

# AgriCure: A web application based layered augmentation-enhanced YOLOv8 for disease and nutrient deficiency detection in bitter gourd leaves

**Abstract**—Bitter gourd is an important cucurbitaceous vegetable widely grown in India and other tropical and subtropical regions and appreciated for its nutritional, medicinal, and economic values. Traditional way of detecting diseases and nutrient deficiencies in bitter gourd leaves requires significant effort and expertise whereas, precision farming and automated disease detection methods can greatly support farmers by facilitating sustainable agriculture. To address this challenge a novel web based application AgriCure was developed which incorporated a multilevel approach to detect the plant disease and nutrient deficiency with high level. It uses a hybrid augmentation-based YOLOv8 DL model for image analysis. The study focuses on detecting diseases like Downy Mildew, Leaf Spot, and Jassid, as well as nutrient deficiencies such as Potassium, Magnesium, and Nitrogen Deficiency and their combinations. The initial dataset of 785 images was increased to 6396 images using advanced data augmentation. The results on the augmented dataset after 100 epochs demonstrated high effectiveness with the augmented dataset. The model achieved an impressive mean Average Precision (mAP50) of 97.0% at an Intersection over Union (IoU) threshold of 0.50 and a mAP50-95 of 94.5% across IoU thresholds from 0.50 to 0.95. Nearly all predicted positive instances were true positives, with a precision rate of 91.8% and a recall of 90.2%, which showed the capacity of the model in identifying true positives. The F1 score of 94.7 highlighted balanced performance of the model between precision and recall, emphasising its reliability and accuracy. The model shows low losses, with a Box loss of 0.1405, a Class loss of 0.1116, and a Distribution Focal Loss (dfl loss) of 0.8192. This approach offered a valuable tool for early and accurate detection of disease and nutrient deficiency. Detection results indicate that, compared to previous methods, the proposed approach significantly improves overall performance and addresses challenges tied to limited dataset sizes.

**Index Terms**—Bitter gourd, Disease detection, Nutrient deficiency detection, YOLOv8, Layered augmentation, Deep learning

## I. GRAPHICAL ABSTRACT

## II. HIGHLIGHTS

A novel web-based application was developed for the automated detection of diseases and nutrient deficiencies in bitter gourd using YOLOv8. The model accurately detects multiple diseases along with nutrient deficiencies of K, Mg and N. The mAP of 97% at IoU 0.50 was achieved, and the model demonstrated high accuracy, with precision of 91.8% and recall of 90.2%.

## III. INTRODUCTION

Bitter gourd is a commercially important and nutritionally rich vegetable widely recognized for its significant medic-

inal properties, including managing diabetes and boosting immunity [1][2]. However, its high susceptibility to nutrient deficiencies poses a significant challenge to farmers [1] [3]. The development of an AI-based automated system, which can detect diseases in a timely manner, can help make timely decisions to enhance productivity and increase the quality of crops. Like other plants, the bitter guard leaves also suffer from many types of nutrient deficiencies, which significantly affect the growth and productivity of the plant. The deficiency of micro and micronutrients can affect the plant's physiological and biochemical behaviour [4][5]. The production of the crop can be enhanced by addressing these issues. Nitrogen (N) deals with the chlorophyll production and health status of the plant. The deficiency of N leads to a lower content of chlorophyll, which leads to chlorosis (leaf yellowing) and dwarfing in the plant. Nitrogen deficiency can reduce the photosynthetic capacity of plants and their overall growth [6] [7][8]. The other nutrients have an important role in energy transfer and storage in the plant. The phosphorus deficiency leads to a dark green or purplish colour of leaves and stunted growth. Its deficiency also leads to poorly developed roots and delayed maturity, which significantly affect the yield [7][8][9][10]. The potassium nutrient works in the water management activity and activates the enzymes in the plant. Its deficiency leads to chlorosis along leaf margins that may advance to necrosis, weak stems, and poor fruit quality. Its deficiency also reduces the resistance of the plant to various stresses and diseases [11][12][13][14]. Though the requirement of the macronutrients in the plant is very low, their role in plant functionality is very crucial. The zinc deficiency leads to stunted growth and small leaves. The iron (Fe) deficiency leads to interveinal chlorosis. The boron (B) deficiency affects the cell wall development and reproductive growth of the plant, which leads to poor quality of the fruits [15][16][17]. The nutrient deficiencies in bitter gourd also have a direct impact on the growth characteristics of bitter gourd, such as plant height, number of leaves, and fruit production [3]. The appropriate boron level in bitter gourd has been shown to enhance growth and production, whereas its deficiency leads to lower fruit yield [16][17]. Like other plants, the shortage of nutrients in bitter guard leads to photosynthesis, water movement, and nutrient uptake. For example, the N and P deficiency can reduce photosynthetic efficiency, and the potassium deficiency can affect water balance and tolerance to various biotic and abiotic stresses [4] [12][18]. Furthermore,

