such as salinity, drought, and pest infestations. Additionally, the effectiveness and predictive ability of stress detection and recognition frameworks are increased through the incorporation of data-driven techniques like machine learning algorithms and computational modeling. Through the utilization of extensive datasets that include genetic data, agronomic factors, and environmental variables, these computational methods provide refined understanding of the intricate interactions among genotype, phenotype, and environmental stimuli. As such, they facilitate the creation of prediction models that are able to predict susceptibility to stress, direct preventive actions, and maximize the use of resources in rice farming activities.

3. METHODOLOGY

3.1 Engineering Design Process

The engineering design process is a sequence of steps that guides engineers as they identify, plan, and solve a problem. The process is iterative as each step can be repeated as much as necessary, bringing forward new design concepts and solutions.



(Image source www.teachengineering.org)
Figure 1 Engineering Design Process

The engineering design process diagram consists of seven iterative steps: Ask; Research; Imagine; Plan; Create; Test and Evaluate, and; Improve as shown in Figure 1. The iterative nature of this process allows repetition of steps before moving to the next step in order to learn from failures and improve. This empowers problem solvers to achieve a high level of optimization and meet requirements of set objectives. Thus, the process will be followed and adhered to in the development of the project. A detailed explanation of each step of the Engineering Design Process and its relation to the project will be further explained below:

1. Ask: Identify the Need and Constraints

The critical part of the engineering design process is the first step by identifying what is the problem of your client, target users, project resources and requirements, specifications, and the main intention of the project. To establish the difficulties and problems of the client, an interview was conducted with Ms. Ellen Morales, Head DA of Lubang, Occidental Mindoro. The meeting resulted giving out the specific concerns that they are currently facing specifically in late detection of rice stress and pest infestation. Upon further discussion, the proposed idea was to develop a web application that can detect the type of rice stress by uploading or taking an image and providing a recommendation. With the suggested solution, the constraints are training time, size of the model, accuracy, image size, and processing power.

2. Research the Problem

It is vitally important to gauge and evaluate the problem, and measure the scalability of its needs to provide fitting solutions. To research the problem, an interview was conducted to gather information and determine the extent of the current problem. With respect to identifying the problem, determining the current method is also needed, to assess how to further better the practice and layout alternatives. In a study about Abiotic Stress Signaling in Rice Crop, Abiotic stresses affecting rice production include drought, salinity, high and low temperatures, UV radiations, etc. In addition, a study about Biotic Stresses states that biotic stresses represent a serious threat to rice production to meet global food demand and thus pose a major challenge for scientists, who need to understand the intricate defense mechanisms. The Department of

Agriculture in Lubang, Occidental Mindoro, has trouble with detecting the rice stress in their region since they manually check and cross validate it with their atlas of rice stresses. This method also causes them weeks of waiting time before they can detect and take action on what rice stress is determined by their agronomist.

3. Image: Develop Possible Solution

The problem has already been identified and the constraints are set, researching the problem and its current practice has also been determined, the next step is to develop a possible solution. After gathering and researching data, the researchers will now evaluate and brainstorm what possible solutions can be. When formulating the project, the researchers took into account the potential solutions, guidelines, and limitations that could play a vital role and should be carefully evaluated. These considerations encompassed factors such as training time, size of the model, accuracy, image size, and processing power. Trade-offs are also considered in each design to emerge with the best possible solution that is fitting for the problem that the client is experiencing.

4. Plan: Select a Promising Solution

Based on the previous section, three design concepts will be created, drawing upon the research and standards that will be gathered. These established standards will be employed in evaluating all three designs. Each design will undergo research and possess distinctive attributes, which will be used to determine the most optimal design. Furthermore, calculations will be performed to compare these designs, and the one that ranks higher than the others will be designated as the best design.

