Rice Leaf Disease Classification Using Knowledge Distillation

Md. Saiful Niaz[1], Tasmia Hassan Shabonty[1], Ashfaqur Rahman Saad[1], Lucky Talukder Tarin[1], Dewan Md. Farid[1], and Huu-Hoa Nguyen[2]

[1] Department of Computer Science and Engineering, United International University United City, Madani Avenue, Badda, Dhaka 1212, Bangladesh

[2] College of Information and Communication Technology, Can Tho University, 92000, Can Tho, Vietnam

Correspondence: Huu-Hoa Nguyen, email: nhhoa@ctu.edu.vn

Received: 31/07/2024, revised: 23/09/2024, accepted: 06/11/2024

Digital Object Identifier:: 10.32913/mic-ict-research-vn.v2024.n3.1310

Abstract: Rice is essential for global food security, particularly in Asian countries where it is a staple food and a significant contributor to the economy. Rice cultivation faces numerous challenges, including a variety of illnesses that can significantly impact the yield and quality of the harvest. Conventional disease detection methods are slow, require a lot of manual effort, and are prone to mistakes, emphasizing the critical requirement for automated approaches. This research explores the implementation of knowledge distillation, a method of transferring expertise from advanced Models to more basic ones, for identifying diseases in rice. We condensed the expertise of complex Models by training them extensively, creating a streamlined and effective Model. We employed DenseNet121 as the Teacher Model, and the Student Model is equipped with dual pipeline Models to address overfitting. Both of these Models passed through Quantization-Aware Training (QAT). The Student-Teacher collaborative Model achieved a precision of 98.47%, using significantly fewer parameters (5.9 million) than the Teacher-only Model. Finally, the whole Model was quantized to an 8-bit integer from 32-bit float points. This leads to a fairer method for effectively and accurately isolating small and complex characteristics. This small-scale is ideal for use on portable gadgets in environments with restricted resources, enabling farmers to quickly identify diseases for better crop management and increased worldwide food security measures.

Keywords: *Deep learning, knowledge distillation, quantization, rice leaf disease classification.*

Tiêu đề: **Phân loại bệnh lá lúa bằng phương pháp chưng cất kiến thức**

Tóm tắt: Lúa đóng vai trò thiết yếu đối với an ninh lương thực toàn cầu, đặc biệt tại các quốc gia châu Á, nơi lúa là thực phẩm chính và nguồn kinh tế quan trọng. Việc trồng lúa đối mặt với nhiều thách thức, bao gồm các loại bệnh có thể gây ảnh hưởng nghiêm trọng đến năng suất và chất lượng mùa màng. Các phương pháp phát hiện bệnh truyền thống thường chậm, tốn nhiều công sức và dễ xảy ra sai sót, do đó cần thiết phải có các giải pháp tự động hóa. Nghiên cứu này tìm hiểu việc áp dụng phương pháp chưng cất kiến thức – phương pháp chuyển giao kiến thức từ các mô hình phức tạp sang các mô hình đơn giản hơn – nhằm nhận diện bệnh trong cây lúa. Chúng tôi đã tối giản hóa kiến thức từ các mô hình phức tạp bằng cách huấn luyện chuyên sâu, tạo ra một mô hình hiệu quả và gọn nhẹ. DenseNet121 được sử dụng làm mô hình giáo viên, và mô hình học sinh được trang bị hệ thống đường dẫn kép để giảm thiểu tình trạng quá khớp. Cả hai mô hình đều trải qua quá trình huấn luyện nhận thức lượng tử (QAT). Mô hình hợp tác giữa giáo viên và học sinh đạt độ chính xác 98,47%, với số lượng tham số ít hơn đáng kể (5,9 triệu) so với mô hình chỉ dùng giáo viên. Cuối cùng, toàn bộ mô hình được lượng tử hóa từ số thực 32-bit sang số nguyên 8-bit, tạo ra một phương pháp hiệu quả hơn để tách biệt các đặc tính nhỏ và phức tạp một cách chính xác. Mô hình nhỏ gọn này lý tưởng cho các thiết bị di động trong môi trường hạn chế tài nguyên, giúp nông dân nhận diện bệnh nhanh chóng, hỗ trợ quản lý cây trồng tốt hơn và tăng cường an ninh lương thực toàn cầu.

Từ khóa: *Deep learning, knowledge distillation, quantization, rice leaf disease classification.*

I. INTRODUCTION

Rice is an important type of food that feeds billions of people all over the world. It is a significant aspect of Asian culture, history, and economy as well as a main food source [1]. In Bangladesh, rice is essential for everyone's daily life and for how the country makes money. Almost 90% of the population here depends on rice for their calories [2]. In the agriculture of Bangladesh, rice production contributes 40% to the country's GDP [2]. However, growing rice is not always easy. When plants get sick, they can't grow well and don't give as much rice. There are different diseases like bacterial leaf blight, blasts, sheath blight, and brown spots that can make this happen. If farmers don't find good ways to deal with these diseases, they can lose a lot of crops [3]. These diseases can also make the rice grains not as good and safe to eat. Rice disease causes global grain loss of 10% to 30% annually [4]. In traditional agriculture, identifying rice diseases typically depends on the experience gained by farmers over time. This method requires a lot of skilled people as well as significant time, effort, and money. However, it often leads to biased judgments and significant errors. As a result, it becomes difficult to correctly detect diseases, which makes it simple to miss the best window of opportunity for disease prevention and control. Moreover, keeping up with the need for real-time monitoring and predicting various diseases across a wide spectrum is quite difficult [5]. Hence, investigating automated techniques to identify common rice illnesses is urgently needed [6]. Thus, although rice is Bangladesh's main food source, it is equally critical to monitor the spread of these diseases and discover measures to contain them. It is crucial to identify rice diseases quickly and accurately in order to ensure rice production and protect global food security [7].