nutrient deficiency can affect the quality parameters of the bitter melon, like vitamin levels and fruit size. In order to maintain the nutritional and medicinal values of the bitter melon, it is important to maintain the adequate nutrients of the plant [3][8]. The conventional methods are prone to error, and a lack of domain experts is always a challenge in the field of agriculture; therefore, there is a prominent need for a computer-based solution that can guarantee a solution is required. In contrast to the conventional approaches, the proposed approach with augmentation provides a new approach towards the detection of disease and nutrient deficiency in bitter melon leaves. In order to provide a feasible solution for the real-time application, a web app, AgriCure, was developed, which can be accessed at [https://github.com/sumit-kumar-2104/Agri\\_Cure](https://github.com/sumit-kumar-2104/Agri_Cure). It is a multilayer plant disease and nutrients detection application that uses YOLOv8 and an advanced augmentation approach, which can be run in real time. It can be used to detect disease and nutrient deficiency in bitter melon in real time, thus contributing to agricultural productivity. Various CNN architectures have been applied in detecting the disease and nutrient deficiency in images due to their capability to learn the spatial hierarchies of features, for instance, ResNet-50 with a 98.18% accuracy [19][20]. Also, the Graph Convolutional Networks (GCNs) have been used to increase the accuracy of plant diseases and nutrient deficiencies by learning regional features using spatial pyramidal pooling. It is referred to as PND-Net, which achieved state-of-the-art performance across various datasets [21]. Aabidi et al. [22] used the augmented training dataset for leaf disease detection, which leads to increased model accuracy in detecting leaf diseases and predicting pest detection [22]. The transfer learning is also a very important technique that uses a pre-trained model on various datasets, such as ImageNet, to enhance the performance of the model in a particular task, for instance, plant disease detection [22]. Other CNN-based studies have shown high accuracy rates for plant disease and nutrients detection, for example, a CNN model attained 99% accuracy in ginger plant disease detection [23]. Another study indicated a 96% accuracy for a model constructed using a heterogeneous dataset with CNN [24]. For real-time applications, the DL models have been integrated with mobile or cloud-based platforms, which offer former actionable insights [25]. The environmental conditions are full of variability, like lightning and background noise, which can affect the performance of the DL model. Also, the limited training data can restrict the model from generalising for various crop varieties and environmental conditions [23]. For real-life practical applications, the latest development of DL models and diverse datasets is crucial for maintaining accuracy. Hence, integration of the DL model with the data augmentation leads to a significant method for identifying the disease and nutrient deficiency in the bitter melon leaves. These DL models can boost agricultural productivity and sustainability and may solve global food security issues. Many DL models like VGG16, EfficientNetB0, ResNet-50, and GoogleNet have also been utilised for plant disease detection with high accuracy. For example, VGG16 with

a Dense Layer obtained 99.747% accuracy in rice disease detection [26], and the ResNet9 was utilised to examine leaf features like colour and texture, which leads to improving diagnostic accuracy through image classification and segmentation [27]. DenseNet121 and VGG16 have been compared to determine their efficiency in the diagnosis of plant diseases, and DenseNet121 has been noted to have superior generalisation performance [28]. An ensemble-based model based on ResNet V50, MobileNetV2, and EfficientNet-B0 demonstrated an accuracy of 99.351% in cucumber leaf disease detection [29]. DenseNet169 is attaining a high accuracy of 99.72% with deep retraining, as it does for plant disease detection [30]. VGG16, though good, demonstrates lesser accuracy compared to other models like EfficientNet B3 [31]. EfficientNet B3 excels in high accuracy and performance and, therefore, is a good model for use in resource-poor environments [31]. GoogleNet also demonstrated competitive outcomes but is no match for EfficientNet B3 [31]. ResNet50 attains 96.49% validation accuracy, which renders it adept at distinguishing plant diseases [32]. ResNet34 and Inception\_v3 also perform well, and ResNet34 has competitive outcomes for some applications [33][34]. YOLOv8 with additions like GhostNet module and Coordinate Attention mechanism demonstrated high accuracy and performance, and therefore is an excellent performer for real-world uses [35]. YOLOv5, in comparison to YOLOv8, demonstrates better performance on some sets and therefore is appropriate for some plant disease identification tasks [36]. Although choosing architecture would be application-specific specifications like computational demand and precision/recall balance, the flexibility of DL models can be constantly developed and optimised. Flexibility is very much needed in light of the requirement for real-time performance and precision in plant disease detection [37]. The PND-Net utilizes GNNs to augment feature learning for disease classification and micronutrient deficiency, with very high accuracy on banana and coffee plant datasets [21]. CNN with Skip Connections (CNN-SC), used for the detection of micronutrient deficiency in banana leaves, performed better than conventional architectures such as VGG16 and DenseNet, with approximately 95% accuracy [38]. Mask R-CNN and YOLO models were used for tomato plants, with 92% and 98% accuracy in nutrient deficiency classification, proving the efficacy of region-based detection techniques [39]. By combining DL models with IoT devices such as Raspberry Pi, real-time prediction and monitoring of nutrient deficiencies can be obtained with prompt feedback to farmers [40]. Methods such as SHAP and Grad-CAM are used to explain CNN decisions, balancing model interpretability with accuracy. This is essential to establish trust in AI systems being used in agriculture [41]. Although DL models provide valuable advantages in predicting nutrient deficiency, limitations lie in the data availability, model interpretability, and computational costs. Integration of such models with IoT devices and application of explainable AI methods can improve their usability and practicality in practical agricultural applications. Moreover, compact model development, such as that of MobileNet,