5. Create: Build a Prototype

To tackle the problem effectively, the researchers create a working prototype. This prototype will be carefully designed to meet the project's goals, client needs, and relevant standards. Picking the right deep learning models is crucial because these models help us detect and categorize stress in rice crops. The researchers make sure the system is safe, dependable, and can provide accurate results by following established standards. The prototype includes making the model and developing the web application for the project. A thorough research to ensure the system works correctly then test and evaluate the prototype's design, setting the stage for the next project steps.

6. Test and Evaluate Prototype

Before a web application is launched for end-users, thorough testing and evaluation of the system are essential to ensure the application functions correctly and to identify potential bugs. Both pre-deployment and post-deployment product testing is a crucial step in this process. By doing so, the application's performance can be assessed to ensure it operates optimally for real-time users. After testing, feedback should be carefully considered as it plays a vital role in enhancing the web-based system. Feedback helps pinpoint pain points from the previous version, enabling developers to make informed decisions and improvements that lead to the successful application for its intended users. Furthermore, any bugs identified can be addressed early in the testing and evaluation stages to ensure a smooth performance of the web application, minimizing user dissatisfaction. Ultimately, developers can optimize the application's performance to deliver a seamless user experience. In this way, testing and evaluating the application determines whether the image classifier correctly identifies rice crop diseases and provides recommendations for farmers and agronomists after classification.

7. Improve: Redesign as Needed

The engineering design process includes multiple iterations of redesigning as needed. In this step of the process and problems encountered during the testing and evaluation of the prototype must be addressed accordingly. The project will be improved iteratively until it meets the project requirements.

3.2 Project Design

The project design is a vital stage of the development of the project wherein it discusses the characteristics, capabilities, and varying designs of the project. The objective of this stage is to develop three designs that can achieve the purpose of the project, each design with the same functionalities identified by the objectives of the project in consideration of constraints and standards. The designs identified must suit the project requirements of the client provided that the whole development process of the system is within one year. The three alternative designs to be discussed are all created with regards to the standards and requirements set by the client. These designs focus on providing efficient and effective classification of rice stress, which will be discussed further in the following sections. The designs are developed to fit the specifications of the project, based on State-of-the-Art classification algorithms that provide results that meet the project objectives. The system will be a web application that accepts images inputted into the system, applies necessary data preprocessing techniques in order to prepare them for classification, and feeds them into the classification algorithm. The classification algorithm will then perform a series of feature extraction techniques, complex calculations based on the extracted features, and finally classifying the image to the resulting rice stress identified. Based on the designs, three state of the art algorithms are proposed: InceptionV4, MobileNetV3, and EfficientNetV2.

Design 1: Deep Learning with Inception-v3

1.1 Design Description

Inception-v3 is a deep learning model based on Convolutional Neural Networks (CNN), which is primarily used for image classification. The inception-v3 is a superior version of the basic model inception-v1 which was introduced as GoogleNet which was developed by a team at Google. Specifically, Inception-v32 was released last 2015. Compared to the previous versions, Inception-v3 has a total of 42 layers and lower error rate. The modifications done to make Inception-v3 better than the other Inception versions are factorization into smaller convolutions, spatial factorization into asymmetric convolutions, utility of auxiliary classifiers, and efficient grid size reduction.

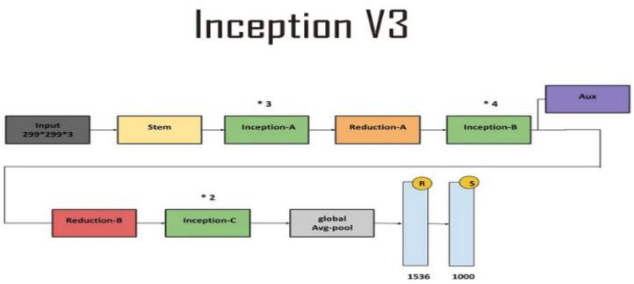


Figure 2 Main Architecture of Inception-v3

1.2 Illustrative Diagram

The illustrative diagram depicts the process of using Inception-v3 for classifying rice crop diseases from scanned images. In the input stage, a scanned image provided by the farmer or agronomist is used in the processing section, where detection is performed using the Inception-v3 CNN architecture. As an output, once the rice crop stress has been detected, the website application will offer recommendations or actionable measures for the end user.