Deep learning technologies have taken revolutionary steps in many fields, and as a result, their potential for detecting agricultural diseases has been gradually investigated. When it comes to rice plant disease detection, deep learning and machine learning have significantly improved. These technologies work by using really smart computers that can look at lots and lots of pictures or data from fields to spot signs of diseases before they become a big problem [8]. Farmers can therefore take prompt action to prevent the illnesses from spreading and damaging their crops. To solve this problem machine learning and convolutional neural networks have implemented different new techniques and obtained very good results as well [9]. Knowledge distillation involves transferring knowledge from a complex model (teacher) to a simpler model (student). In paddy disease detection, this means training a large, resource-intensive model to identify diseases accurately, and then distilling its knowledge into a smaller, more efficient model suitable for deployment in resource-constrained environments like rural areas. This technique minimises computing resources as well as maintains performance. By placing these lightweight models on low-power devices, like cellphones or edge devices, farmers can acquire reliable disease detection tools even in the absence of internet connectivity, enabling timely interventions to safeguard their crops and livelihood.

Distilling knowledge from a big, complex model to a smaller one works well, even when data is limited. Distillation can condense the benefits of ensemble learning into a single, deployable model, like for Android voice search. It's possible to improve the performance of large networks by training specialist nets [10]. By using the knowledge distillation technique, Musa et al. [11] proposed a new method for detecting plant diseases in smart hydroponic systems. This system uses a special type of deep learning model designed to be less powerful and use less energy. This is achieved by using a simpler network structure and a technique called knowledge distillation. The authors found that their model achieved a high accuracy of 99.4% in detecting diseases while significantly reducing power consumption compared to existing models. Hu et al. [12] used knowledge distillation techniques to detect maize leaf diseases using a lightweight image recognition model. This model is based on an improved version of YOLOv5s which is a popular detection system.

In the proposed methodology, we have leveraged knowledge distillation from larger models to function as mentors in instructing pared-down models with reduced parameters. This methodology necessitated the training of numerous high-capacity models such as Google's Inception, MobileNet, and Xception, as well as the VGG-19, DenseNet-121, and NasNetmobile architectures. We obtained the DenseNet-121 model, which demonstrated the highest level of accuracy, and opted to utilise it for training our student model. We amalgamated two distinct models based on DenseNet-121 and MobileNet-V2 architectures, incorporating custom layers for training on the teacher's dataset. Without the teacher-student model, accuracy was 93.87%. The collaborative efforts of the teacher and student ensemble yielded an accuracy rate of 98.47% utilising parameters 5,91,108, indicating the model's lightweight characteristics.

The rest of the paper is organised in the following manner. Section 2 discusses the related works. The basic techniques of pre-trained models are introduced in Section 3. The proposed methodology is presented in Section 4. Section 5 provides experimental results. Finally, the paper concludes with Section 6.

II. RELATED WORK

Paddy Doctor [13] proposed a study using Convolutional Neural Networks (CNN) and transfer learning: VGG16, MobileNet, Xception, and ResNet34. The used dataset consists of 16,225 images of paddy leaves divided into 12 classes for diseases and one for healthy leaves. According to the experimental results, ResNet34 had the highest F1-score of 97.50%. Shweta Lamba [14] proposed a GCL model for disease classification. GCL uses CNN for feature extraction, LSTM for classification, and GAN for data augmentation. The model achieved 97% accuracy on a dataset from GitHub, UCI, Mendeley, and Kaggle. MS-DNet [15] provides a lightweight network design for effective plant disease detection. The model scored 98.32% accuracy, outperforming any current model. It combined publicly accessible data (the PlantVillage dataset) with locally gathered images of paddy and corn plant diseases. Shankaranarayanan Nalini [16] proposes a DNN classification model for paddy leaf disease detection. The model achieved 96.96% accuracy, 95.92% precision, and 96.41% recall during testing. This suggests that DNN CSA has potential for real-time plant disease detection on devices. This literature review strengthens the foundation for this research on paddy leaf disease detection using transfer learning. Studies like Paddy Doctor [13] demonstrate the effectiveness of transfer learning with CNNs, achieving high accuracy (97.50% F1-score). This validates this research approach and provides a benchmark for success. Nalini [16] demonstrated the high efficiency of Convolutional Neural Networks in detecting rice diseases. Additionally, Lamba [14] highlights the value of pre-trained models, supporting this research utilisation of YOLOv8's pre-trained CNN capabilities. Overall, these findings firmly support the chosen research method.

