International Journal of Academic Information Systems Research (IJAISR)
ISSN: 2643-9026
Vol. 9 Issue 6 June - 2025, Pages: 12-20

Deep Learning For Grapevine Disease Detection

Salah-Aldin .S. Aldaya and Samy S. Abu-Naser

Department of Information Technology,
Faculty of Engineering & Information Technology,
Al-Azhar University - Gaza, Palestine

***Abstract**: The global cultivation of grapes reaches approximately 77.8 million tons annually, according to the International Organization of Vine and Wine. While grapes remain a vital agricultural commodity and dietary staple worldwide, their production faces serious threats from common diseases like black rot, Esca, and leaf blight. Current disease detection methods in modern vineyards primarily depend on manual visual inspection, a practice that often delays diagnosis and leads to reduced yields and compromised fruit quality. The integration of automated detection methods, particularly those based on machine learning, is crucial for promoting sustainable viticulture. This study leverages the power of deep learning, a subfield of machine learning particularly adept at interpreting image data. Specifically, we implemented Convolutional Neural Networks (CNNs) to classify images of grapevine leaves as either healthy or diseased. The models employed included a custom-built baseline CNN and a range of advanced transfer learning models: DenseNet121, EfficientNetB7, MobileNetV2, ResNet50, and VGG16. While we initially hypothesized that ResNet50 would yield the highest accuracy due to its deep architecture, experimental results revealed that EfficientNetB7 outperformed all other models. To further enhance classification performance, we constructed a max-voting ensemble using the top three performing models. This ensemble approach outmatched the performance of individual models. The final model was deployed via a web-based interface, enabling vineyard professionals and growers to detect black rot, Esca, leaf blight, or confirm leaf health by uploading images captured in real-world vineyard environments.*

INTRODUCTION

According to the International Organization of Vine and Wine, global grape production reaches 77.8 million tons annually. Grapes have played an integral role in human consumption for centuries, used in various forms such as fresh fruit, wine, and raisins. However, grapevine production is increasingly jeopardized by diseases like black rot, Esca, and leaf blight. These infections not only diminish crop yields but also compel farmers to rely heavily on costly fungicides.

Black rot, caused by the fungus *Guignardia bidwellii*, thrives in warm and humid climates, rendering grapes unmarketable and earning the title “Achilles’ heel” of grape cultivation in regions like the Middle East. Esca, a destructive trunk disease often linked to *Phaeoacremonium aleophilum*, typically manifests during the summer months and causes significant economic losses — in France alone, up to 13% of vineyards are affected annually, amounting to over 1 billion euros in damages. Leaf blight, triggered by *Xylophilus ampelinus*, has been documented to reduce yields by more than 70% in affected areas.

Traditionally, vineyard diseases are managed through cultural practices and fungicide applications. Black rot can be controlled by enhancing ventilation and sanitation within vineyards. Esca, lacking a definitive chemical cure, necessitates preventive strategies, while leaf blight is typically managed via fungicide treatments and moisture regulation. Early and precise disease identification is essential for limiting fungicide use and minimizing crop losses.

Conventional methods of disease detection rely on expert visual inspection, a process limited by subjectivity and the availability of specialists. This limitation underscores the need for innovative approaches such as machine learning, particularly deep learning with Convolutional Neural Networks (CNNs), which offer promising potential for automated disease diagnosis.

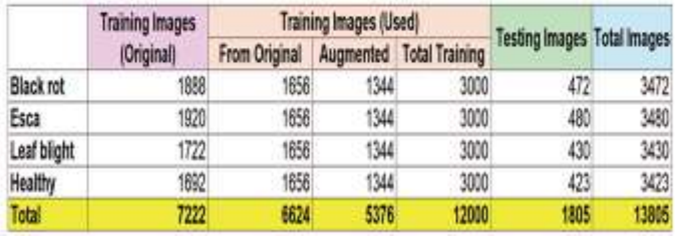
Machine learning, a subset of artificial intelligence, empowers systems to learn from experience and improve their performance autonomously. Deep learning, a specialized branch of machine learning, utilizes neural networks inspired by biological neural structures to extract patterns and make predictions from complex datasets. CNNs, in particular, have proven highly effective in analyzing visual data.

Dataset Description and Table 1

In contrast to traditional Support Vector Machine (SVM) approaches, Convolutional Neural Networks (CNNs) are capable of training on larger datasets, thereby enhancing their ability to recognize diverse disease patterns with higher accuracy. This approach aligns with the goals of precision agriculture, which advocates for the use of technology to optimize crop health and yield.

In this study, we employed a grapevine leaf image dataset obtained from kaggle.com, which included four distinct classes: black rot, Esca, leaf blight, and healthy. The dataset originally contained unequal numbers of images per class. To maintain balance during training, we selected an equal number of images for each class and applied data augmentation to generate a consistent number of training samples across all categories. Each class ultimately contained an equal number of original and augmented images to ensure uniform training and improve model generalization.