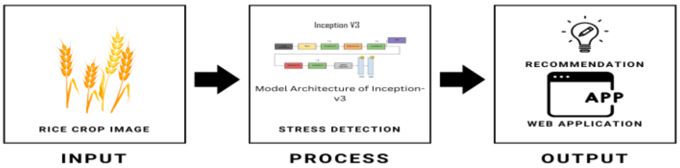


Figure 3 Illustrative Diagram using Inception-v3 CNN Architecture

1.3 Design Constraints

When considering a design with a CNN architecture of Inception-v3, the following constraints should be considered first.

Accuracy

The accuracy of the system's prediction stands as an important design parameter. This measures the system's capability to deliver accurate and reliable outcomes. To adhere to this constraint, the system must give priority to accuracy, ensuring its ability to make precise predictions in the identification of rice crop stress. The design should encompass mechanisms that enhance the accuracy of the system's outputs, maintaining a consistently high level of dependability in its evaluations. The model accuracy of Deep Learning using Inception-v3 has an error rate of 0.063.

Training Time

Training time represents the duration during which a model learns from a specific dataset. It encompasses the total time required for various computational tasks to evaluate the model's performance. When training an image classifier, the training time can be considerably longer compared to working with simple numeric data. Therefore, it's essential to pay close attention to training time in order to devise the most efficient architecture to use. The training time of Deep Learning using Inception-v3 is 36 hours.

Model Size

The size of the machine learning model used in the system is a fundamental design constraint. It assesses the physical and computational footprint of the model, impacting aspects such as storage requirements and deployment efficiency. To adhere to this constraint, the system should prioritize the development of compact and efficient models that minimize storage demands and computational resources. The design should focus on optimizing the model size to ensure practicality and effective deployment, especially in resource-constrained environments. The model size of Deep Learning using Inception-v3 is 92 MB.

Image Size

Selecting the size of the images is an essential decision. It's similar to picking the level of clarity and detail in the pictures. To optimize the system's performance, choosing the image size should be considered. The correct image size helps the system in its task of analyzing and classifying images. The image size of Deep Learning using Inception-v3 is 299x299. Image preprocessing is a critical component of the system and can significantly impact the maximum accuracy achieved by the model during training.

Processing Power

Processing power contains the abilities of both the central processing unit (CPU) and the graphics processing unit (GPU) to execute inference steps. The processing capabilities of both the CPU and GPU significantly influence the speed and efficiency of the inference process throughout the training phase, making them crucial components in the overall performance of the system. The processing power of Deep Learning using Inception-v3 is 42.2 ms.

Design 2: Deep Learning with EfficientNetV2L

2.1 Design Description

EfficientNetV2 is a convolutional network with faster training speed and better parameter efficiency than EfficientNet. It is developed through a combination of training-aware neural architectures search and scaling, optimizing both training speed and model efficiency. EfficientNetV2 achieved a good performance in image classification using ImageNet dataset (Tan & Le, 2021).

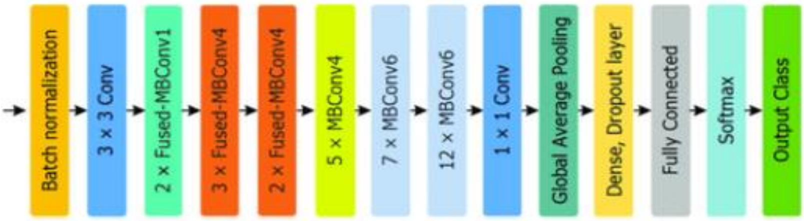


Figure 4 Architecture of EfficientNetV2 (Albattah, Nawaz, Masood, & Javed, 2022)

2.2. Illustrative Diagram

The illustrative diagram illustrates how the system functions. This diagram is composed of labels and arrows, which displays the connections between the components and the methods the project employs, facilitating a deeper understanding of the subject under observation.