Md. Saiful Niaz [17] proposed a model based on YOLOv8 and CNN models such as Inception-Resnet-V2, ResNet-101, and Xception. The dataset consists of 1224 images to train the model. Rectified Linear Unit (ReLU) activation function and Max pooling were used to represent the complex association with the dataset and to reduce the feature maps' spatial dimensions. The model achieved 99.16% accuracy. The paper of Chinna Gopi Simhadri [18] focused on the application of transfer learning and deep learning techniques to automatically detect rice leaf diseases with 15 pre-trained CNN models. The InceptionV3 model outperformed other models with an average accuracy of 99.64%, where a dataset of 10,080 images of ten rice leaf diseases is used. R. P. Narmadha [19] suggested DenseNet169-MLP for rice plant disease detection with an accuracy of 97.68% that utilises DenseNet169 as a feature extractor. Vinay Gautam's [20] proposed model using InceptionV3, VGG16, ResNet, SqueezeNet, and VGG19 achieved an accuracy rate of 96.4% along with a precision of 66.3% for identifying diseases with 1500 leaf images taken from Kaggle and Mendeley. VGG-19, Inception-ResNet-V2, ResNet-101, and Xception have been used for paddy leaf disease detection by Md Ashiqul Islam [21] and Inception-ResNet-V2 has achieved the highest accuracy of 92.68% among them. The suggested solution offers an efficient method to develop a high-quality object detector. We have used transfer learning instead of starting from scratch to train YOLOv8, which is a powerful object identification model. Pre-trained CNNs are already highly excellent at differentiating basic forms and textures in photos. Thus, training time is significantly decreased by applying the information to the original layers of YOLOv8 from these pre-trained CNNs.

Ali Ghofrani [22] provides an approach for increasing the accuracy of a small model by utilizing information from a larger model. The PlantVillage dataset consists of 54,306 photos of healthy and sick plant leaves. The study achieves 97.58% accuracy on MobileNet, which is close to a large Xception model on the server's 99.73% accuracy. The small model's classification rate is increased by 2.12% employing the teacher-student approach. The proposed method of Yanxin Hu [23] uses novel additions to the YOLOv5s model and uses channel-wise knowledge distillation in model training for boosting the detection of maize leaf diseases with a dataset of 12,957 images containing five leaf diseases. The suggested approach contains 15.5% fewer parameters than YOLOv5s but achieves higher mAP(0.5) and mAP(0.5:0.95) scores of 3.8% and 1.5%, respectively. A low-power deep learning model to identify plant disease in smart hydroponics is proposed by Aminu Musa [24] with knowledge distillation techniques which contains a simple network structure with reduced parameters that results in lower computational requirements and power consumption. An accuracy of 99.4% is achieved by this model with a maximum power consumption of 6.22W, which is significantly lower compared to existing models. In terms of accuracy and power consumption, the proposed low-power student model outperforms other models.

Below Table I gives a gist of the related works that used KD in their core work compared to our work.

Table I
COMPARISON OF TECHNIQUES USED IN RELATED WORKS (KD).

Ref.	Efficiency	Relevance	Overfitting	Innovation
Musa et al. [11]	Low powered	-	-	-
Hu et al. [12]	Less parameters	-	Not mentioned	Standard KD
Ali Ghofrani [22]	Accuracy +2.12%	-	-	-
Our work	Achieved 97.65% accuracy, <50% parameters, KD + QAT + PQT.	Limited resources	Early Stopping, Data Augmentation, batch dataset training	Combo of KD, QAT, PQT

III. PRE-TRAINED MODELS

Pre-trained models are machine learning models that have been trained on large datasets for specific tasks, such as recognising objects in images or understanding text. They come ready-made with learned features, which can save time and resources when building new models. By leveraging these pre-trained features, developers can create more accurate and efficient models for their own tasks without starting from scratch. Pre-trained models have become popular due to their ability to provide state-of-the-art performance and save development time. The pre-trained models we used in this paper are stated in Table II.

Table II
PRE-TRAINED MODELS USED TO TEST DATASET PERFORMANCE. T-PARAMETERS = TRAINABLE PARAMETERS

Models	T-Parameters	Dimension	Batch	Acc
Densenet-121	10,54,725	-	-	96.93%
Mobilenet-V2	13,16,869	-	-	94.38%
Inception-V3	21,01,251	-	-	91.33%
Xception	229,09,228	224x224	32	89.67%
VGG-19	1236,47,861	-	-	87.63%
NasNetMobile	15,85,897	-	-	87.24%
Googlenet	27,48,981	-	-	72.44%

These pre-trained models are used to find the best teacher among them. They have been tested with 3 datasets [25] [26] [27] and used "ImageNet" weights for training the classifiers. We used an early stop with "Patience 5" to tackle overfitting of the models. Therefore, these results show the best values from the training, these values may differ without early stopping. From the scenario, it is visible that DenseNet-121 performed the best with faster data accuracy. Therefore, we used "DenseNet-121" as our teacher model.

IV. METHODOLOGY

Using Knowledge Distillation (KD) alone in the Teacher-Student Model was our first approach. However, we discovered that we had to use Quantization Aware Training (QAT) and Post Training Quantization (PTQ) because we had chosen to use both pre-trained Models in the Teacher and Student roles. This choice was chosen since KD by itself cannot shrink the pre-trained Model's size because its parameters are fixed. To guarantee that the Student Model might be successfully quantized in the future, we applied QAT to both the Teacher and Student Models. By including quantized weights, bias, activation, and convolutional layers in the Model, QAT emulates the effects of quantization. A key component of the network that adds flexibility to the learning capabilities of CNNs is the activation function that allows for non-linear processing of data from the input. Whereas, Bias is an additional parameter in neural networks, including CNNs, that allows the model to fit the data better by shifting the activation function. Bias helps the model learn more flexible patterns in the data. The way that QAT replicates the effects of quantization is seen in the diagram below Figure 1. M1 and M2 denotes two Student Models.

