# IT1244 PROJECT REPORT: RICE LEAF DISEASE DATASET

**Team 40:** Vu Minh Duc, Nguyen Tuan Dung, Do Trong Phuoc Nguyen, Le Ha Dang

## Abstract

Rice is one of the most widely consumed foods in the world. Many recognized diseases can destroy rice crops and hamper rice production, leading to many socioeconomic consequences. This report will focus on the application of AI to detect rice diseases using four main models: Artificial Neural Network, Convolutional Neural Network, Transfer Learning MobileNet and Finetuning YOLOv8. To ensure the robustness of the data, the given dataset was subjected to pretraining schemes and partitioned into a training set and a validation set, with different metrics used to compare the efficiency between models. Our results showed that YOLOv8 attained the highest score in all metrics while ANN and CNN did not perform well. We therefore concluded that Finetuning YOLOv8 is the most effective model for classifying rice diseases among our sample. With that said, more research can help further optimize this model to better achieve our task.

## Introduction

Rice plays a critical role in the social, cultural, and economical lives of many people on Earth, responsible for feeding over half of the world's population (Edmond 2023) and is therefore a key factor in efforts to reduce poverty and uphold sustainable development in many regions. As a result of rice's integral position within our society, scientific research has been invested in maximizing harvest and treating rice diseases. Rice diseases can reduce yields by up to 80% and lead to significant consequences: lowering agricultural earnings, decreasing food availability, and raising consumer costs (Shew et al., 2019). Thus, maintaining sustainable rice production and food security for millions of people worldwide depends on the efficient management of rice leaf diseases (Damalas and Koutroubas, 2016).

Normal methods of identifying rice diseases by hand are usually labor-intensive and require further guidance from specialists, resulting in its inaccessibility and inefficiency to many low-income farmers. In contrast, machine learning and deep learning algorithms enable accurate, timely, and convenient identification and classification of crop diseases, thereby simultaneously addressing the accessibility issue and enhancing agricultural output and quality.

Research investigating the application of AI into rice-leaf diseases classification shows varying results based on the approaches used. Previous report using CNN, VGG-16, VGG-19, Residual Network (RESNET), and Xception discovered that the 5-layer convolutional network performed the best at identifying rice leaves, but all models were inefficient (Tejaswini et al. 2022). Other reports using pre-trained VGG-16 model and Transfer Learning have shown increased accuracy of detection (Ghosal and Sarkar 2020). Additionally, other pre-trained models that have been incorporated into Transfer Learning are: InceptionResNetV2 (Krishnamoorthy et al 2021), Inceptionv3 and Resnet152v2 (Mavaddat et al 2023). Generally, the performance of these models were inconclusive between these studies. We also noticed that not many studies have been done on MobileNet and YoloV8 and compare their performance with traditional CNN models.

In this report, we are using four different models: basic Artificial Neural Network (ANN), basic Covolutional Neural Network (CNN), Transfer Learning Keras MobileNet, and Finetuning YoloV8 to train and validate the dataset, using accuracy, precision, recall, AUC, F1 score value and confusion matrix to evaluate between the models.

From our research, we obtain some preliminary weaknesses and advantages of the examined models. Firstly, ANN is found unable to effectively process high-dimensional data, including spatial features (Gunawan et al 2021). Secondly, while basic CNN can recognize and categorize plant illnesses with a higher accuracy than ANN, there are still signs of overfitting and unfounded prediction. Thirdly, MobileNet, with its versatility and lightness, can be integrated into a wide range of applications. Nonetheless, although MobileNet is pre-trained using large-scale models, its efficiency still depends on the quality of the actual dataset. Lastly, Finetuning YoloV8 was used because of its high accuracy and precision in recognizing objects of varying sizes and orientations. However, YoloV8 may struggle with detection of small objects in images and clustered scenes.

## Descriptive Statistics

The dataset includes 4 labels – Bacterial Leaf Blight (138 images), Brown Spot (111), False Smut (93) and Healthy (234). Overall, while the data was imbalanced due to the Healthy category, the other labels contained relatively similar amounts of data.

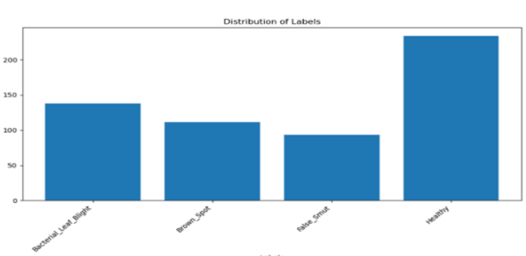


**Figure 1**. Bar Chart showing the distribution of 4 labels

The original dataset was also split 80/20 into training data and validation data prior to training. For all models, several preprocessing schemes are employed to introduce variants of the original images, enlarging the input size of the training data to decrease overfitting probability and enhance generalizing ability. Generally, these schemes are rescaling, shearing, zooming, and flipping (horizontally and vertically). Other additional schemes, such as color grading and "occluding" (omission of rectangular parts of the images), are introduced in model 4 as built-in pretraining schemes of YOLOv8.