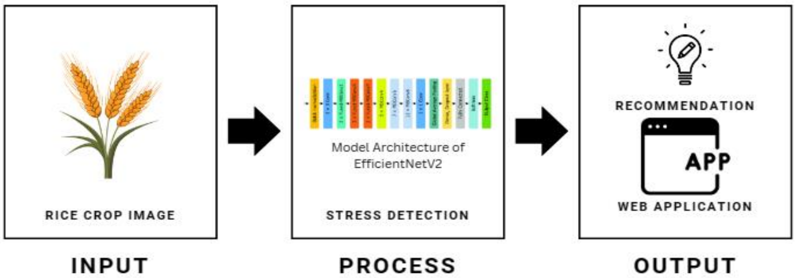


Figure 5 Illustrative Diagram using the EfficientNetV2

Figure 5 shows the illustrative diagram of the web application. In the diagram above, the user will input a rice crop image in the web application. This will be processed through the use of EfficientNetV2 to identify the crop stress. Once the model detects the crop stress, it displays the type of crop stress in the web application and it will provide recommendations based on the detected type of stress.

2.3 Design Constraints

There are multiple constraints bounding the design. The constraints include the following:

Accuracy

The accuracy of the system's prediction stands as an important design parameter. This measures the system's capability to deliver accurate and reliable outcomes. To adhere to this constraint, the system must give priority to accuracy, ensuring its ability to make precise predictions in the identification of rice crop stress. The design should encompass mechanisms that enhance the accuracy of the system's outputs, maintaining a consistently high level of dependability in its evaluations. EfficientNetV2L achieves 97.5%

accuracy on ImageNet dataset, showcasing its precise object classification capabilities (Tan & Le, 2021). The high accuracy value emphasizes the model's dependability and efficiency in detecting features and patterns, making it a valuable choice for tasks demanding precise image analysis and classification.

Training Time

The training time needed for the deep learning model to undergo training is an important design constraint. It quantifies the time required for the model to acquire proficiency through the provided data. To address these constraints, the system should target efficient training times, ensuring that the model is trained within the acceptable time frames. The design should take into account procedures and optimization techniques to reduce the training duration, thus improving the overall efficiency of the deployment and development. The training time for EfficientNetV2L is 24 hours, faster than the models from ImageNet (Tan & Le, 2021). This training period indicates that the EfficientNetV2L has a fast-learning process while performing a good performance. This means that the model is a significant achievement that has the potential to democratize access to powerful image classification models.

Model Size

The size of the deep learning model used in the system is a fundamental design constraint. It assesses the physical and computational footprint of the model, impacting aspects such as storage requirements and deployment efficiency. To adhere to this constraint, the system should prioritize the development of compact and efficient models that minimize storage demands and computational resources. The design should focus on optimizing the model size to ensure practicality and effective deployment, especially in resource-constrained environments. The model size of the EfficientNetV2L is 479 MB using the ImageNet dataset (Tan & Le, 2021). This answers the fast-training time of the model as it has a relatively smaller model size. This indicates that the model is less complex and requires less computation to train and deploy.

Image Size

The dimensions and size of the input images represent a critical design constraint for the system. They define the resolution and visual detail available for analysis by the machine learning model. To address this constraint, the system should establish and adhere to specific image size parameters that optimize the model's performance and efficiency. The design should strike a balance between image size and computational demands, ensuring that the selected image dimensions align with the available processing power and storage capacity while meeting the system's objectives for image analysis and classification The image size of 224 x 224 pixels is set to train the EfficientNetV2L using the ImageNet dataset (Tan & Le, 2021). The image size contributes to the performance of the model. With higher image size, the deep learning model can effectively handle and examine images, finding the right balance between computational requirements and image quality. This standardized image size is selected to guarantee alignment with the model's architecture, enhancing its effectiveness in tasks like image classification and analysis.