Table 1. Number of Images per Disease Class in the Dataset



	Training Images (Original)	Training Images (Used)			Testing Images	Total Images
		From Original	Augmented	Total Training		
Black rot	1888	1656	1344	3000	472	3472
Esca	1920	1656	1344	3000	480	3480
Leaf blight	1722	1656	1344	3000	430	3430
Healthy	1692	1656	1344	3000	423	3423
Total	7222	6624	5376	12000	1805	13805

These balanced datasets facilitated fair comparisons among models during training and evaluation. Data augmentation techniques such as zooming, rotating, altering brightness, and shearing were used to simulate a wide range of real-world image conditions. These transformations helped improve the model's robustness to variations in leaf appearance due to environmental conditions or image quality.

In agricultural research, Support Vector Machines (SVMs) have been used to detect grapevine diseases. Although effective in distinguishing healthy from diseased leaves, these models were constrained to binary classification and trained on a relatively small dataset of 250 diseased and 400 total leaf images, limiting their practical application for farmers needing to differentiate between multiple diseases.

In contrast, CNNs can be trained on large-scale image datasets, enabling multi-class classification with higher accuracy. This methodology aligns with the principles of precision agriculture, which leverages technology to enhance crop monitoring and productivity. Furthermore, transfer learning — the process of adapting pre-trained networks to new tasks — has expanded the capabilities of deep learning in agriculture. State-of-the-art models such as DenseNet, EfficientNet, MobileNet, ResNet, and VGG, originally designed for general image recognition, have been successfully repurposed for disease detection in crops. The max-voting ensemble approach, which aggregates predictions from multiple models to improve accuracy, has also emerged as a powerful technique in this domain.

RESULTS

The grapevine dataset comprised images of leaves classified into four categories: black rot, Esca, leaf blight, and healthy (Figure 1). To enhance model robustness and prevent overfitting, various data augmentation techniques were applied. These included adjustments in zoom, rotation, brightness, and shearing, which allowed the models to generalize better to unseen, real-world images.

Our initial evaluation focused on comparing the classification accuracies of the different CNN architectures. Among all models tested, EfficientNetB7, ResNet50, and the baseline CNN demonstrated the highest accuracy, achieving 99.6%, 99.2%, and 99.1%, respectively (Figure 2A). These three models were selected to form the foundation of a max-voting ensemble. The remaining models—VGG16, DenseNet121, and MobileNetV2—achieved accuracies of 98.3%, 96.7%, and 94.3%, respectively.

To gain further insights into the models' behavior, we constructed confusion matrices for each CNN model (Figure 2B). These matrices revealed that the most common classification errors occurred between black rot and Esca images. Notably, models such as DenseNet121, EfficientNetB7, ResNet50, and VGG16 frequently misclassified these two classes, likely due to their visual similarity in spot patterns and shapes.

To understand the source of misclassifications, we analyzed error visualizations from the EfficientNetB7 model. The analysis showed that some leaf images displayed features that were nearly indistinguishable between two disease classes, contributing to the difficulty in accurate classification.

We also evaluated the learning behavior of the models by plotting training and validation accuracy curves (Figure 3A). These graphs indicated that the training accuracy was generally higher than the validation accuracy, which is expected. However, the baseline CNN exhibited notable fluctuations in its validation accuracy, suggesting potential overfitting during training.

Further, we analyzed the relationship between model weight size and validation accuracy by plotting their correlation (Figure 3B). The results indicated a strong positive correlation—models with larger weight files tended to achieve higher accuracy. This trend underscores the trade-off between model complexity and performance.

We then deployed the three top-performing models—EfficientNetB7, ResNet50, and baseline CNN—in a max-voting ensemble. This ensemble aggregated the predictions of each model and selected the final label based on the majority vote. The ensemble demonstrated outstanding performance with a classification accuracy of 99.8%. It misclassified only four images: two black rot leaves were predicted as Esca, and two Esca leaves were classified as black rot (Figure 4).

To assess the efficiency of the ensemble model, we recorded inference times with and without a GPU. With GPU acceleration, the average time to classify a single image was 12.17 milliseconds. Without a GPU, the time increased significantly to 383 milliseconds. This clearly highlights the role of hardware in improving the speed of deep learning inference tasks.

DISCUSSION

This study demonstrated that both EfficientNetB7 and ResNet50 are highly effective in the classification of grapevine leaf diseases. While we initially hypothesized that ResNet50 would outperform the other models due to its deep residual architecture, EfficientNetB7 ultimately proved to be the most accurate, achieving a classification accuracy of 99.6%. Its ability to scale efficiently and minimize parameter count contributed significantly to its superior performance across various image conditions.

The utilization of ensemble methods further elevated classification accuracy. The max-voting ensemble, combining EfficientNetB7, ResNet50, and the baseline CNN, achieved an impressive 99.8% accuracy—surpassing the results of each individual model. This ensemble approach leveraged the complementary strengths of each model, improving robustness and reducing individual model biases.