ensures the applicability of such solutions in even constrained-resource environments. CNN-based mobile apps have been established for real-time detection of plant diseases, which provide farmers with early intervention services [42]. IoT-based platforms have been constructed for disease detection and data gathering, indicating global applicability possibilities [42]. A system incorporating InceptionV3 and optimized fertilizer design has been proposed for optimal resource usage and sustainable cultivation [44]. DL model performance relies on large datasets, which are necessary for effective disease detection [29]. Data augmentation and hyperparameter tuning are critical in order to support model generalization and address class imbalance [21]. Future research can focus on making models more scalable and computationally efficient to allow real-world usage in agriculture [19]. Though DL offers hopeful solutions for the detection of plant diseases, it is still being challenged by problems of data unavailability and the complexity of the model. Eliminating these concerns through innovative methods and current studies will play an important role in the general applicability of DL in agriculture. This integration of technology supports precision farming and accords with initiatives at the international level to secure food and guarantee sustainable agricultural applications. DL models have been exceptionally promising in plant disease detection, with several models producing high-accuracy results. Comparison among models indicates that while all the models have a lot to offer, some yield better results compared to others, depending on the situation. DenseNet121, for instance, is well known to have superior performance in generalizing and is very computationally effective, so it is an appropriate candidate in real-world situations [19]. On the contrary, when heavily retrained, DenseNet169 and AlexNet have high accuracy levels of 99.72% and 99.24%, respectively, and therefore are suitable for detecting plant diseases [30]. DL models have been up-and-coming, with several models giving high accuracy levels [45][46]. The comparison among different architectures reveals that while all models are robust, some models are superior under specific conditions. For instance, DenseNet121 has superior generalization and computational efficiency and is a potential contender for real-world applications [19].

#### IV. MATERIALS AND METHODS

##### V. DATASET

The OLID dataset contained 4,749 images of different diseases of various species of plants in the state, mainly nutritional deficiencies, insect attacks, and a healthy state of the plant leaf. Specifically, this research is guided by detecting these issues in the Bitter gourd plant species, thereby extracting 785 images of healthy leaves damaged due to nutrient-related issues and insect-affected leaves. The dataset from which this was extracted had two types of diseases that affected the plants, namely Downy Mildew and Leaf Spot, and one type of insect that affected the plants, namely Jassid. In addition, plants suffered from five nutrition deficiencies: potassium, potassium and magnesium, nitrogen and potassium, and nitrogen and magnesium. Roboflow was used to ease

and perfect preprocessing, augmentation, and labelling images. Detailed information about the labels is shown in Figure 1, whereas Table 1 provides an overview of the classes and their sample sizes. Figure 1: Dataset visualisations for bitter gourd leaf classification, including class distribution (top-left), bounding box patterns (top-right), and scatter plots of object centroids (bottom-left) and aspect ratios (bottom-right). Table 1: Bitter gourd leaf dataset classification

##### VI. DATA SPLITTING

Various augmentation techniques were applied to enhance the dataset and improve model generalization, such as rotation, flipping, noise addition, Blurring, adding saturation. These augmentations were evaluated based on the resulting change in mean Average Precision at a 50% Intersection over the Union threshold, forming the base of our proposed dataset. The original dataset was initially splitted into 70:30 ratio for train and test respectively resulting in 548 for train and 237 for test. Data augmentation techniques were exclusively applied on train part of dataset only in order to enhance it's generalizability. Later the train part after augmentation was splitted into train and valid part of dataset combined with the Un-augmented part of dataset resulting in the ratio of 81:15:4 for train, valid, test respectively. It is important to note that the test subset contained only the 30% images of original dataset and this 30% was chosen so that the model can be tested on an sufficient amount of original images. The final augmented dataset was split into 81%, 15%, and 4% for training, validation, and testing, respectively, as shown in Table 2. The 81:15:4 ratio was chosen because it strikes an ideal balance between maximizing training data for effective model learning and retaining sufficient validation and testing data for accurate performance monitoring and generalization assessment. Other ratios, like 90:5:5, risk insufficient validation feedback, while splits like 70:20:10 reduce training data, potentially compromising model accuracy. The associated dataset has been provided in data availability section. Training set (81%): Used to train the model. Validation set (15%): Used to tune hyperparameters and prevent overfitting. Test set (4%): Used for the final evaluation of the performance of the model. Table 2: Bitter gourd dataset split into training, validation, and testing after augmentation