Processing Power

The processing power available for the system is a significant design constraint. It assesses the computational resources and capabilities required for efficient model training, inference, and real-time execution. To meet this constraint, the system should prioritize the development of models and algorithms that are optimized for the available processing power. The design should aim to strike a balance between the computational demands of the system and the processing power at hand, ensuring that the system can operate effectively without overwhelming the available resources. The processing power per interference step shown above for EfficientNetV2L is 1010 ms (Hu et.al, 2022). This metric represents the time required for the model to perform a single inference step on a central processing unit (CPU). The relatively low inference time of the model indicates that the model is suitable for tasks involving classification and image analysis.

Design 3: Deep learning with MobileNetV3

3.1 Design Description

MobileNetV3 is a lightweight convolutional neural network (CNN) architecture that is designed for mobile and embedded devices. It is the successor to MobileNetV1 and MobileNetV2, and it achieves state-of-the-art accuracy on image classification and object detection tasks while using significantly fewer parameters and computational resources. (Howard et al., 2019).

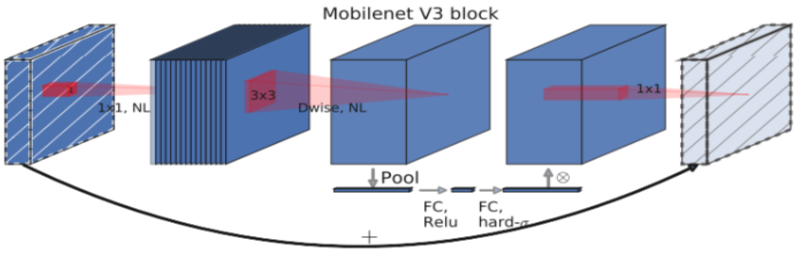


Figure 6 Model Architecture of MobileNetV3 (Image Source: www.paperswithcode.com)

The figure above shows the model architecture of the third design, MobileNetV3. This architecture comprises a series of inverted residual blocks, each consisting of a depth wise separable convolution, a pointwise convolution, and a squeeze-and-excitation module. The depth wise separable convolution independently filters each input channel, and the pointwise convolution combines their outputs efficiently, reducing parameters and computations. The squeeze-and-excitation module weights feature maps based on importance, enhancing the focus on critical features. These inverted residual blocks are stacked, progressively learning more complex features with increasing channel sizes. The network concludes with global average pooling and a fully connected layer, achieving remarkable efficiency and accuracy, particularly suited for mobile and embedded devices.

3.2 Illustrative Diagram

An illustrative diagram, featuring labels and arrows, illustrates how the system functions. This diagram enables an examination of the connections between system components and the methods it employs, facilitating a deeper understanding of the subject under observation.

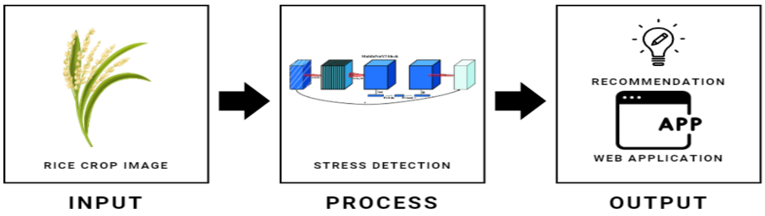


Figure 7 Model Architecture of MobileNetV3

Figure 7, illustrates the diagram for the third design. In this setup, the user-provided rice crop image serves as the input data, which is then processed by the MobileNetV3 deep learning algorithm to identify rice crop stress. The web application responds by providing recommendations based on the detected stress type.