**Figure 2.** Original image and processed image

## Model Training and Evaluation

To maximize the efficiency of the model considering our dataset and task, we utilized optimizer "adam" and loss function "categorical_crossentropy." "Adam" was chosen for being computationally light, fast convergent and for its ability to overcome local minima, thus improving our models' effectiveness. "Categorical crossentropy," on the other hand, is suitable for our classification task because of the few numbers of labels that can be one-hot encoded to reduce computational complexity.

Our models are evaluated using the following matrices: loss, recall, precision, accuracy, F1 score, and AUC (Area Under Curve) (area under the ROC curve), which are plotted against the number of epoch and compared between the training and validation datasets. Apart from the typical metrices, F1 score is introduced to account for the class imbalance in our input data as it is sensitive to minority classes. Lastly, confusion matrices are further used to examine the model's success and failure regarding specific labels.

Early stopping based on validation loss was also incorporated to reduce resources used and prevent overfitting. Validation loss was chosen since it is more sensitive to overfitting and is a clear signal for the model's performance and learning capacity for the validation set. Furthermore, this metric is relatively smooth and continuous, especially within the earlier epochs while other metrics tend to fluctuate more. If validation loss does not improve without 10 epochs, training will be halted, and the best weight attained from the previous epochs will be chosen.

This study explores the effectiveness of four distinct models: Simple Artificial Neural Network (ANN), Convolutional Neural Network (CNN), Transfer Learning using Keras MobileNet, and Fine-tuning YOLOv8. Each model is evaluated for its ability to accurately detect rice leaf diseases, contributing to the advancement of precision agriculture and crop management practices.

All four models take an input shape of (224, 224, 3) which indicates images with a height and width of 224 pixels and 3 channels (RGB).

## Result and Discussion

### Model 1: Simple Artificial Neural Network

We first start with a simple Artificial Neural Network comprising of only dense layers with no convolutional layers. The input image is flattened and immediately fed into the dense layers.

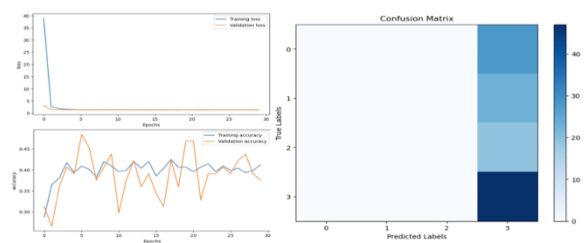


**Figure 3.** Loss, Accuracy and Confusion matrix of Model 1

The model predicts all data point as label 3, suggesting that it cannot learn any features from the data. This is because the lack of convolutional layers prevent the model from capturing the spatial information from the image.

### Model 2: Convolutional Neural Network

Convolutional Neural Networks (CNNs) are a class of deep learning models designed for processing grid-like, multidimensional data like images or videos. A simple CNN comprised of convolutional, pooling, and fully connected layers, allows it to automatically derive hierarchical representations of features from raw input. In particular, convolutional layers extract spatial patterns and features by sliding the kernels across the input, with pooling layers reducing spatial dimensions. Fully connected layers integrate these features to make predictions by transforming and keeping only the most significant value of the features.

In this section, we will try with a simple CNN model and try to hyperparameter-tune it to achieve better performance.

**Model 2A**: Simple CNN

We start with a simple CNN with only 2 convolutional layers and a limited number of nodes in each. We use ReLU

(Rectified Linear Unit) as the activation functions except for the last dense layer to transform the result into probability.

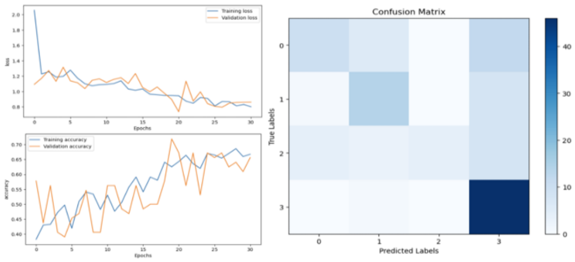


**Figure 4.** Loss, Accuracy and Confusion matrix of Model 2A

We observe a significant improvement from Model 1 as the CNN model has been able to capture the spatial information from the image and given more informed decision about the class, achieving an accuracy of 63%. However, the performance is still unstable, and the confusion matrix still show the tendency of predicting label 3. This indicates that the model may be underfitting and too simple to capture all the valuable information from the data.

This prompts us to experiment with more complex models, firstly with hyperparameter tuning.

**Model 2B**: Hyperparameter tuning of Model 2A

To obtain the optimal hyperparameters, we use "kerastuner" as the framework to automatically find and choose the most feasible model. We target validation accuracy and tune the number of nodes in each layer and the dropout rate.