An analysis of the training and validation accuracy curves (Figure 3A) revealed expected trends. Training accuracies were generally higher than validation accuracies. However, the baseline CNN exhibited erratic validation accuracy, indicating susceptibility to overfitting. This instability was not observed in the transfer learning models, which showed consistent and stable learning behavior across epochs.

The discrepancy in validation accuracy between the baseline CNN and transfer learning models reflects differences in architectural robustness. The transfer learning models, pre-trained on large-scale image datasets and adapted to grapevine disease classification, showed superior generalization capabilities. These models were not only more consistent but also more reliable in maintaining accuracy during validation.

Data augmentation played a pivotal role in boosting performance, particularly for lower-accuracy models such as MobileNetV2, DenseNet121, and VGG16. These models exhibited improved generalization after being trained on augmented datasets, indicating that they initially struggled with overfitting. In contrast, top-performing models—EfficientNetB7, ResNet50, and the baseline CNN—showed only marginal improvements, highlighting their inherent robustness.

The relationship between model size and performance was also explored (Figure 3B). Our findings revealed a positive correlation between model weight size and accuracy, particularly among models with sizes below 150 megabytes. Beyond this threshold, however, gains in accuracy became marginal. This diminishing returns effect suggests that excessively large models may not yield proportional performance improvements and could be prone to overfitting or computational inefficiency.

These results u nderscore a critical balance between model complexity and deployability in real-world agricultural applications. While larger models offer increased accuracy, they may not be feasible for deployment in resource-constrained environments. Thus, selecting models that offer both high performance and computational efficiency is essential for practical field use.

Furthermore, inference speed is a significant factor in the real-world usability of deep learning models. With GPU support, the ensemble model achieved an average inference time of just 12.17 milliseconds per image. Without GPU acceleration, inference time rose to 383 milliseconds. This stark contrast emphasizes the need to optimize models for faster performance, especially in settings where high-end hardware is not available.

To address these constraints in future work, we propose two strategies: **model pruning** and **depthwise separable convolutions**. Model pruning involves removing redundant parameters to streamline the network without compromising accuracy, while depthwise separable convolutions decompose standard convolutions into more efficient operations, significantly reducing computational overhead. Although these methods may introduce trade-offs in accuracy, they present promising avenues for improving inference speed and enabling deployment on lower-end devices

In this study, we learned that EfficientNetB7 and ResNet50 worked well individually for detecting diseases in grapevine plants. We hypothesized that ResNet50 would be the most accurate as it allowed deep networks to train with high accuracy. While ResNet50 performed very well with 99.2% accuracy, EfficientNetB7 was even more accurate with 99.6% accuracy. EfficientNetB7, with its ability to train efficiently, easily adapted to a wide variety of datasets, and massively reduced the number of parameters in a model, outperforming all other models. Moreover, ensemble methods such as maxvoting enabled us to achieve 99.8% accuracy on testing data with better results than any single transfer learning CNN model.

From the training and validation accuracy curves (**Figure 3A**), we noticed that the training accuracy is higher than the validation accuracy. This made sense intuitively. However, we noticed that the baseline CNN's validation accuracy during training was inconsistent. This hinted that the model might have overfitted the training data. The rest of the line graphs (**Figure 3A**) had no issues.

The validation accuracy curve of the baseline CNN model highlighted that simpler model architectures could achieve high accuracy but might exhibit variability across epochs due to overfitting to the training dataset. In contrast, the five transfer learning CNN models, which were derived from the TensorFlow library (an open-source software library developed by the Google Brain team for machine learning and deep neural network research) and not from the baseline CNN model, demonstrated more consistent validation accuracies. These models were carefully modified to accommodate the grapevine disease detection dataset, ensuring their performance metrics are distinct and independent from those of the baseline CNN model. The robust architectures of these transfer learning models, which had been pre-trained on large and diverse datasets, contributed to their stability and reliability in validation accuracy, underscoring their effectiveness for this application.

In this study, data augmentation improved the performance of lower-accuracy models, such as MobileNetV2, DenseNet121, and VGG16, suggesting these models initially suffered from overfitting and benefited from the more varied training examples. Conversely, the higher-performing models – EfficientNetB7, ResNet50, and the baseline CNN – showed only a slight increase in accuracy with augmented