3.3 Design constraints

There are multiple constraints bounding the design. The constraints include the following:

Accuracy

The accuracy of the system's predictions is an essential design constraint. It measures the system's ability to provide correct and reliable results. To meet this constraint, the system should prioritize accuracy, ensuring that it makes precise predictions when identifying rice crop stresses. The design should incorporate mechanisms that enhance the accuracy of the system's outputs, maintaining a high level of reliability in its assessments. MobileNetV3 demonstrates a remarkable accuracy rate of 91.34% on the MobileNet dataset, showcasing its proficiency in precisely classifying objects within images (Howard et al., 2019). This high accuracy level underscores the model's reliability and effectiveness in discerning features and patterns, making it a valuable choice for tasks requiring precise image analysis and classification.

Training Time

The training time for the system's machine learning model is a crucial design constraint. It measures the duration required for the model to learn from the provided data and reach a level of proficiency. To address this constraint, the system should aim for efficient training times, ensuring that the model is trained within acceptable timeframes. The design should consider streamlined processes and optimization techniques to minimize the training duration, thereby enhancing the overall efficiency of the system's development and deployment. The training time for MobileNetV3, using the MobileNet dataset, extended over a substantial duration, requiring approximately 48 hours to complete (Xiang Shang, 2023). This significant training period indicates the model's intricate learning process, where it analyzes vast amounts of data to acquire the proficiency needed for accurate image classification. This lengthy training time serves as a practical consideration for resource planning during model development, as it demands a substantial computational commitment to achieve its high accuracy.

Model Size

The size of the machine learning model used in the system is a fundamental design constraint. It assesses the physical and computational footprint of the model, impacting aspects such as storage requirements and deployment efficiency. To adhere to this constraint, the system should prioritize the development of compact and efficient models that minimize storage demands and computational resources. The design should focus on optimizing the model size to ensure practicality and effective deployment, especially in resource-constrained environments. The model size of MobileNetV3 is 4.88 MB as reported by Kang et al. This concise model size is a significant advantage, as it aligns with constraints related to storage and computational efficiency, making MobileNetV3 a practical choice for various applications with limited resource capacities.

Image Size

The dimensions and size of the input images represent a critical design constraint for the system. They define the resolution and visual detail available for analysis by the machine learning model. To address this constraint, the system should establish and adhere to specific image size parameters that optimize the model's performance and efficiency. The design should strike a balance between image size and computational demands, ensuring that the selected image dimensions align with the available processing power and storage capacity while meeting the system's objectives for image analysis and classification. The image size utilized for MobileNetV3 is standardized at 224x224 pixels (Howard et al., 2019). This specific image dimension serves as a crucial parameter in the system's design. It allows the machine learning model to efficiently process and analyze images, striking a balance between computational demands and image quality. This standardized image size is chosen to ensure compatibility with the model's architecture, optimizing its performance in tasks such as image classification and analysis.

Processing Power

The processing power available for the system is a significant design constraint. It assesses the computational resources and capabilities required for efficient model training, inference, and real-time execution. To meet this constraint, the system should prioritize the development of models and algorithms that are optimized for the available processing power. The design should aim to strike a balance between the computational demands of the system and the processing power at hand, ensuring that the system can operate effectively without overwhelming the available resources. The processing power, measured in milliseconds per inference step (CPU), for MobileNetV3 is 51.2 milliseconds (ms) (Keras, n.d.). This metric represents the time required for the model to perform a single inference step on a central processing unit (CPU). The relatively low inference time of 51.2 ms underscores the model's efficiency, making it suitable for real-time applications and tasks where swift image analysis and classification are essential.

4. RESULTS

In the following table, we provide a summarized overview of the constraints for each of the three project designs, along with their respective values.