##### VII. PREPROCESSING

Preprocessing is the step where raw data is converted into a format that can be inputted into the YOLOv8 model, facilitating the automation of the pipeline. This process resolves inconsistencies and eliminates duplicates, which could reduce the accuracy of the model. It also ensures that inaccuracies and missing values, caused by either human errors or technical glitches, are removed. Moreover, image augmentation helps reduce training time while improving model performance. Correct image rotations, often stored in EXIF metadata, are standardized through auto-orientation techniques to maintain consistent pixel alignment. To ensure uniformity and decrease file size, images are resized to 640x640 pixels, which also helps

speed up training. Contrast adjustments are done automatically to enhance edge detection around objects. The Roboflow was used to augment and annotate the sample images. YOLOv8: You Only Look Once Version 8 The YOLO algorithm has gained popularity as a highly effective and efficient method for object detection in computer vision. YOLOv8, the most recent version, introduces significant improvements in speed and accuracy over previous versions. It retains the core principle of real-time object detection with high precision. YOLOv8 employs a single convolutional neural network (CNN) to infer bounding boxes and class probabilities in a single pass from full images, unlike traditional two-stage detectors. The architecture of YOLOv8 integrates state-of-the-art methodologies from DL and computer vision, such as advanced feature extraction layers, anchor box clustering optimization, and new training techniques. This flexibility enables efficient detection of objects across various sizes and classes, with wide-ranging applications in research and industry. The experiments in this paper utilize the YOLOv8n model.

### VIII. EXPERIMENTAL SETUP

In this research, we utilized a dataset of bitter melon leaves to classify leaf diseases and applied the YOLO model, initially pre-trained on the COCO dataset. The experimental setup was conducted in a Jupyter Notebook environment, leveraging the capabilities of an NVIDIA GeForce RTX 3050 Ti GPU with 4 GB of VRAM, integrated alongside Intel® Iris® Xe Graphics with an additional 4 GB of shared memory. The system operated on 32 GB of DDR5 RAM, powered by an Intel® Core™ i7 12th-generation processor. We employed the nano version of YOLOv8, referred to as YOLOv8n, and trained it on various augmented datasets. The optimization process utilized the AdamW optimizer, with a learning rate set at 0.01 and a momentum coefficient of 0.937. The YOLOv8n model, consisting of 168 layers and 3,006,818 parameters, was built with zero gradients and required 8.1 GFLOPs for computation. The model was optimized using Python-3.10.14 torch-1.11.0+cu115. The parameters of the model are detailed in Table 3, and the associated YAML configuration (Supplementary 1) file has been provided as supplementary content. Table 3: Important parameters for YOLOv8 training

### IX. PROPOSED DATA AUGMENTATION PROCESS

In our study, the original dataset consisted of 785 images of leaves across nine distinct classes: Healthy (H), Downy Mildew (DM), Leaf Spot (LS), Jassid (JAS), Potassium and Magnesium Deficiency (K Mg), Nitrogen Deficiency (N), Nitrogen and Potassium Deficiency (N K), Nitrogen and Magnesium Deficiency (N Mg), and Potassium Deficiency (K). The dataset was first split into 70:30 for train and test. To ensure an unbiased evaluation of model performance as shown in Figure 2, the 30% test dataset was kept as untouched. The remaining 70% portion of dataset was used for train and validation subset and used for further augmentation. However, some classes were underrepresented in comparison to others, making it necessary to balance the dataset prior to applying

augmentation. To address this, horizontal and vertical flipping with a 50% probability for each was applied to the underrepresented classes, effectively equalizing the number of images across all nine categories. This balancing step increased the dataset from 785 images to 1,492, represented in Figure 3. Following the balancing process, data augmentation was applied uniformly to all classes to expand the dataset further. This augmentation increased the dataset size from 1,492 images to 6,396 images. The proposed augmentation techniques improve model generalization by artificially increasing the training data while preserving the essential features of the image. Figure 2: Dataset processing steps and the splitting of dataset initially into 70:30 ratio in train and test and later combining after a series of processes in the ratio of 81:15:4 for Train, Valid, Test respectively. The steps of augmentation were as follows: Rotation by 90°: Images were rotated by 90° clockwise, 90° counterclockwise, or 180° (upside down). Saturation adjustment: After rotation, to simulate variations in lighting and environmental conditions, the saturation was adjusted between -25% and +25%. Salt and pepper noise: Finally, 0.1% of the pixels of salt-and-pepper noise were added, which is a type of stochastic noise that can occur in real-world data collection processes, such as sensor noise or environmental factors. When the size of the dataset is low, the augmentation caters for the problem of overfitting and enhances the robustness of the model. it further improves the generalization by exposing the model to different transformations on the original dataset. Each augmentation technique was selected based on its ability to simulate real-world variations while retaining the essential features of the leaf images. Their impact on performance metrics further demonstrates the importance of these augmentation techniques. Table 4 shows the effect of each technique on the mean Average Precision at 50% Intersection over Union (mAP50) over 100 epochs. As illustrated in Table 4, the introduction of techniques like noise significantly boosted the performance of the model, increasing mAP50 from 0.698 (without augmentation) to 0.945 when both saturation and noise were applied. Table 4: Effect of various augmentations on map50 scores across 100 epochs While the initial dataset used in this study is relatively small, extensive augmentation increased it from 785 to 6,396 images, simulating real-world variability in lighting, orientation, and environmental conditions. Augmentation techniques like random rotation, saturation, and noise have been proven to enhance model robustness in agricultural disease detection [47]. Although augmentation improves generalization, future research will explore synthetic data generation using GANs or integrating external datasets to expand data diversity [48]. Figure 3: The proposed model for data augmentation with different operations like 90° rotation (clockwise/counterclockwise), saturation variation (-25% to +25%), salt and pepper noise (up to 0.5%) and dataset balancing approach As shown in Figure 3, each image in the dataset was processed sequentially through a layered augmentation pipeline. The use of augmentations such as rotation, saturation, and noise were informed by their ability to increase variance in a way that preserves critical