Our chosen tuning algorithm is Hyperband. Hyperband is used to efficiently search the hyperparameter space of machine learning models, combining random search and early stopping to find the best set of hyperparameters within a limited computational budget. Hyperband begins by training a randomly sampled set of hyperparameter configurations for limited epochs. Potential configurations will be allocated more resources and testing while low performance models will be discarded. By focusing computational resources on promising configurations, Hyperband is particularly useful when training can be computationally expensive.

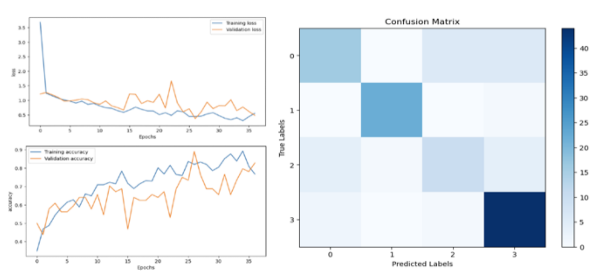


**Figure 5.** Loss, Accuracy and Confusion matrix of Model 2B

As expected, the result from the tuning process shows that a bigger model with an increased number of nodes in each layer and reduced dropout rate is more effective for our task. This new Model 2B has an improved validation accuracy of 78% and training accuracy of up to 90%. However, the validation plot still fluctuates largely, with big differences between training and validation metrics, suggesting overfitting.

**Model 2C**: Complex CNN Model

For model 2C, we try to tune the model 2A by adding one more convolutional layer before flattening. We also use another training scheme, "reduce learning rate on plateau", on validation loss to help regulate the learning rate when the model does not improve on this metric.

The result is that this model seems to perform worse than the simple CNN model. An explanation for this result is that a bigger model requires more parameters, while our training dataset is limited to only about 400 images. Since the model does not have enough data to train, overfitting occurs.

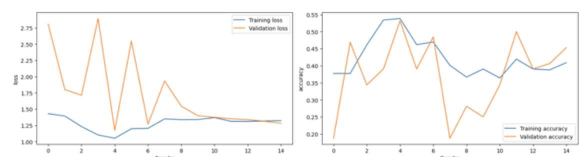


**Figure 6.** Loss, Accuracy and Confusion matrix of Model 2C

## Model 3: Transfer Learning with Keras MobileNet

Keras MobileNet is a lightweight convolutional neural network (CNN) architecture designed for mobile and embedded devices, supporting fast deep learning models suitable for applications where computational resources are limited, such as our problem.

In this model, we utilized pretrained MobileNet on ImageNet as the base model, without using the top layer. The pretrained model is used as a fixed feature extractor to leverage the knowledge gained from a large general dataset like ImageNet. We then added some top layers and trained only these layers on the leaf diseases dataset. This is particularly useful as we only have a small data set.

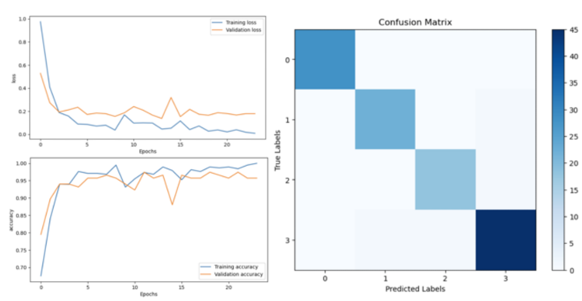


**Figure 7.** Loss, Accuracy and Confusion matrix of Model 3

Overall, this Model 3 outperformed the previous two models with up to 96% accuracy, precision, recall, and F1 score. From the loss and accuracy graphs, we see that the model is

`

relatively stable, and the performance stays consistently high. This indicates that this model generalizes well to new data in the validation set and thus suitable for our task.

### Model 4: Finetuning with YOLOv8

YOLOv8, short for "You Only Look Once version 8," is well known for its lightning-fast speed and accuracy, uses a single neural network to predict bounding boxes and class probabilities for several objects in an image at the same time. Its architecture leverages innovative techniques in computer vision, such as feature pyramids and multi-scale predictions, enabling it to detect objects with unparalleled efficiency. With its robust performance across various scenarios, YOLOv8 has become a cornerstone in the field of object detection, offering immense potential for applications ranging from autonomous vehicles to surveillance systems.

In our final model, we fine-tuned YOLOv8 on our rice leaf dataset. Different from the previous transfer learning model, finetuning involves retraining the network using our target data, allowing the model to learn the distinctive features and characteristics of the objects of interest.

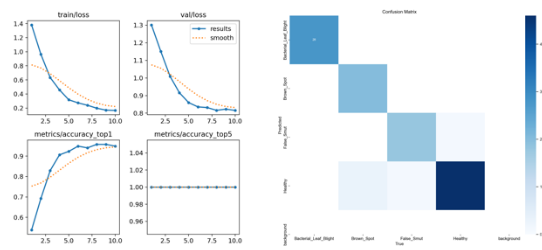


**Figure 9.** Loss, Accuracy and Confusion matrix of Model 4

The YOLOv8 model has an impressive performance with accuracy vary from 0.95 to 1.00 on validation batches with a few mislabeled images. With our limited data, this has shown the power of pretrained models.