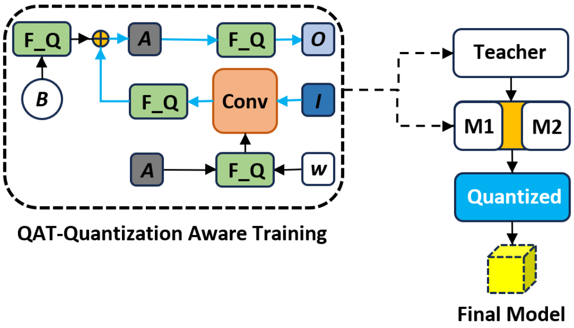


Figure 1. Quantization Applied to both Teacher and Student Model. F_Q = Fake Quantization, W = Weights, A = Activation Funtion, B = Bias, O = Output, I = Input

In order to develop a composite Student Model, our study combines the QAT-DenseNet-121 and QAT-MobileNet-V2 Models. This approach leverages the unique strengths of both architectures: DenseNet-121 excels at complex feature extraction, while MobileNet-V2 excels at high-level feature abstraction. Combining their separate outputs gives the Model a wider range of features to choose from, which improves overall generalization and reduces over-fitting.

These Models are comprised of Convolution layers, Translation layers, Dense blocks, and Linear Bottlenecks. Convolutional layers are composed of a 7x7 conv2D operation with a stride of 2, Batch Normalization for mitigating internal co-variate shift, ReLu activation function, and Max Pooling of size 3x3 with a stride of 2. The Dense Blocks are comprised of numerous convolutional layers with interconnecting connections between each adjacent layer. Translation layers increase the ability to manipulate feature mapping through the process of down sampling. The utilization of Inverted Residual Blocks in this context involves the integration of depth-wise convolution and point-wise convolution, which are then linked with a Linear Bottleneck structure. This bottleneck increases the spatial resolution of the feature maps before downsizing it, allowing for the extraction of intricate patterns. Contrastingly, Residual Connection serves to enhance the propagation of gradients during the training process. Finally, we extracted the results of these two algorithms through the process of Global Average Pooling. The ultimate results of these individual minimalist structures are consolidated into a 256 Dense layer tailored to the specific task at hand, and are aggregated for a four-class categorization with subsequent output forecasts all are drawn in Figure 2.

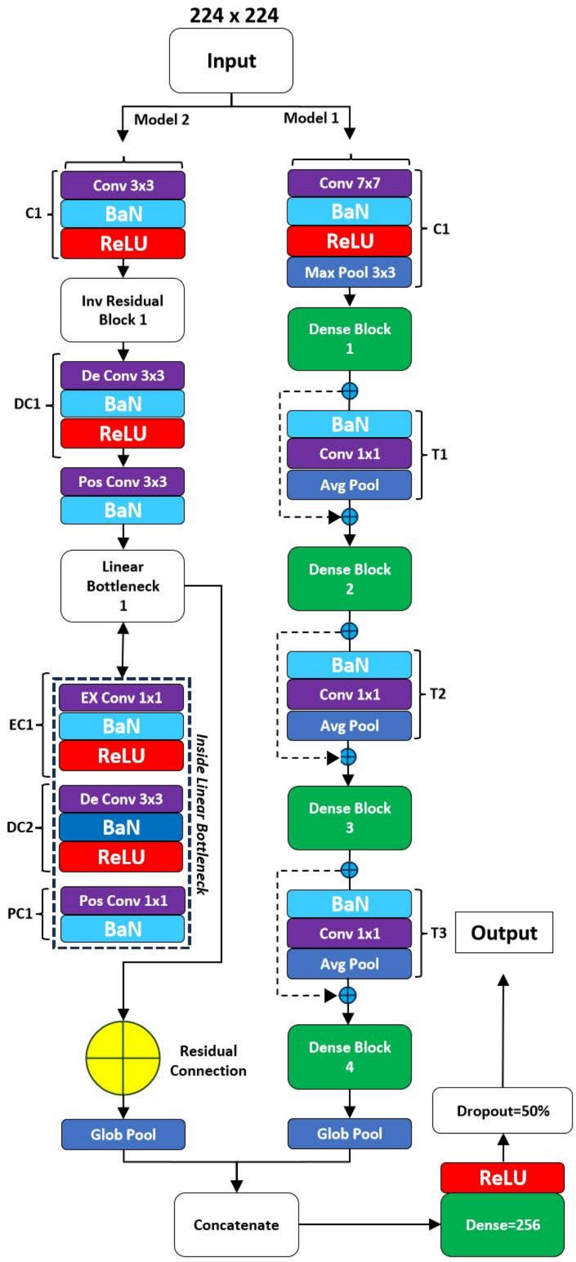


Figure 2. The proposed model architecture.

The Student neural network is trained in conjunction with the expertly trained Teacher neural network (KD). In this scenario, forward propagation is executed on both the mentor and learner networks, followed by back-propagation solely on the learner network. Two loss functions Figure 3 have been specified. One factor to consider is the loss in Student and the loss in distillation function. In addressing Student Loss, the Categorical cross-entropy function is employed, while the Kullback-Leibler Divergence (KL-Divergence) loss function is utilized for Distillation Loss. The Kullback-Leibler Divergence loss function is utilized for determining the dissimilarity between two probability distributions. The objective is to reduce the Kullback-Leibler Divergence in order to align the predictions of the Student network with those of the Teacher network.