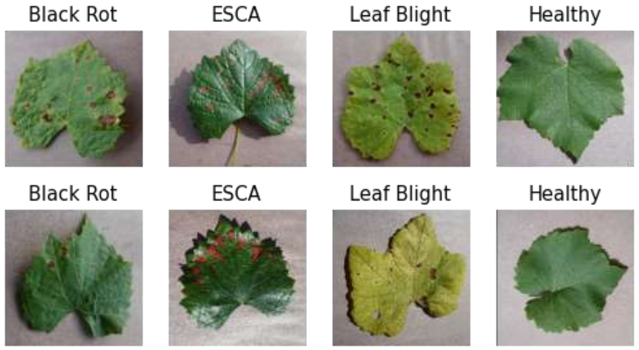
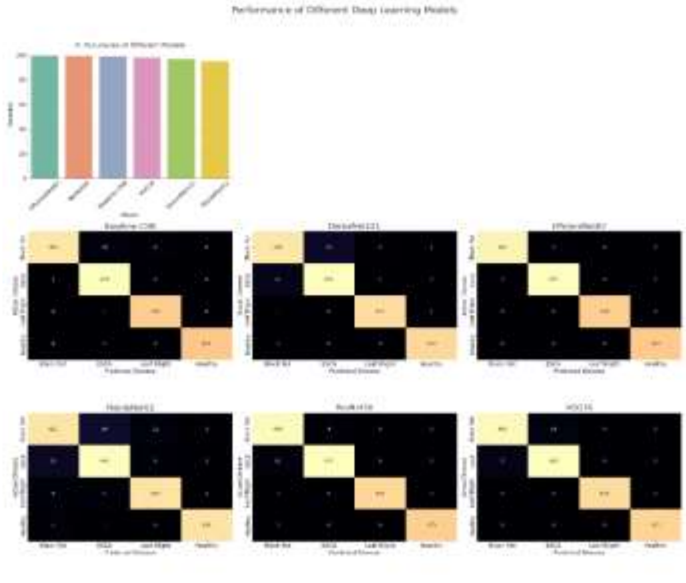


Figure 1: Examples of images showing grapevine leaves that are infected with black rot, Esca, leaf blight, or are healthy. There are 2 images for each class of grapevine leave diseases that are photographed on a plain neutral colored background, one at a time.

Data, reflecting their inherent robustness and effective generalization from the training set. These observations highlighted the role of data augmentation in boosting model performance, especially by mitigating overfitting in models that were less optimized.

The correlation between model size and accuracy (**Figure 3B**) presented a nuanced insight into the effectiveness of different models. The observed trend indicated that larger models generally achieve higher accuracy. We could attribute this correlation to several factors that impact a model's size, including the depth and number of layers, the complexity of the network architecture, and the volume of parameters and weights that the model must learn during training. Larger models like EfficientNetB7 and ResNet50, which had more extensive and intricate architectures, were capable of learning more complex features and patterns in the data, contributing to their higher accuracy rates.

However, the relationship between model size and accuracy was not linear. As indicated in our results, beyond a threshold of around 150 megabytes, the improvements



i **Figure 2: Prediction accuracies and confusion matrices of the models tested.**(A) Prediction accuracy on the grapevine leaves testingdata. The bar graph shows the accuracies of six different models: Baseline CNN, DenseNet121, EfficientNetB7, MobileNetV2, ResNet50, and VGG16. The accuracies ranged from 94.3% to 99.6%. The models are sorted in decreasing order based on their respective accuracy. (B) Six confusion matrices depicting errors of the models tested. For each confusion matrix, the x-axis has the predicted disease while the y-axis has the actual disease. The number of testing images that fall into a given category is written inside each square. The squares that are colored orange represent a large number of data points and black represent a small number of data points.

in accuracy became marginal (**Figure 3B**), which was particularly revealing. This plateau effect suggested that while more complex models with a greater number of parameters could capture nuanced patterns in data more effectively, there was a point beyond which additional complexity did not yield significant benefits. This could be due to overfitting, where the model became so well-tuned to the training data that it failed to generalize effectively to new, unseen data. It

also reflected the principle of diminishing returns in machine learning, where the cost (in terms of computational resources and time) of increasing model complexity did not always translate into proportional improvements in performance. Our findings highlighted that the relationship between model size and accuracy significantly affected the feasibility of deploying machine learning models in real-world agricultural contexts. Our research goals were to develop models