## Conclusion

Through our project, we provide a critical comparison between four distinct types of neural network models in their ability to classify rice-leaf diseases. Our primary conclusion is that the Finetuning YOLOv8 model is the most capable for the task due to its apex performance in all the metrics. Furthermore, we argue that ANN is strictly unsuitable for our project due to its inability to process spatial data. Our secondary conclusion is that CNN models can be optimized by varying hyperparameters and number of convolutional layers. However, having more convolutional layers does not necessarily translate to a better model, especially in our project as there might not be enough features for the model to learn from. Finally, we support the use of pre-trained models to enrich our solution as these models are often trained using an extensive amount of general data and can compensate very well if the specific task has a limited dataset.

## Reference

Damalas, C., & Koutroubas, S. (2016). Farmers' exposure to pesticides: Toxicity types and ways of prevention. *Toxics*, *4*(1), 1. https://doi.org/10.3390/toxics4010001

Edmond, C. (2023, June 6). *Rice is both a victim and a villain in terms of the climate crisis. Here's why*. World Economic Forum. https://www.weforum.org/agenda/2023/06/rice-climate-crisis-food-security/

Ghosal, S., & Sarkar, K. (2020). Rice leaf diseases classification using CNN with transfer learning. *2020 IEEE Calcutta Conference (CALCON)*. https://doi.org/10.1109/calcon49167.2020.9106423

Mavaddat, M., Naderan, M., & Alavi, S. E. (2023). Classification of rice leaf diseases using CNN-based pre-trained models and transfer learning. *2023 6th International Conference on Pattern Recognition and Image Analysis (IPRIA)*. https://doi.org/10.1109/ipria59240.2023.10147178

N, K., Narasimha Prasad, L. V., Pavan Kumar, C. S., Subedi, B., Abraha, H. B., & V E, S. (2021). Rice leaf diseases prediction using deep neural networks with transfer learning. *Environmental Research*, *198*, 111275. https://doi.org/10.1016/j.envres.2021.111275

Tejaswini, P., Singh, P., Ramchandani, M., Rathore, Y. K., & Janghel, R. R. (2022). Rice leaf disease classification using CNN. *IOP Conference Series: Earth and Environmental Science*, *1032*(1), 012017. https://doi.org/10.1088/1755-1315/1032/1/012017