image features, ultimately enhancing the learning capability of the model. This pipeline was designed to expose the model to various forms of transformations while maintaining the underlying structure of the data, leading to improved performance across all metrics.

## X. RESULT AND EVALUATION METRICS

### XI. DEVELOPMENT OF A WEB APPLICATION

In this study, we developed a web-based application, AgriCure, to create a user-friendly platform for plant disease detection, as depicted in Figure 5. The application begins with a login or signup page that allows users to access their accounts securely. The login page has an option to select the plant. The user can upload or capture an image from a smart smartphone through a web browser and upload the image to the model. An object rejection model further rejects the image if it is not a leaf and filters out non-plant images, such as those containing humans or unrelated objects. Then Plant Detection Model which identifies that the input image is a valid plant image or invalid test case such as a green screen or unrelated object. Further, the proposed model ensures that only a high-confidence image with a precision threshold above 75% will proceed further to the disease detection model. To determine optimal confidence threshold value for the proposed model a comprehensive analysis of key metrics was done on a test dataset (LINK) made specifically for threshold calculations of the model. It consists of mixture of test plant images and unrelated random object images, the metrics for calculating threshold included precision, recall, F1 score, accuracy, false positive rate, and false negative rate as shown in the Figure 4. The analysis was done on various threshold ranging from 0.10 to 1.0 at the interval of every 0.05 and the results indicated that the threshold value of 0.75 that is 75% strikes the best balance offering a meeting point of precision, recall, F1 score, accuracy. Most importantly at this threshold, the false negative value which is crucial in multi layered pipeline is very low after 75% the false negative values increases at a high pace. Therefore, applying a threshold of 75%, the model ensures the real-life applicability and ensure only high confidence, truly plant related images proceed further which improves the reliability of the system and reducing the risk of misinformation in critical decision making of agricultural practices. The proposed model further identifies plant diseases or nutrient deficiencies, such as downy mildew, leaf spot, and Jassid, as well as deficiencies in potassium, magnesium, or nitrogen. If a disease is detected, users are provided with an option to learn more about the disease. This redirects them to an information page containing detailed insights into the identified disease, including its symptoms, causes, and potential management practices. The AgriCure web application seamlessly integrates advanced YOLOv8 technology with an intuitive design, providing a reliable tool for farmers and researchers to diagnose plant health efficiently. The link to the application is provided in the data availability section. Figure 4. Threshold sensitivity analysis for precision plant validation model on the basis of metrics Precision, Recall, F1 Score, Accuracy, false Positive,

False Negative Figure 5: The deployment of a web application Integrated with the Yolov8 model and with the integration of 2 separate object detection and plant detection models. It depicts different stages of App Home, Login/Signup, Selecting Plant to detect, Selecting picture of the plant, Results, which if the plant is not present will return “No”, and if the plant is present will return the disease present in it with details of it.

### XII. EVALUATION METRICS

The performance of the model was evaluated using several metrics to assess its accuracy in detecting and classifying different leaf conditions. The primary metrics used include precision, recall, F1 score, mAP. These metrics are calculated based on fundamental parameters such as True Positives (TP), False Positives (FP), False Negatives (FN), and True Negatives (TN), and are described below: • Accuracy is calculated by dividing the number of correct predictions by the total number of predictions • Precision: Precision measures the proportion of correct positive predictions out of all positive predictions made by the model. • Recall: Recall measures the proportion of true positives detected out of all actual positives in the dataset. • F1 Score: The F1 score is the harmonic mean of precision and recall, balancing the two metrics. • Mean Average Precision (mAP): mAP is the mean of the Average Precision (AP) across all classes, where AP is the area under the precision-recall curve. mAP is typically evaluated at different IoU thresholds, such as 50% (mAP50) and between 50% and 95% (mAP50-95). • mAP50(B): This is the mean Average Precision calculated specifically for bounding box detection at an IoU threshold of 50%.