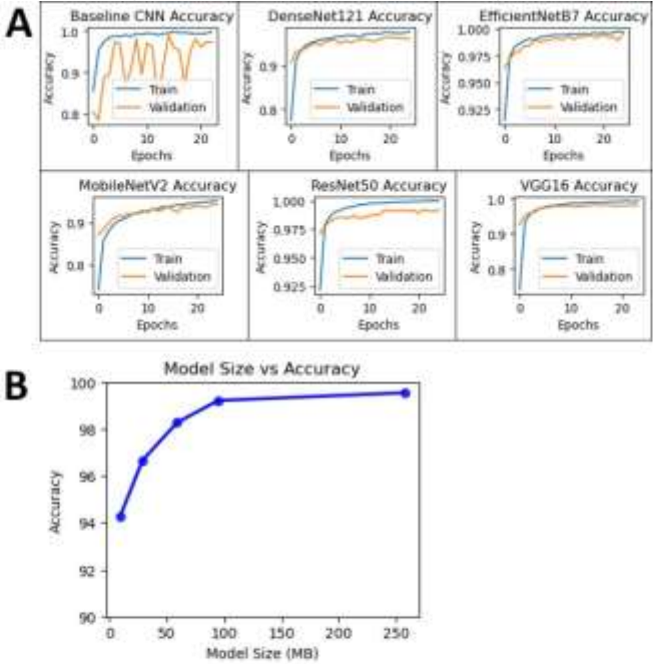
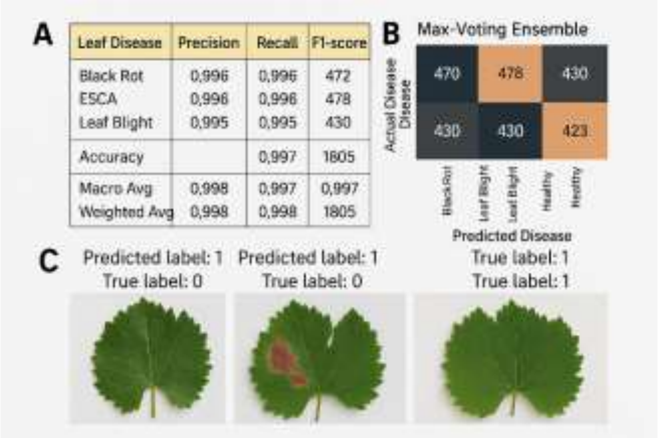


Figure 3: Training and validation accuracies of different models as



Leaf Disease	Precision	Recall	F1-score
Black Rot	0,996	0,996	472
ESCA	0,996	0,996	478
Leaf Blight	0,995	0,995	430
Accuracy		0,997	1805
Macro Avg	0,998	0,997	0,997
Weighted Avg	0,998	0,998	1805

Figure 4: Classification metrics, confusion matrix and error visualization for max-voting ensemble deep learning model. (A) Classification metrics for the max-voting ensemble including precision, recall, f1-score, accuracy, macro average, and weighted average. (B) A confusion matrix depicting the errors made by the max-voting ensemble. The x-axis has the predicted disease while the y-axis has the actual disease. The number of testing images that fall into a given category is written inside each square. These squares are colored orange representing a large number of data points and black representing a small number of data points. (C) All of the grapevine leaf images that were misclassified by the max-voting ensemble are depicted. The predicted label and true label are shown above the image. The numbers 0, 1, 2, and 3 refer to black rot, Esca, leaf blight, and healthy respectively. Out of the 4 misclassified images, 2 black rot-affected images were predicted as Esca, and another 2 Esca-affected images were classified as black rot

well as the relationship between model size and accuracy. (A) Six line graphs depicting the training and validation accuracies of all models tested. The x-axis represents the epoch number, and the y-axis represents the accuracy. The blue line represents the training accuracy, and the orange line represents the validation accuracy. (B) A line graph depicting the positive correlation between the size of the model's weights in megabytes and the accuracy (in %) on testing data.

that not only exhibited high accuracy but also maintained computational efficiency for ease of use in the field. This balance ensured that the models were sophisticated enough to detect subtle indicators of disease effectively yet streamlined enough to be deployed on platforms with limited computational power, which was often the case in agricultural environments. The findings from this study would inform the selection of future models, guiding a targeted approach that prioritized both precision and practicality for end-users such as vineyard operators.

In this study, it took the max-voting ensemble 12.17 milliseconds to predict a single image with a GPU. However, most farmers lack access to GPUs. Without one, it took the max-voting ensemble 383 milliseconds to predict a single image. In future, this time could be reduced to apply this technology effectively in rural areas with poor farmers. To enhance model inference time in future work, we plan to implement two key strategies. The first is model pruning, a process where redundant network parameters are identified and removed. This approach aims to streamline the model without sacrificing its original accuracy, by focusing on the most crucial aspects of the network. The second strategy involves the use of depthwise separable convolutions. This technique separates the spatial and depth convolutions, significantly reducing the computational load and thus speeding up the inference process. However, there is some uncertainty regarding how this change might affect the model's ability to accurately capture complex features, and consequently, its overall accuracy.

In summary, this research successfully demonstrated the superior performance of EfficientNetB7 and ResNet50 in detecting grapevine diseases, with data augmentation proving particularly beneficial for models susceptible to overfitting. While ensembling methods like max-voting significantly enhanced accuracy, the study also revealed a diminishing returns effect in the relationship between model size and accuracy, guiding practical model deployment in agriculture. Future work aims to reduce inference time through model pruning and depthwise separable convolutions, crucial for application in resource-constrained settings. Enhancing the website for bulk processing and considering a mobile app for local processing present further opportunities to increase the utility and accessibility of this technology in agricultural practices.

MATERIALS AND METHODS

Our study involved seven major steps that included data collection, exploratory data analysis, data preprocessing and augmentation, baseline CNN model development, transfer learning, model evaluation and website deployment.