`

**Appendix:** Performance plot

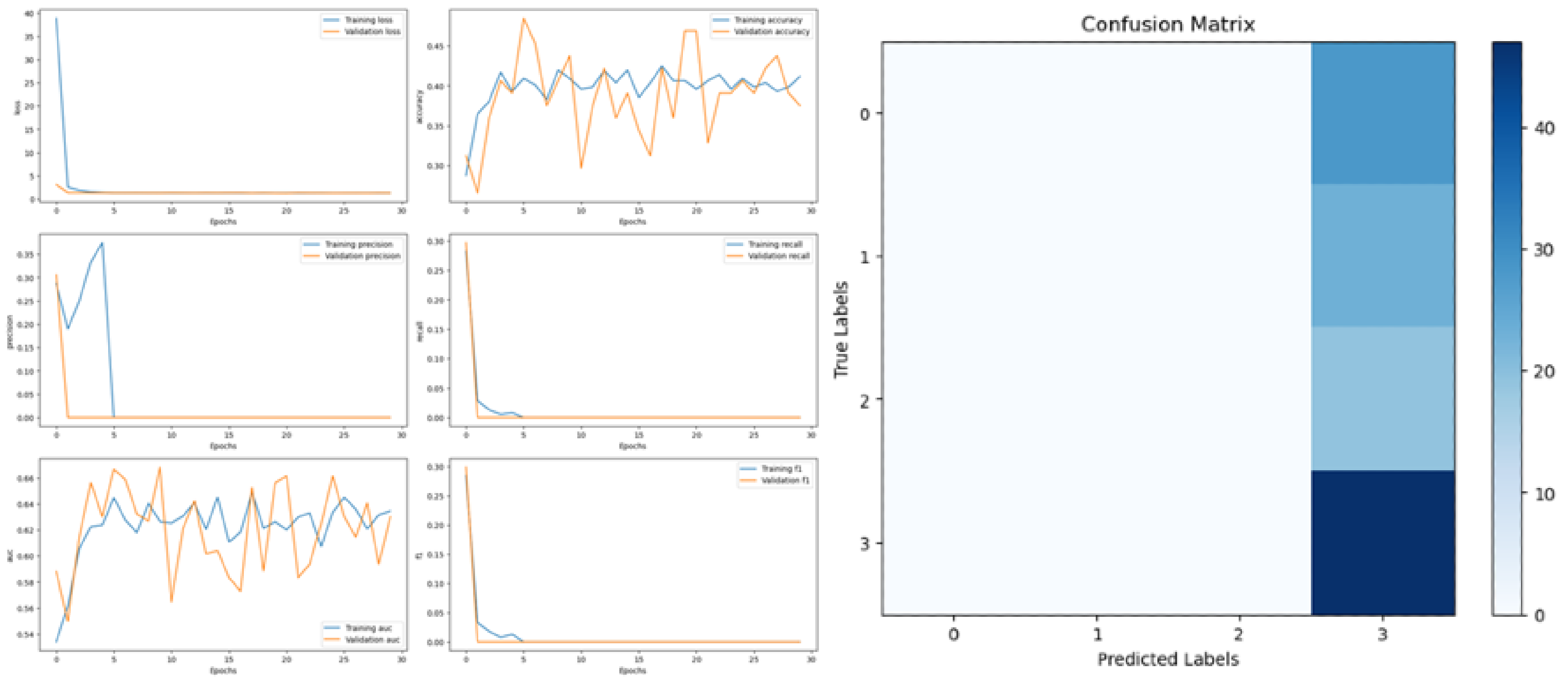


**Figure 10.** Model 1 performance

`