### XIII. RESULT ANALYSIS OF THE PROPOSED APPROACH

The results of the proposed augmented dataset are highly impressive, as evidenced by the performance metrics in Table 6 and Supplementary 2. The proposed model achieves an outstanding mAP of 97 at an Intersection over the Union (IoU) threshold of 0.50 and 94.5 across IoU thresholds ranging from 0.50 to 0.95. Precision reaches 91.8%, meaning maximum of predicted positive instances correspond to true positives, while recall is 90.2%, demonstrating that the model successfully identifies maximum of true positives. The F1 score of 94.7 further underscores the balanced performance between Precision and Recall. Among the classes, N (Nitrogen deficiency) and Healthy images show excellent Precision (91.4%), indicating extremely less false positives, but the Recall for Healthy images is slightly lower (89%), suggesting a few missed detections. Similarly, N K (Nitrogen-Potassium) and N Mg (Nitrogen-Magnesium) perform consistently well, with all metrics at or near 97%. The lowest Precision (60.6%) is observed in LS, but its Recall (100%) compensates, leading to balanced mAP values and the precision of class DM is perfect (100%) indicating no false positives. The results on the original dataset (unaugmented dataset) are shown in Table 5. This result is not very impressive due to the limited original dataset. Table 5: The outcome of the model on the original dataset (unaugmented dataset) of different classes. Table 6:

Final outcome of the proposed approach for different classes on the augmented dataset. The model exhibits low losses, with a Box loss of 0.1405, a Class loss of 0.1116, and a Distribution Focal Loss (dfl loss) of 0.8192. These low-valued parameters of the model suggest minimal error in detection and high accuracy. The graph in Figure 5 illustrates the training and validation performance of the YOLOv8 model over 100 epochs, which shows clear improvements in key metrics. The box, classification, and distribution focal losses all decrease significantly, reflecting effective learning of object localization and classification. Both training and validation losses steadily decline without signs of overfitting. Furthermore, as shown in Figure 6, precision and recall increase sharply, nearing 1.0, which confirms the excellent performance of the model in detecting true positives and identifying relevant instances. The mAP50 stabilizes around 97%, while the mAP50-95, which evaluates detection across various IoU thresholds, converges near 94.5%. In particular, the box loss, class loss, and distribution focal loss in both training and validation phases show a steady downward trend over the epochs, while the Precision and Recall metrics improve rapidly, stabilising at near-perfect values. The overall model performance indicates robustness, with strong generalization capabilities, and it demonstrates exceptional accuracy across all evaluated metrics. Figure 6: Training and validation performance metrics of the YOLOv8 model over 100 epochs. The top row shows training box loss, classification loss, distribution focal loss, precision, and recall. The bottom row illustrates validation box loss, classification loss, distribution focal loss, and mAP metrics at IoU thresholds 0.50 (mAP50) and 0.50-0.95 (mAP50-95). The confusion matrix in Figure 7 concisely evaluates the classification performance of the model across various categories. The diagonal elements represent the true positives, where the predicted class matches the actual class. The model correctly classified 69 instances of "DM" 49 of "JAS" 73 of "K" and 103 of "healthy" during validation set. These correct classifications highlight the ability of the model to differentiate between classes. The off-diagonal elements reflect minimal misclassifications (false positives and false negatives). There was only one misclassification of "K" as "background" and two "healthy" sample classified as "background". With only 26 misclassifications out of 964 total predictions, the model demonstrates high accuracy and robustness, with very little confusion between categories. Overall, the confusion matrix of the model of validation set in Figure 8 underscores the strong performance, with a significant majority of correct predictions and minimal misclassifications, attesting to its effectiveness in classifying diverse categories. The confusion matrix in Figure 8 concisely evaluates the classification performance of the model across various categories on the original dataset, whereas Figure 7 shows the confusion matrix of our proposed approach. Figure 7: Confusion matrix of validation set for classification model performance on the augmented dataset Figure 8: Confusion matrix of validation set for classification model performance of the original dataset without augmentation The four plots in Figure 9 illustrate the classification performance of the

model across key metrics: recall, precision, and F1-score, with respect to confidence. The Recall-Confidence Curve shows that recall remains high (near 1.0) for all classes, dropping only at the highest confidence levels, indicating the model correctly identifies most positive instances. The Precision-Recall Curve reflects an almost perfect precision (0.995) for all classes, with a strong balance between precision and recall. In the Precision-Confidence Curve, precision rises sharply with confidence, stabilizing near 1.0 for confidence values above 0.8, indicating high accuracy in positive predictions at higher confidence thresholds. The F1-Confidence Curve demonstrates that F1-scores also rise and remain near 1.0 with increasing confidence, which shows a strong balance between precision and recall across all classes. Figure 9: Performance Metrics: Recall-Confidence, Precision-Recall, Precision-Confidence, and F1-Confidence Curves Figure 10 visually compares the actual and predicted labels for leaf disease detection. The bounding boxes and associated labels show the minimal discrepancies between the actual and predicted classes, which indicates the high accuracy of the model. The confidence scores displayed with the predicted boundary box exhibit the high reliability of the model, as high confidence is consistently associated with accurate classifications. Figure 10: Visual comparison of actual versus predicted leaf classifications