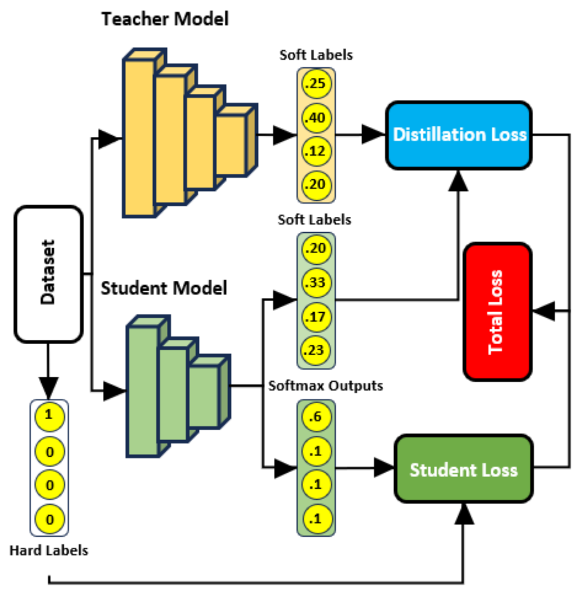


Figure 3. Knowledge Distillation Architecture.

Hard labels in machine learning use binary or one-hot encoding to assign a single, definitive class to every data point. They are a typical representation of assurance in supervised learning tasks. Conversely, Soft labels show a probability distribution among classes, signifying ambiguity and offering more details regarding the Model's confidence in certain results. Soft labels are especially helpful in processes such as KD Figure 3, where they facilitate the learning of a smaller Model (Student) from the richer outputs of a bigger Model (Teacher), hence enhancing performance and generalization.

The Softmax with temperature function adjusts the confidence of the Softmax output (Soft Labels) by introducing a temperature parameter T. This controls the sharpness of the probability distribution. When $T = 1$, the function behaves like the standard Softmax. For $T > 1$, the output probabilities become more spread out and less confident, while for $T < 1$, the probabilities become sharper, focusing more on the largest logit values. The equation is given as:

$$(SoftLabels)\sigma(z_i) = \frac{\exp\left(\frac{z_i}{T}\right)}{\sum_{j=1}^{n} \exp\left(\frac{z_j}{T}\right)} \tag{1}$$

where z_i is the i-th logit, T is the temperature, and n is the number of classes.

$$(DTL)L_{KL}(p^s(\tau), p^t(\tau)) = \tau^2 \sum_j p_j^t(\tau) \log \frac{p_j^t(\tau)}{p_j^s(\tau)} \tag{2}$$

KL Divergence used in Distillation Loss (DTL), denoted as L_{KL} in equation (2), measures how different the Student's softened probability distribution is from the Teacher's. The goal of distillation is to minimize this divergence, meaning the Student Model adjusts its output probabilities to closely resemble the Teacher's. The equation sums over all classes, with the Teacher's probability for each class $p_j^t(\tau)$ guiding the log ratio between the Teacher's and Student's probabilities. This way, the Student is encouraged to mimic the Teacher's output distribution.

Additionally, the τ^2 scaling factor ensures that the temperature appropriately influences the loss. When temperature is high, it places more importance on aligning the relative probabilities across all classes, rather than just focusing on the top prediction.

Student Loss in KD uses the Cross-Entropy loss functions also known as "hard target loss" because it directly compares the Student Model's output with the true labels. This loss is typically combined with the distillation loss (which encourages the Student Model to mimic the Teacher's behavior). The equation goes like this:

$$(StudentLoss)\mathcal{L}_{\mathrm{CE}}(\mathbf{p}^s(1), \mathbf{y}) = \sum_j -y_j \log p_j^s(1) \quad (3)$$

The formula shown is the categorical cross-entropy loss, denoted as $\mathcal{L}_{CE}$. This type of loss is commonly used for classification tasks where $\mathbf{y}$ represents the true label (one-hot encoded) and $\mathbf{p}^s(1)$ represents the predicted probability distribution from the Student Model. In this equation: $\mathbf{y}_j$ is the true label for class j, $\mathbf{p}_j^s(1)$ is the probability assigned by the Student Model to class j.The summation is over all classes j. The loss measures the difference between the predicted probabilities $\mathbf{p}^s(1)$ and the true distribution $\mathbf{y}$. The goal is to minimize this loss during training, which encourages the Student Model to produce a probability distribution that closely matches the true labels.

Finally the total loss $\mathcal{L}$ is a weighted sum of the categorical cross-entropy (CE) loss between the Student's predictions and the true labels, and the Kullback-Leibler (KL) divergence between the softened outputs of the Student and Teacher Models.

The equation is:

$$(TotalLoss)\mathcal{L} = (1-\alpha)\mathcal{L}_{\mathrm{CE}}(\mathbf{p}^s(1), \mathbf{y}) + \alpha\mathcal{L}_{\mathrm{KL}}(\mathbf{p}^s(\tau), \mathbf{p}^t(\tau)) \quad (4)$$

In this equation: $\mathcal{L}_{\mathrm{CE}}(\mathbf{p}^s(1), \mathbf{y})$ is the standard cross-entropy loss between the Student's predictions $\mathbf{p}^s(1)$ (for a temperature of 1) and the true labels $\mathbf{y}$. $\mathcal{L}_{\mathrm{KL}}(\mathbf{p}^s(\tau), \mathbf{p}^t(\tau))$ is the Kullback-Leibler divergence between the softened output probabilities of the Student Model $\mathbf{p}^s(\tau)$ and the Teacher Model $\mathbf{p}^t(\tau)$, where τ represents the temperature applied to smooth the probabilities. α is a hyperparameter that balances the contribution of the cross-entropy loss and the KL divergence. When α is closer to 1, more weight is given to matching the Teacher's softened predictions, and when it is closer to 0, more weight is given to fitting the true labels.

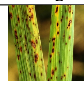
This combination of losses enables the Student Model to learn both from the ground-truth labels and the generalization captured by the Teacher.