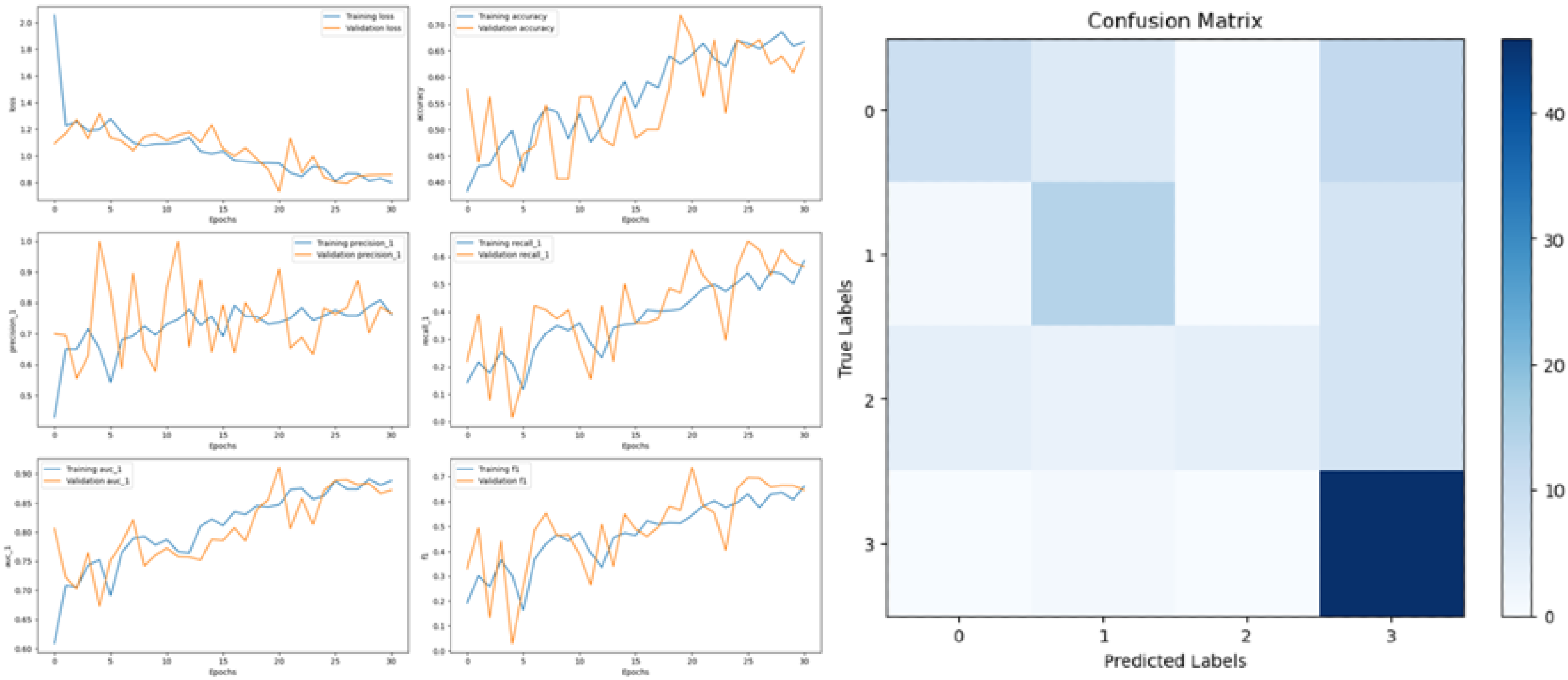


**Figure 11.** Model 2A performance

`

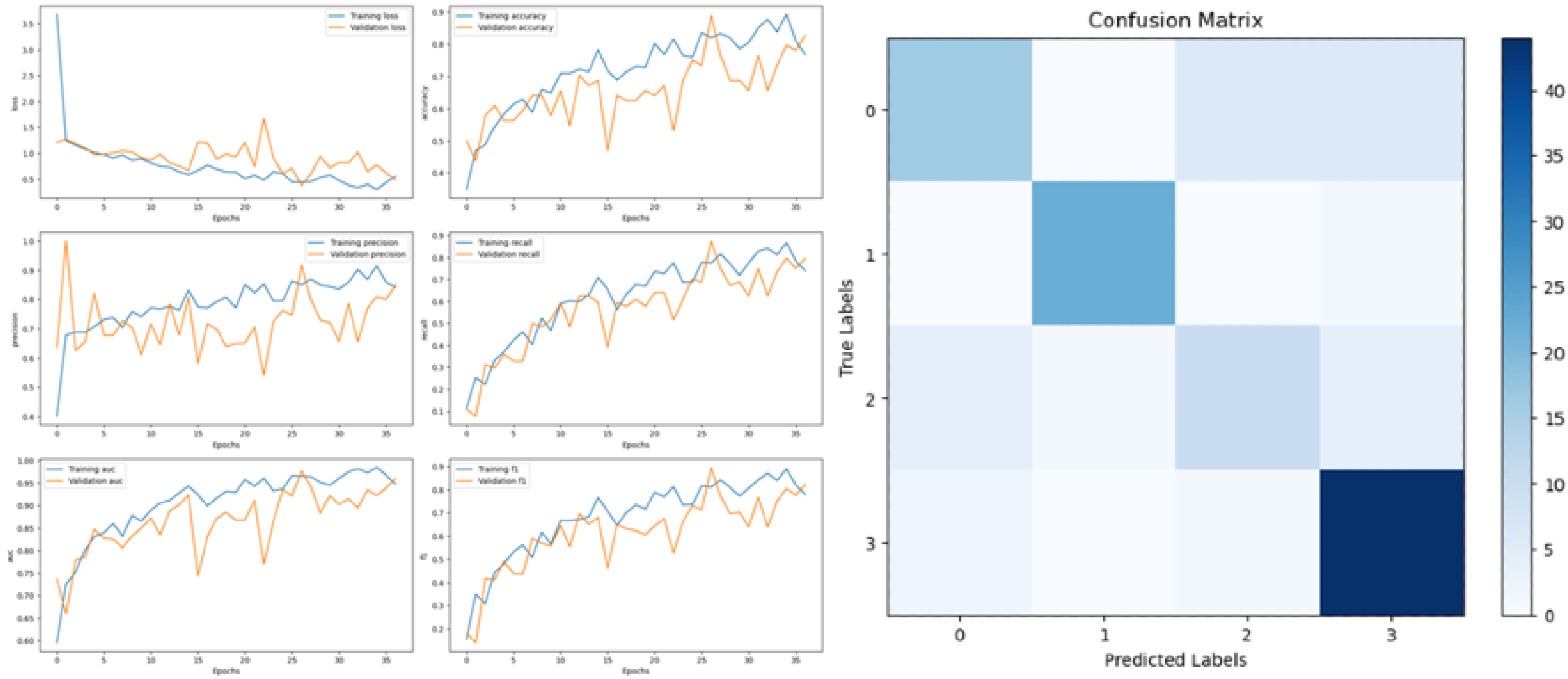


**Figure 12.** Model 2B performance

`