Data Collection

To find a suitable dataset, we navigated to kaggle.com/ datasets to look for a dataset with at least three different classes, a healthy class, and at least 500 images per class. We found a satisfactory dataset, titled "Grape_disease," that was published by the kaggle.com user "Pushpa Lama" on July 1, 2021, It contained four classes (black rot, Esca, leaf blight, and healthy) with about 2,000 training images and about 500 testing images in each class (**Table 1**). These original images were stored as Joint Photographic Experts Group (JPG) files with a size of
256 pixels by 256 pixels. Consequently, the aspect ratio was 1:1. In total, the original dataset had 7,222 training images and 1,805 testing images (**Table 1**). Afterward, we created a kaggle.com notebook and added the dataset to the input directory.

Exploratory Data Analysis

To get a better feel for the data, we read all the image files and stored them in a dataframe. Next, we plotted a bar graph and pie chart showing the image distribution across classes. We found that black rot, Esca, leaf blight, and healthy images were 26.0%, 26.4%, 24.2%, and 23.3%, respectively, of the total images in the original dataset. In addition, eight images for each class were plotted on a grid. It appeared that leaves were taken off grapevines and photographed with a plain background.

Data Preprocessing and Augmentation

Next, we performed data preprocessing. Resizing the images to 224 pixels by 224 pixels allowed us to utilize a variety of transfer learning models in the subsequent steps. Additionally, we normalized the images by dividing all pixel values by 255, a common practice in image processing. This step scaled the pixel values to a range of 0 to 1, facilitating the model's learning process by ensuring numerical stability and speeding up convergence during training. Normalizing the data in this way helped to achieve faster, more efficient training by reducing the computational burden on the model. We decided to perform data augmentation to ensure that the final model could handle low-resolution images as well as images with natural backgrounds. We did the augmentations (**Figures 5A – D**), randomly on the entire dataset by techniques like zooming in or out of the image, flipping the image horizontally or vertically, rotating the image, increasing or decreasing the brightness of the image, and shearing the image.

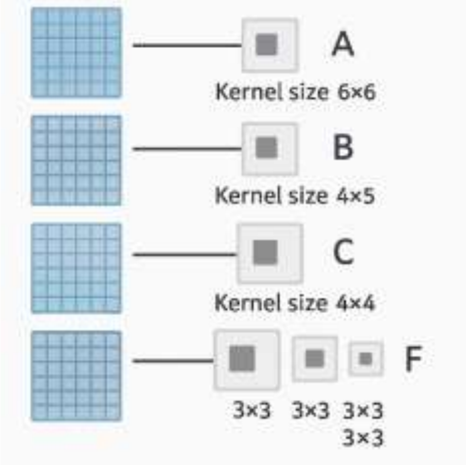


Figure 6: The baseline CNN model architecture. The input layer is denoted with the letter A and has dimensions 224x224x3. The convolutional layers with letters B, C, D, E, F, and G have dimensions 112x112x32, 56x56x32, 28x28x32, 14x14x32, 7x7x32, and 3x3x32 respectively. The kernel size is shown underneath the visual representation of the corresponding convolutional layer. All convolutional layers are shown along with the kernel size. The maxpooling from each convolutional layer to the next causes the width and height to halve. Finally, there are three hidden layers with 288, 512, and 512 nodes respectively and an output layer with 4 nodes. These layers are represented with a column of small orange boxes.

Finally, we combined 1,656 images from the original dataset with 1,344 augmented images for each class. Please note that we used 1,656 training images for each class as the original dataset had a different number of training images in each class, and we wanted to ensure the models were trained equally for each class. This led to each class having 3,000 images, with 12,000 images in total (**Table 1**). We published this newly created dataset on kaggle.com for others to use. Augmented images were used to help the model generalize, but they were not included in the testing data so the model could be evaluated on the real images present in the dataset.

Baseline CNN Model

First, we created the model architecture (**Figure 6**) and built it as a "sequential model" in TensorFlow. Next, we added two fully connected dense layers with the relu activation function and an output layer with four output neurons at the top of the model (**Figure 6** and **Appendix**). At this stage, we checked for overfitting and underfitting using training and validation loss. Additionally, we evaluated the model's performance using validation accuracy, a confusion matrix, and a classification report. The images that the model predicted inaccurately were visualized to understand the model's pitfalls better. We used Python Version 3.0.12 for programming, Jupyter Notebook Version 7.0.6 for development, and various Python libraries including TensorFlow, seaborn, scikit-learn, pandas, numpy and matplotlib for CNN modeling (**Appendix**).