V. EXPERIMENTAL ANALYSIS

1. Dataset Description

In this paper, we have selected three diseases and one healthy leaf. Those diseases are brown spot, leaf blast, leaf blight. The dataset we have used is a combination of three different datasets Dataset-1, Dataset-2, Dataset-3 are publicly available datasets collected from Kaggle and Roboflow. Table III shows some detail about the diseases.

Table III
SYMPTOMS OF PADDY DISEASES [17]

Type	Spot Shape	Spot Color	Image
Brown Spot	Circular or Pinhead shaped	Yellow brown	
Leaf Blast	Elliptical	Gray, white centers, brown to red brown margins	
Bacterial Leaf Blight	Stripes	Grayish green and curl up	

Total images are 3774 among them training, validation and testing set is split into 80%, 15%, 5% accordingly. We have used data augmentations in the original datasets to increase the variation for the Model which includes shear_range=0.2, zoom_range=0.2, horizontal_flip=True, rotation_range=20, width shift range=0.2, height shift range=0.2, vertical]_flip=True. So, the total images (1.928 x 3008) that the Model will be trained on will be 5799 images.

Allocating 80% of the data for training, 15% for validation, and 5% for testing enhances model development by supporting effective tuning and assessment. A larger validation set enables reliable hyperparameter optimization, model selection, and early stopping by providing a more accurate representation of unseen data, thus reducing the risk of overfitting or underfitting. It also ensures stable performance metrics, making it easier to gauge the model's progress during training. By keeping the test set minimal and untouched, it remains an unbiased measure of final model performance, ensuring that the test results accurately reflect generalization to truly new data.

2. Experimental Setup and Results

The training device is equipped with an AMD Ryzen 7 4800H processor,16 GB of RAM, and a dedicated NVIDIA GeForce 1650 GPU with 4 GB of VRAM. In order to compile the code, the team employed Jupyter Notebook and the TensorFlow libraries. TensorBoard was employed for visualising the data. We have leveraged DenseNet-121 as the reference Model for our knowledge distillation approach, transferring its learned weights and parameters to our Student Model. The training process began by running each model for 100 epochs, repeated five times, to select the best-performing Teacher model. The chosen Teacher model was then fine-tuned using Quantization-Aware Training (QAT), with three runs of 100 epochs each. In this scenario, we employed a parameter value of α = 0.6 and a temperature value of T = 6. Finally, Knowledge Distillation (KD) was applied to transfer knowledge from the Teacher to the Student model, combining KD with QAT over four separate runs of 100 epochs.

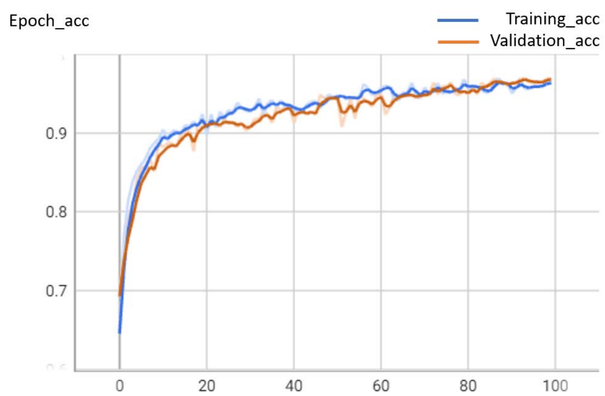


Figure 4. Distilled model training loss vs validation loss.

In the following figures Figure 4 shows how Distillation helped the Student Model to balance between Training Loss and Validation Loss with minimal over-fitting. Figure 5 depicts the Student Loss in various stages as the training carried on. Finally, Figure 6 shows in per epoch the Distillation loss got smoothed out as Teacher helped the Student to learn features more accurately.

When analyzing the Teacher Model in contrast to the Student Model, the Teacher Model had 10,54,725 trainable parameters and Student Model had 5,91,108. With a difference of 4,63,617 instances, the Student Model outperformed the Teacher Model, Table IV shows the performance of the

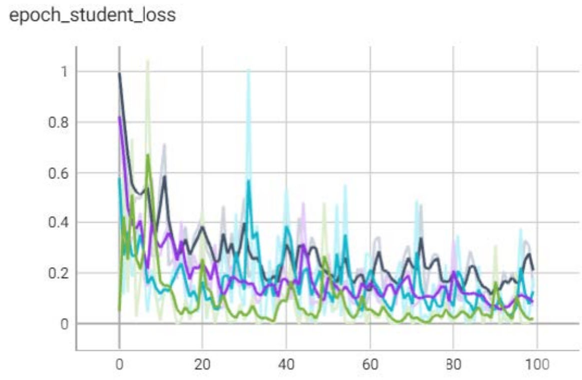


Figure 5. Student loss in multiple tuning stages.

Figure 6. Distillation Loss Per Epoch.

Student Model in multiple performance sections other than accuracy. These results came out to be the best with the combination of parameters we used and will vary depending on different parameter values and machine. Afterwards applying Full Quantization Table V reduced the size of the Student Model.

VI. CONCLUSION

In the course of our research, we successfully achieved identification of three types of diseases in paddy crops: Brown Spot, Leaf Blast, and Leaf Blight. The DenseNet-121 Model achieved the highest accuracy of 96.93%. Afterwards the bespoke dual channel Student Model successfully executed the learning task and surpassed the Teacher's threshold level with a 98.47% accuracy rate. The Student Model achieved increased accuracy with nearly 50% less parameters being optimized. This will facilitate expedited inference, compact footprint, and advantageous for devices with limited memory capacity as Quantization is applied afterwards though the accuracy is also significant 97.65%.