#### XIV. DISCUSSION

DL algorithms offer notable advantages in terms of accuracy, precision and recall in recognizing crop pests, nutrient deficiency and diseases. However, there is a paucity of studies on bitter guard plants in terms of disease and nutrient detection. The present study focuses on either disease detection or nutrient deficiency detection. To the best of our knowledge, no study has been reported that combines disease and nutrient deficiency detection. This study addresses this issue by employing a layered augmentation-based YOLOv8 model. The proposed model results showed a massive improvement in detecting disease and nutritional deficiencies in bitter guard plant leaves compared to previous studies. Our proposed model used a YOLOv8 model combined with a layered augmentation scheme. The result of the proposed model (Table 6) shows a significant improvement with layered augmentation in comparison to original dataset (Table 5) also shown in the confusion matrix in Figure 6 and Figure 7. Using the proposed augmented dataset yields nearly perfect metrics In contrast, the model trained on the original dataset attained only about 69.80% mAP50, Precision 52.7%, Recall 72.7%, and mAP50-95 68.4%. Importantly, every class in the proposed model has precision and recall near 90-91%, which reflects a balanced, uniform performance. These near-perfect, class-consistent results highlight the reliability and robustness of the proposed approach. The layered augmentation not only expanded the data massively but also yielded significantly improved, stable detection across all disease and deficiency categories. The proposed model achieved an excellent mAP of 97 at an Intersection over the Union (IoU) threshold of 0.50 and 94.5 across IoU thresholds ranging from 0.50 to 0.95. Pre-

cision reaches 91.8%, meaning maximum of predicted positive instances correspond to true positives, while recall is 90.2%, demonstrating that the model successfully identifies almost all true positives. The F1 score of 94.7 further underscores the balanced performance between Precision and Recall. This was an achievement that stood out more than the performances of models in a number of key prior studies. For example, Tran et al. [49] reported an accuracy of 87.27% from the use of the Inception-ResNet v2 model to detect nutrient deficiencies but found that the model lacked the strength to detect calcium and N deficiencies correctly. It can be observed that the YOLOv8-based model works better for all classes of micronutrient deficiencies. This could be due to the nature of rich augmentation strategies, making this dataset much more diverse and richer. Sathyavani et al. [50] discussed identifying nutrient deficiency in leaves through CNN and IoT devices and pointed out that ResNet outperforms VGG16 and DenseNet. However, their results have been influenced by the over-labeling and noise of the dataset. On the other hand, the proposed model exhibits good performance on precision, recall, and F1 score and shows a decrease in false positives. Our study outperforms theirs with 97% mAP while demonstrating how critical the augmentation of the model is for tackling generalization problems over different environments and lighting conditions. Ponce et al. [51] apply a CNNAHN classifier with an accuracy of 95.57%. Their approach surpasses traditional CNN approaches. However, it indicates that more extensive databases and better evaluations under various light conditions are warranted. In contrast, our model robustly performs in diverse scenarios. Kusanur and Chakravarthi [52], classified the given test samples into various nutrient deficiencies using VGG16 with SVM classifiers and Inception-V3 with random forest classifiers and reported improved classification accuracy. However, their approach is based on pre-trained models, and manual inspection efforts underline the limitations of traditional approaches. Our proposed work not only automates the detection process but also shows better generalization over multiple nutrient deficiencies with minimal manual intervention, reducing the precision agriculture workflow. Similarly, Hidayah et al. [53] compared the performance of several versions of YOLO models like YOLOv5 and Scaled-YOLOv4 against the YOLOv5, with an mAP reaching as high as 94.2%. Even though it is impressive, the authors have noticed some constraints related to the dataset size and variations in hardware use. Haque and Sohail [54] have focused on disease detection and used Inception V3 and MobileNet. They achieved accuracy rates of 96.11% for Inception V3 and 93.74% for MobileNet. In contrast, Elaraby et al. [55] obtained an accuracy of 98.83% for multi-crop disease classification using the PlantVillage dataset. Pan et al. [56], whose model CDDLite-YOLO achieved an mAP of 90.6%. Additionally, Rai and Pahuja [57] proposed work based on a DCNN for classifying cotton plant diseases and obtained an accuracy of 97.98%. Patel et al. [58] used CenterNet to distinguish between brinjal leaves and weeds, which achieved an 88% accuracy and a mean IoU of 86%. While their model was effective for specific applications, the proposed model