Transfer Learning

The five CNN models tested using transfer learning were DenseNet121, EfficientNetB7, MobileNetV2, ResNet50, and VGG16. For each transfer learning model, we imported its pre-trained weights from ImageNet. Next, we created a sequential model containing the transfer learning model. We then added a GlobalAvgPool2D layer to the sequential model. By calculating the mean of the input's width and height, this layer performed downsampling. It reduced the total number of parameters and the chance of overfitting. Finally, we added a dense layer with a softmax activation. By doing this, raw neural network outputs were converted into probability vectors. Next, we set an EarlyStopping callback that stopped training if validation accuracy did not improve after eight epochs. In addition, we implemented a ModelCheckpoint callback that saved the model with the lowest validation loss. We compiled the sequential model with the Adam optimizer. Finally, we trained each model for 25 epochs, aligning with established conventions in machine learning. This duration struck a balance between undertraining and overfitting: too few epochs might prevent the model from fully learning from the dataset, while too many could cause the model to learn the training data too well, failing to generalize to new data.

Model Evaluation

We determined the most suitable CNN models for the max-voting ensemble using validation accuracies, confusion matrices, and error visualizations. In a max-voting ensemble, each base model made a prediction on an image. Each prediction counted as a vote, and the disease with the most votes was the final prediction. If there was a tie, the final prediction was decided by the highest-scoring base model. We evaluated the ensemble using accuracy score, confusion matrix, classification metrics, and error visualization. We also

Grapevine Disease Detection

This model can classify your image as Black Rot, ESCA, Leaf Blight, or Healthy.

Upload Image

Uh oh! Your grapevine plant has been diagnosed with having Black Rot with a probability of 0.987.

Figure 7: Grapevine disease detection website in action. A grapevine leaf that is affected with black rot has been uploaded. The user learned that the leaf is infected with black rot with 0.987 probability.

found out how long the ensemble took to classify a single image with and without a GPU. After that, we downloaded the HDF5 file containing the most accurate models' weights.