Table IV
DISTILLER MODEL CLASSIFICATION REPORT.

Labels	Precision	Recall	F1 Score
Brown Spot	1.00	1.00	1.00
Healthy	0.92	0.97	0.95
Leaf Blast	0.97	0.91	0.94
Leaf Blight	1.00	1.00	1.00

Table V
TRAINED MODEL ACCURACY REPORT.

Trained Models	**Accuracy**	**Size**
Teacher	96.93%	40.0 MB
Student	93.87%	38.7 MB
Student with KD	98.47%	38.7 MB
Student with KD+QAT+PQT	97.65%	18.6 MB

Future research endeavors in this area will involve the utilization of diverse instructional paradigms to educate Student Models across various proficiency levels.

REFERENCES

[1] S. Sen, R. Chakraborty, and P. Kalita, "Rice-not just a staple food: A comprehensive review on its phytochemicals and therapeutic potential," *Trends in Food Science & Technology*, vol. 97, pp. 265–285, 2020.

[2] "What is the importance of rice in bangladesh? kernel description," accessed: 2024-03-17. [Online]. Available: https://typeset.io/questions/what-is-the-importance-of-rice-in-bangladesh-4ibrf6r1us

[3] R. Reinke, S. Kim, and B. Kim, "Developing japonica rice introgression lines with multiple resistance genes for brown planthopper, bacterial blight, rice blast, and rice stripe virus using molecular breeding," *Molecular Genetics and Genomics*, vol. 293, no. 6, pp. 1565–1575, 2018.

[4] M. Agrawal and S. Agrawal, "Rice plant diseases detection & classification using deep learning models: A systematic review," *Journal of Critical Reviews*, vol. 7, no. 11, pp. 4376–4390, 2020.

[5] J. Chen, D. Zhang, Y. Nanehkaran, and D. Li, "Detection of rice plant diseases based on deep transfer learning," *Journal of the Science of Food and Agriculture*, vol. 100, no. 7, pp. 3246–3256, 2020.

[6] S. Jadhav, V. Udupi, and S. Patil, "Identification of plant diseases using convolutional neural networks," *International Journal of Information Technology*, vol. 13, no. 6, pp. 2461–2470, 2021.

[7] T. Daniya and S. Vigneshwari, "Deep neural network for disease detection in rice plant using the texture and deep features," *The Computer Journal*, vol. 65, no. 7, pp. 1812–1825, 2022.

[8] J. Li, X. Zhao, H. Xu, L. Zhang, B. Xie, J. Yan, L. Zhang, D. Fan, and L. Li, "An interpretable high-accuracy method for rice disease detection based on multisource data and transfer learning," *Plants*, vol. 12, no. 18, p. 3273, 2023.

[9] A. Sharma, A. Jain, P. Gupta, and V. Chowdary, "Machine learning applications for precision agriculture: A comprehensive review," *IEEE Access*, vol. 9, pp. 4843–4873, 2020.

[10] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," 2015. [Online]. Available: https://arxiv.org/abs/1503.02531

[11] A. Musa, M. Hassan, M. Hamada, and F. Aliyu, "Low-power deep learning model for plant disease detection for smart-hydroponics using knowledge distillation techniques," *Journal of Low Power Electronics and Applications*, vol. 12, no. 2, p. 24, 2022.

[12] Y. Hu, G. Liu, Z. Chen, J. Liu, and J. Guo, "Lightweight one-stage maize leaf disease detection model with knowledge distillation," *Agriculture*, vol. 13, no. 9, p. 1664, 2023.

[13] B. Petchiammal, Kiruba and P. Murugan, Arjunan, "Paddy doctor: A visual image dataset for automated paddy disease classification and benchmarking," in *CODS-COMAD '23*. Association for Computing Machinery, New York, NY, USA, 2023, pp. 203–207.

[14] S. Lamba, A. Baliyan, and V. Kukreja, "A novel gcl hybrid classification model for paddy diseases," *International Journal of Information Technology*, 2022.

[15] W. Chen, J. Chen, R. Duan, Y. Fang, Q. Ruan, and D. Zhang, "Ms-dnet: A mobile neural network for plant disease identification," *Computers and Electronics in Agriculture*, vol. 199, p. 107175, 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0168169922004926

[16] S. Nalini, N. Krishnaraj, J. Thangaiyan, K. Vinothkumar, A. Britto, K. Subramaniam, and C. Bharatiraja, "Paddy leaf disease detection using an optimized deep neural network," *Computers, Materials & Continua*, vol. 680, pp. 1117–1128, 2021.

[17] M. Niaz, L. Tarin, T. Shabonty, M. Saad, and D. Farid, "Paddy leaf disease detection employing transfer learning," in *2023 26th International Conference on Computer and Information Technology (ICCIT)*, 2023, pp. 1–6.

[18] C. Simhadri and H. Kondaveeti, "Automatic recognition of rice leaf diseases using transfer learning," *Agronomy*, vol. 13, no. 4, 2023. [Online]. Available: https://www.mdpi.com/2073-4395/13/4/961

[19] R. Narmadha, N. Sengottaiyan, and R.J.K., "Deep transfer learning based rice plant disease detection model," *Intelligent Automation & Soft Computing*, vol. 31, no. 2, pp. 1257–1271, 2022. [Online]. Available: http://www.techscience.com/iasc/v31n2/44553

[20] V. Gautam, N. Trivedi, A. Singh, H. Mohamed, I. Noya, P. Kaur, and N. Goyal, "A transfer learning-based artificial intelligence model for leaf disease assessment," *Sustainability*, vol. 14, no. 20, 2022. [Online]. Available: https://www.mdpi.com/2071-1050/14/20/13610

[21] M. Islam, M. Shuvo, M. Shamsojjaman, S. Hasan, M. Hossain, and T. Khatun, "An automated convolutional neural network based approach for paddy leaf disease detection," *International Journal of Advanced Computer Science and Applications*, vol. 12, no. 1, 2021.