Table 1: Summary of Constraints

Design	Constraints				
	Functionality (Error Rate)	**Manufacturability (Training Time)**	**Environmental (Model Size)**	**Performance (Image Size)**	**Economic (Resource Requirement)**
Design 1: Deep Learning with InceptionV3	0.063	36 Hours	92 MB	299x299 Pixel (89,401)	42.2 ms
Design 2: Deep Learning with EfficientNetV2 L	97.5%	24 Hours	479 MB	224x224 Pixel (50176)	1010 ms
Design 3: Deep Learning with MobileNetV3	91.34%	72 Hours	4.88 MB	224x224 Pixel (50176)	51.2 ms

Based on the table above, design 2 has the highest accuracy compared to all the designs, with MobileNetv3 having the lowest. It also has the highest training time of 48 hours, with Design 2 having the fastest training time with a total of 24 hours. In terms of Model size, design 3 has the lowest with a size of 4.88 MB while design 2 has the highest with 479 mb. In terms of image size, both designs 2 and 3 have the same values, while design 1 has a 299x299 size. And finally, the processing power of design 1 performed the best,

followed by designs 3 and 2, respectively. The three designs for constructing a rice crop disease detection were evaluated using the Pareto Multi-Criteria Decision Making (MCDM) technique, which uses trade-off analysis to identify the designs that satisfy the given criteria in a Pareto-optimal way. Each design is labeled by its level of importance, performed repeatedly to other criteria as well.
Table 2 below shows the goal of each constraint in minimization and maximization as well as the level of importance of each design constraint. When it comes to the model's accuracy constraint, the higher the accuracy the model performs, the better the design can be regarded. The training time constraint tells how long the model will be trained, meaning the lower the training time is, the better it is for the specific design. Model size pertains to the capacity of the model, which means that the smaller the model will be trained, the better for the design. The constraint image size has the lowest level of importance among all the levels; it pertains to the dimensions of the image that will be used to feed the model. In that way, the smallest image size will be preferred in the design. Lastly, processing power constraints pertain to the capabilities of the CPU and GPU to put the model into production, or, in simple terms, to train the image classification model. The lowest processing power requirement will be regarded as the best use in this design. The constraint that is regarded as the most important is accuracy, with a level of importance of 10, followed by processing power with a level of importance of 9, model size with a level of importance of 8, training time with a level of importance of 7, and lastly, image size with a level of importance of 6.

Table 2: Goal and Level of Importance of the Constraints

CRITERIA	LEVEL OF IMPORTANCE	WEIGHT	GOAL
Functionality (Error Rate)	10	25.00%	MIN
Manufacturability (Training Time)	7	17.50%	MIN
Environmental (Model Size)	8	20.00%	MIN
Performance (Image Size)	6	15.00%	MIN
Economic (Resource Requirement)	9	22.50%	MIN

Table 3 shows the summarized normalized values of the constraints of each design, InceptionV3, EfficientNetV2L, and MobileNetV3. The highest normalized value is 10 and the lowest normalized value is 1. Getting the sum of the normalized value for each design, EfficientNetV2L displays the highest score, with MobileNetV3 placing second to the highest, and InceptionV3 showing the lowest normalized value.

Table 3: Normalized Values of Three Design Alternatives

CRITERIA	LEVEL OF IMPORTANCE	WEIGHT	DESIGN			GOAL
			InceptionV3	EfficientNetV2L	MobileNetV3	
Functionality (Error Rate)	10	25.00%	4.448051948	10	1	MIN
Manufacturability (Training Time)	9	22.50%	7.75	10	1	MIN
Environmental (Model Size)	7	17.50%	8.346241458	1	10	MIN
Performance (Image Size)	6	15.00%	1	10	5.834832984	MIN
Economic (Resource Requirement)	8	20.00%	10	1	9.916305022	MIN
TOTAL	40	100.00%	31.54429341	32	27.75113801	

5. DISCUSSIONS

The project integrates several constraints to ensure its effective execution and optimal performance. These constraints are vital as they delineate the project's boundaries and limitations, guiding its development. They encompass critical factors such as accuracy, training time, model size, image size, and processing power, each playing a pivotal role in shaping the project's design and overall outcomes. These constraints are aligned with engineering requirements and standards, serving as cornerstones for the design process. A comprehensive evaluation of various design alternatives is essential to gauge their alignment with these constraints and ascertain whether the project successfully fulfills its intended objectives.