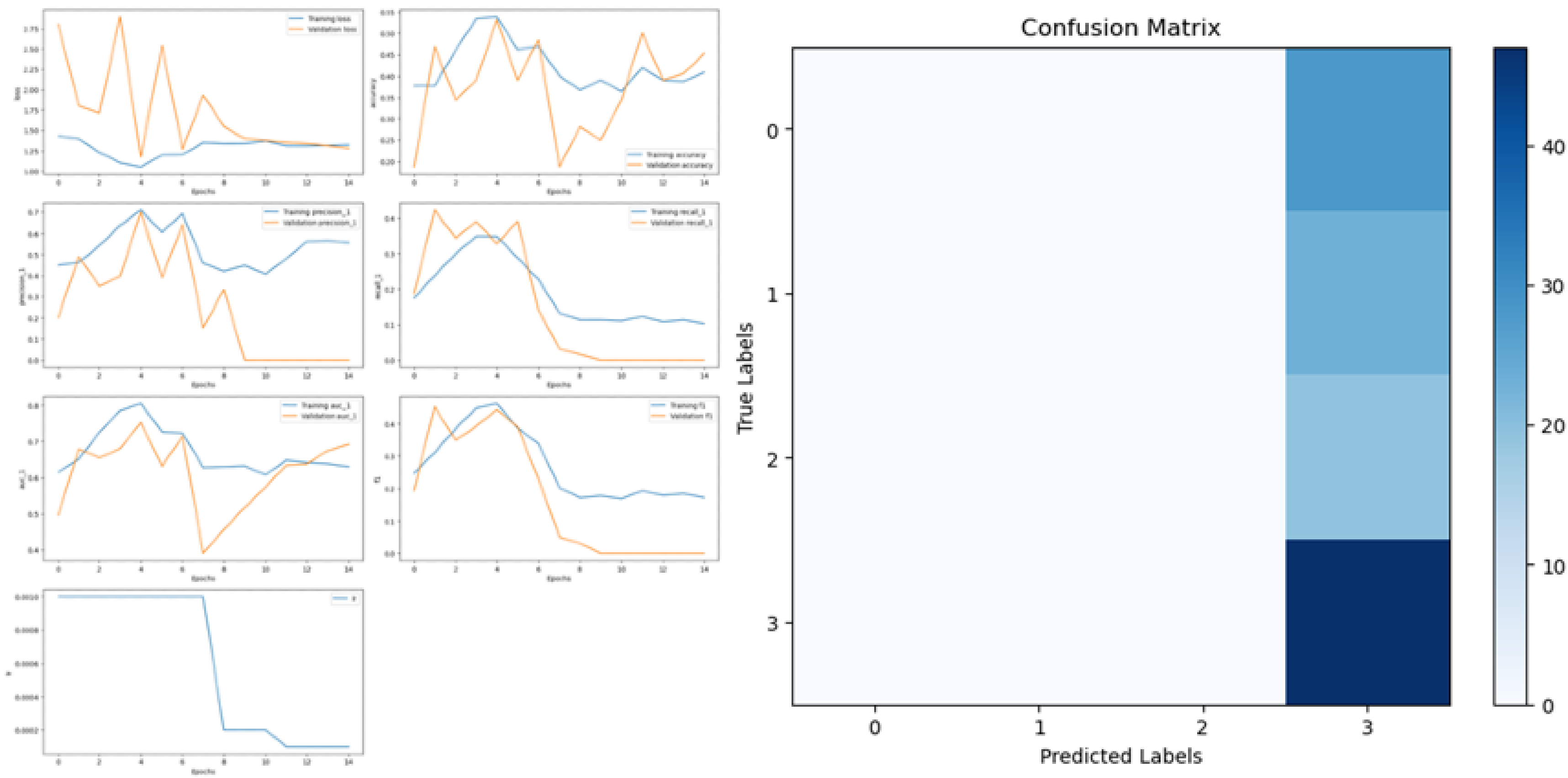


**Figure 13.** Model 2C performance

`

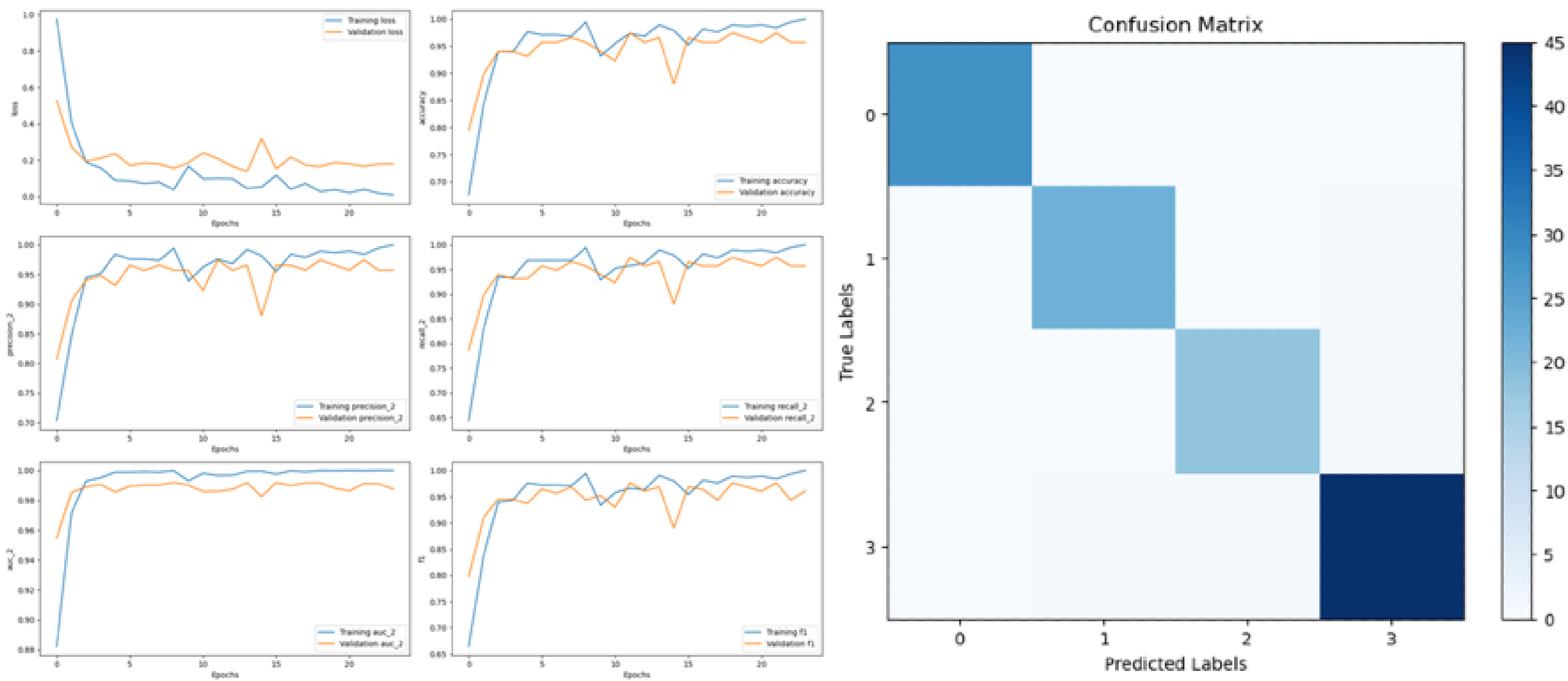