[22] A. Ghofrani and R. Mahdian Toroghi, "Knowledge distillation in plant disease recognition," *Neural Computing and Applications*, vol. 34, no. 17, pp. 14 287–14 296, 2022. [Online]. Available: https://doi.org/10.1007/s00521-021-06882-y

[23] Y. Hu, G. Liu, Z. Chen, J. Liu, and J. Guo, "Lightweight one-stage maize leaf disease detection model with knowledge distillation," *Agriculture*, vol. 13, no. 9, 2023. [Online]. Available: https://www.mdpi.com/2077-0472/13/9/1664

[24] A. Musa, M. Hassan, M. Hamada, and F. Aliyu, "Low-power deep learning model for plant disease detection for smart-hydroponics using knowledge distillation techniques," *Journal of Low Power Electronics and Applications*, vol. 12, no. 2, 2022. [Online]. Available: https://www.mdpi.com/2079-9268/12/2/24

[25] "Rice leaf disease detection obj dataset," visited on 2024-03-18. [Online]. Available: https://universe.roboflow.com/project-khcjh/rice-leaf-disease-detection-obj

[26] "Rice leaf diseases dataset kernel description," accessed:

2023-08-14. [Online]. Available: https://www.kaggle.com/datasets/vbookshelf/rice-leaf-diseases

[27] "Rice diseases image dataset," accessed: 2023-08-14. [Online]. Available: https://www.kaggle.com/datasets/minhhuy2810/rice-diseases-imagedataset/data

Md. Saiful Niaz holds a Bachelor of Science in Computer Science and Engineering from United International University. His academic and research pursuits focus on machine learning, deep learning, computer vision, and natural language processing, fields in which he has contributed to several peer-reviewed conference publications. Mr. Niaz is also credentialed in networking and cybersecurity through Cisco's CCNA program, underscoring his technical proficiency across programming, web development, and robotics. His background reflects a commitment to advancing knowledge in applied computing and intelligent systems.
Email: mniaz201152@bscse.uiu.ac.bd

Tasmia Hassan Shabonty is currently completing her BSc in Computer Science at United International University, Bangladesh. She was awarded the Study in Canada Scholarship 2022 and spent a semester at Lakehead University, Thunder Bay, Canada. Her research interests span artificial intelligence, machine learning, deep learning, and human-computer interaction (HCI). She is particularly focused on leveraging AI and ML techniques to enhance user interactions and learning experiences in digital platforms. In her recent work, she explores how advanced AI models can be applied within HCI frameworks to create more intuitive and adaptive systems that support user needs.
Email: tshabonty201099@bscse.uiu.ac.bd

Md. Ashfaqur Rahman Saad is currently completing his BSc in Computer Science at United International University, Bangladesh. His research interests include machine learning, deep learning, artificial intelligence, computer networks, and cybersecurity. Saad is particularly interested in the intersection of AI and cybersecurity, focusing on how machine learning models can be leveraged to enhance network security and protect against emerging cyber threats. His work aims to develop intelligent systems capable of adaptive threat detection and response to strengthen digital infrastructure.
Email: msaad201075@bscse.uiu.ac.bd

Lucky Talukder Tarin is a Computer Science and Engineering graduate from United International University, is committed to advancing technology through impactful research and innovative solutions. Her recent publications highlight her dedication to creating tools that promote transparency, sustainability, and improved outcomes across various sectors.
Email: ltarin201070@bscse.uiu.ac.bd

Dewan Md. Farid is a Professor of Computer Science and Engineering at United International University. He is an IEEE Senior Member and Member ACM. Prof. He holds a PhD in Computer Science and Engineering from Jahangirnagar University, Bangladesh in 2012. Part of his PhD research has been done at ERIC Laboratory, University Lumière Lyon 2, France by Erasmus-Mundus ECW eLink PhD Exchange Program. His PhD was fully funded by Ministry of Science and Information and Communication Technology, Government of the People's Republic of Bangladesh and European Union (EU) eLink project. Prof. Farid has published 142 peer-reviewed scientific articles, including 33 highly esteemed journals like Expert Systems with Applications, IEEE Access, Journal of Theoretical Biology, Journal of Neuroscience Methods, Bioinformatics, Scientific Reports (Nature), Proteins and so on in the field of Machine Learning, Data Mining and Big Data.
Email:dewanfarid@cse.uiu.ac.bd

Huu-Hoa Nguyen received his Engineering Degree in Computer Science from Can Tho University, Vietnam. He earned his MSc in Information Systems from HAN University, the Netherlands, and his PhD in Informatics from Lyon University, France. Dr. Nguyen is currently a senior lecturer at the College of Information and Communication Technology, Can Tho University, Vietnam. His research interests span a wide range of computer science topics, including artificial intelligence, machine learning, data mining, knowledge management systems, computer networks, and cybersecurity. He has led multiple international research projects, receiving funding from prestigious sources such as the Horizon-2020 European Commission and the Newton Fund.
Email: nhhoa@ctu.edu.vn