in this study surpasses these results by achieving a higher degree of accuracy in a more complex task. Liu and Zhao [59] reported an mAP of 88.84% using an enhanced YOLOv7-obb model for eggplant recognition. Tran et al. [60] obtained an accuracy of 97.2% in the identification of calcium deficiency stages with ResNet50. The results obtained with YOLOv7 were at a low accuracy of 85.8%. Our proposed approach significantly outperformed the given results. Li et al. [61] proposed a method for cotton pest and disease identification based on the CFNet-VoV-GCSP-LSKNet-YOLOv8s and achieved a Precision at 89.9%, Recall Rate (R) at 90.7%, and Mean Average Precision (mAP@0.5) at 93.7%. Nazeer et al. [62] detected Cotton Leaf Curl Disease (CLCuD) at an accuracy rate of 99% when tested against a proprietary dataset. This study deals only with detecting Cotton Leaf Curl Disease. Gao et al. [63] employed transfer learning and YOLOv8 to further improve cotton disease detection, with a development accuracy in both cotton pest and cotton disease detection set at 94%. Their models showed limitations in dataset scope and precision, which the k-fold cross-validation and the larger dataset in this study successfully address. Shahid et al. [64] used the GoogLeNet to get an accuracy of 93.4% and an F1-score of 95% when trained on CWT-extracted features, with AlexNet and Inceptionv3 scoring 93.4% and 91.8%, respectively. The key aspect of the study is the utilization of YOLOv8 model with the proposed augmentation scheme for the detection of disease as well as nutrient deficiency in the bitter guard leaves. This paper has developed a very highly accurate and reliable model in the detection of diseases and nutrient deficiency in bitter guards, and it outperforms many existing models in terms of precision, recall, and overall accuracy. The proposed study focused on detecting nutrient deficiencies and did not address overlapping stressors like drought explicitly. In practical settings, drought-induced chlorosis can mimic nitrogen or potassium deficiencies. Integrating additional environmental data such as soil moisture levels or using multi-modal imaging techniques (infrared for moisture content) could resolve this ambiguity [65]. Future extensions will consider these enhancements, combining visual and sensor-based diagnostics for more comprehensive plant health monitoring.

## XV. CONCLUSION AND FUTURE PERSPECTIVES

In this work, we proposed a DL-based solution for disease and nutritional deficiency detection of bitter guard leaves using the improved YOLOv8 model with layered augmentation. The model achieves an excellent mAP of 97 at an Intersection over union threshold of 0.50 and 94.5 across IoU thresholds ranging from 0.50 to 0.95. Precision reaches 91.8%, meaning maximum of predicted positive instances correspond to true positives, while recall is 90.2%, demonstrating that the model successfully identifies almost all true positives. The F1 score of 94.7 further underscores the balanced performance between Precision and Recall. This was an achievement that stood out more than the performances of models in a number of key prior studies. Our approach offered a valuable tool for

early, accurate intervention, and it has enhanced crop health, improved agricultural productivity if properly used, and helped overcome common problems such as false positives, variability in the dataset, and environmental generalization. In future studies, efforts will be made to adapt the proposed model in the real world, and improvements in the model will be made so as to extend its applicability across different domains and situations. The experiments were conducted on bitter gourd leaves, yet the method presented here, in principle, can be taken further to other crops and nutrient deficiencies. The same model, therefore, could be used to carry out future research on other crops, such as cereals, legumes, fruits, etc., for a general understanding of how applicable and effective such models are in diverse agricultural contexts. This bigger set, with higher variability, such as lighting, humidity, and soil quality, will make the model much more resilient and adaptable. In smart farming, the detection system can be further integrated with the real-time monitoring system using drones or IoT devices to enhance its capability to detect disease and nutrient deficiency promptly in large-scale farm settings and make timely interventions to minimize losses. In this respect, multi-modal learning might be implemented while incorporating visual data with additional data sources, such as soil tests or climate data, for increased precision and contextual relevance of predictions. Finally, the model refinement process, so that the resource-constrained device, like a cell phone or an edge device, can handle it, makes it much more attractive to use with those smallholder farmers who do not have high-end computing resources. Development of lightweight real-time applications will empower farmers and make it easier to diagnose the deficiency of crops on-site, whereas greater independence would walk hand in hand with more sustainable farming practices.

#### XVI. DECLARATION OF COMPETING INTEREST

Authors declare no Conflicts of interest/competing interests.

#### XVII. ACKNOWLEDGEMENTS

Work on plant abiotic stress tolerance in SSG laboratory was partially supported by University Grants Commission (UGC), Science and Engineering Research Board (SERB), Council of Scientific & Industrial Research (CSIR), Govt. of India. RG, SSG also acknowledges partial support from DBT-BUILDER grant (No. BT/INF/22/SP43043/2021). SSG also acknowledges partial support from MDU Post Seed Grant (No. DRD/25/1076-81). We sincerely apologize to our contemporaries whose work could not be discussed in this article due to space restrictions.

#### REFERENCES