**Figure 14.** Model 3 performance

`

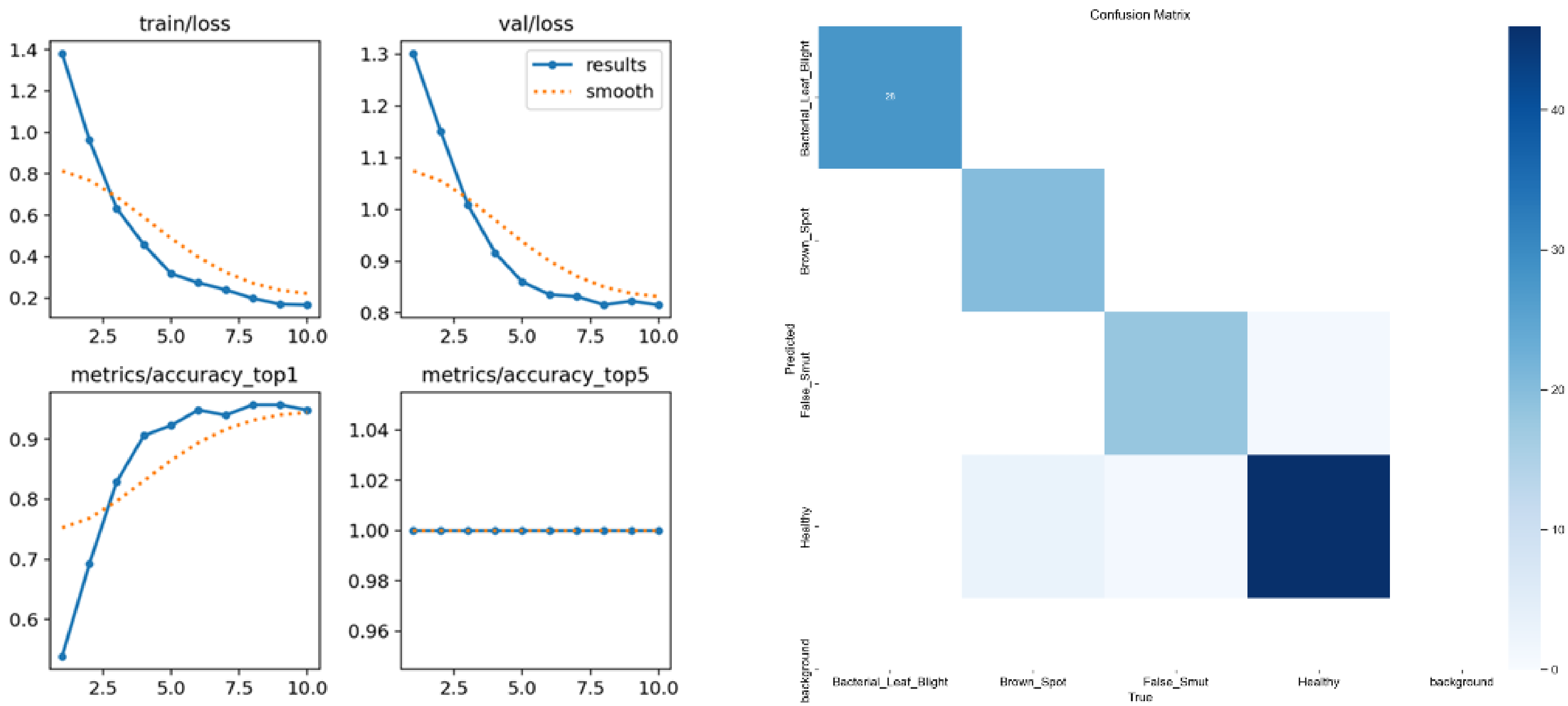


**Figure 15.** YOLOv8 performance

`