6. CONCLUSIONS AND RECOMMENDATIONS

The successful development of the web-based stress classification system for rice crops represents a pivotal achievement in addressing the challenges faced by the Department of Agriculture Region IV-B (MIMAROPA). The collaboration with the Department of Agriculture, particularly with Ms. Ellen Morales, Head DA of Lubang, Occidental Mindoro, provided valuable insights into the manual and time-consuming processes hindering timely detection of stress conditions in rice crops. The system, designed with an accurate classification algorithm utilizing artificial intelligence, not only met but exceeded the outlined objectives, achieving a commendable accuracy rate of at least 95%. Near real-time stress detection, efficient user interface, and a recommendation system collectively contribute to mitigating crop losses and enhancing agricultural productivity. However, to further enhance the user experience and system effectiveness, it is recommended to incorporate additional features into the website. Users would benefit from expanded edit functions in their profiles, allowing them to customize and manage their accounts more comprehensively. Additionally, the inclusion of tutorial videos on how to classify rice crop stress on the web application would provide users, especially farmers and agronomists, with practical guidance and foster a more intuitive interaction with the system. Furthermore, recognizing the importance of a stress classification system, there is room for improvement in the classifier by expanding its focus beyond biotic stresses. Currently, the system predominantly identifies biotic stressors; however, enhancing the classifier to include abiotic stresses for rice crops would provide a more comprehensive solution. By incorporating features that detect abiotic stress factors, such as weather-related conditions, the system can offer a more nuanced analysis, equipping farmers and agronomists with a broader understanding of the challenges their crops may face. This expansion aligns with the continuous evolution of agricultural technology and ensures a future-ready solution for the dynamic landscape of rice farming in the Philippines.

REFERENCES

1. Rujing Wang, Lin Jiao, Kang Liu (2023) Deep Learning for Agricultural Visual Perception Crop Pest and Disease Detection
2. Mohamamd Farukh Hashmi, Avinash G. Kesakr (2023) Machine Learning and Deep Learning for Smart Agriculture and Applications
3. N. Rajathi, K. Yogajeeva, V. Vanitha, P. Parameswari (2023) Computer Vision and Machine Learning in Agriculture, Volume 3
4. Galieni, A., D'Ascenzo, N., Stagnari, F., Pagnani, G., Xie, Q., & Pisante, M. (2021). Past and future of plant stress detection: An overview from remote sensing to positron emission tomography. Frontiers in Plant Science, 11. https://doi.org/10.3389/fpls.2020.609155
5. Kollist, H., Zandalinas, S. I., Sengupta, S., Nuhkat, M., Kangasjärvi, J., & Mittler, R. (2019). Rapid responses to abiotic stress: Priming the landscape for the signal transduction network. Trends in Plant Science, 24(1), 25–37. https://doi.org/10.1016/j.tplants.2018.10.003
6. Miraglia, M., Marvin, H. J. P., Kleter, G. A., Battilani, P., Brera, C., Coni, E., Cubadda, F., Croci, L., De Santis, B., Dekkers, S., Filippi, L., Hutjes, R. W. A., Noordam, M. Y., Pisante, M., Piva, G., Prandini, A., Toti, L., van den Born, G. J., & Vespermann, A. (2009). Climate change and food safety: An emerging issue with special focus on Europe. Food and Chemical Toxicology, 47(5), 1009–1021. https://doi.org/10.1016/j.fct.2009.02.005
7. Prosekov, A. Y., & Ivanova, S. A. (2018). Food security: The challenge of the present. Geoforum, 91, 73-77. https://doi.org/10.1016/j.geoforum.2018.02.030 Expo Documentation. (n.d.). https://docs.expo.dev/versions/latest/#support-for-android-and-ios-versions