References

1. Alzamily, J. Y. I., et al. (2022). "Classification of encrypted images using deep learning–resnet50." Journal of Theoretical and Applied Information Technology 100(21): 6610-6620.
2. Al-Zamily, J. Y. I., et al. (2023). A survey of cryptographic algorithms with deep learning. AIP Conference Proceedings, AIP Publishing.
3. Arqawi, S. M., et al. (2022). "Predicting employee attrition and performance using deep learning." Journal of Theoretical and Applied Information Technology 100(21): 6526-6536.
4. Arqawi, S. M., et al. (2022). "Predicting university student retention using artificial intelligence." International Journal of Advanced Computer Science and Applications 13(9).
5. Ashqar, B. A. M. and S. S. Abu-Naser (2019). "Identifying Images of Invasive Hydrangea Using Pre-Trained Deep Convolutional Neural Networks." International Journal of Academic Engineering Research (IJAER) 3(3): 28-36.
6. Ashqar, B. A. M. and S. S. Abu-Naser (2019). "Image-Based Tomato Leaves Diseases Detection Using Deep Learning." International Journal of Academic Engineering Research (IJAER) 2(12): 10-16.
7. Ashqar, B. A., et al. (2019). "Plant Seedlings Classification Using Deep Learning." International Journal of Academic Information Systems Research (IJAISR) 3(1): 7-14.
8. Atallah, R. R. and S. S. Abu Naser (2014). "Data mining techniques in higher education an empirical study for the university of Palestine." International Journal Of Modern Engineering Research (IJMER) 4(4): 48-52.
9. Attia, M. W., et al. (2023). "Predicting Fire Alarms in Smoke Detection using Neural Networks." International Journal of Academic Information Systems Research (IJAISR) 7(10): 26-33.
10. Awadi, S., et al. (2020). "Effect of Oxygen Consumption of Thylakoid Membranes (Chloroplasts) From Spinach after Inhibition Using JNN." Effect of Oxygen Consumption of Thylakoid Membranes (Chloroplasts) From Spinach after Inhibition Using JNN.
11. Ayyad, M. and S. S. Abu-Naser (2024). "Fish Classification Using Deep Learning." International Journal of Academic Information Systems Research (IJAISR) 8(4): 51-58.
12. Bakr, M. A. H. A., et al. (2020). "Breast Cancer Prediction using JNN." International Journal of Academic Information Systems Research (IJAISR) 4(10): 1-8.
13. Baraka, R. F. and S. S. Abu-Naser (2023). "Predicting Books' Rating Using Just Neural Network." International Journal of Engineering and Information Systems (IJEAIS) 7(9): 14-19.
14. Barhoom, A. M. and S. S. Abu-Naser (2022). "Diagnosis of Pneumonia Using Deep Learning." International Journal of Academic Engineering Research (IJAER) 6(2): 48-68.
15. Barhoom, A. M., et al. (2019). "Predicting Titanic Survivors using Artificial Neural Network." International Journal of Academic Engineering Research (IJAER) 3(9): 8-12.
16. Barhoom, A. M., et al. (2022). "Bone abnormalities detection and classification using deep learning-vgg16 algorithm." Journal of Theoretical and Applied Information Technology 100(20): 6173-6184.
17. Barhoom, A. M., et al. (2022). "Deep Learning-Xception Algorithm for upper bone abnormalities classification." Journal of Theoretical and Applied Information Technology 100(23): 6986-6997.
18. Barhoom, A. M., et al. (2022). "Prediction of Heart Disease Using a Collection of Machine and Deep Learning Algorithms." International Journal of Engineering and Information Systems (IJEAIS) 6(4): 1-13.
19. Barhoom, A., et al. (2022). "Sarcasm Detection in Headline News using Machine and Deep Learning Algorithms." International Journal of Engineering and Information Systems (IJEAIS) 6(4): 66-73.
20. Barhoom, A., et al. (2023). A survey of bone abnormalities detection using machine learning algorithms. AIP Conference Proceedings, AIP Publishing.
21. Belbeisi, H. Z., et al. (2020). "Effect of Oxygen Consumption of Thylakoid Membranes (Chloroplasts) From Spinach after Inhibition Using JNN." International Journal of Academic Health and Medical Research (IJAHMR) 4(11): 1-7.
22. Dalffa, M. A., et al. (2019). "Tic-Tac-Toe Learning Using Artificial Neural Networks." International Journal of Engineering and Information Systems (IJEAIS) 3(2): 9-19.
23. Dawood, K. J., et al. (2020). "Artificial Neural Network for Mushroom Prediction." International Journal of Academic Information Systems Research (IJAISR) 4(10): 9-17.
24. Dawoud, A. M. and S. S. Abu-Naser (2023). "Predicting Life Expectancy in Diverse Countries Using Neural Networks: Insights and Implications." International Journal of Academic Engineering Research (IJAER) 7(9): 46-54.
25. Dheir, I. M., et al. (2020). "Classifying Nuts Types Using Convolutional Neural Network." International Journal of Academic Information Systems Research (IJAISR) 3(12): 12-18.
26. El_Jerjawi, N. S. and S. S. Abu-Naser (2018). "Diabetes Prediction Using Artificial Neural Network." International Journal of Advanced Science and Technology 121: 55-64.
27. El-Ghoul, M. and S. S. Abu-Naser (2024). "Vegetable Classification Using Deep Learning." International Journal of Academic Information Systems Research (IJAISR) 8(4): 105-112.
28. El-Habibi, M. F. and S. S. Abu-Naser (2024). "Tomato Leaf Diseases Classification using Deep Learning." International Journal of Academic Information Systems Research (IJAISR) 8(4): 73-80.
29. El-Habil, B. Y. and S. S. Abu-Naser (2021). "Cantaloupe Classification Using Deep Learning." International Journal of Academic Engineering Research (IJAER) 5(12): 7-17.
30. El-Habil, B. Y. and S. S. Abu-Naser (2022). "Global climate prediction using deep learning." Journal of Theoretical and Applied Information Technology 100(24): 4824-4838.
31. El-Kahlout, M. I. and S. S. Abu-Naser (2020). "Peach Type Classification Using Deep Learning." International Journal of Academic Engineering Research (IJAER) 3(12): 35-40.
32. El-Khatib, M. J., et al. (2019). "Glass Classification Using Artificial Neural Network." International Journal of Academic Pedagogical Research (IJAPR) 3(2): 25-31.
33. El-Mahelawi, J. K., et al. (2020). "Tumor Classification Using Artificial Neural Networks." International Journal of Academic Engineering Research (IJAER) 4(11): 8-15.
34. Elmahmoum, A. S., et al. (2025). "Image-Based Classification of Date Types Using Convolutional Neural Networks." International Journal of Academic Information Systems Research (IJAISR) 9(1): 10-16.
35. El-Mashharawi, H. Q., et al. (2020). "Grape Type Classification Using Deep Learning." International Journal of Academic Engineering Research (IJAER) 3(12): 41-45.
36. Elqassas, R., et al. (2024). "Convergence of Nanotechnology and Artificial Intelligence: Revolutionizing Healthcare and Beyond." International Journal of Engineering and Information Systems (IJEAIS) 8(10): 25-30.
37. Elsharif, A. A. E. F. and S. S. Abu-Naser (2022). "Retina Diseases Diagnosis Using Deep Learning." International Journal of Academic Engineering Research (IJAER) 6(2): 11-37.
38. Elsharif, A. A., et al. (2020). "Potato Classification Using Deep Learning." International Journal of Academic Pedagogical Research (IJAPR) 3(12): 1-8.
39. El-Tantawi, J. and S. S. Abu-Naser (2024). "The Fast Food Image Classification using Deep Learning." International Journal of Academic Information Systems Research (IJAISR) 8(4): 37-41.
40. Elzamly, A., et al. (2015). "Classification of Software Risks with Discriminant Analysis Techniques in Software planning Development Process." International Journal of Advanced Science and Technology 81: 35-48.
41. Elzamly, A., et al. (2015). "Predicting Software Analysis Process Risks Using Linear Stepwise Discriminant Analysis: Statistical Methods." Int. J. Adv. Inf. Sci. Technol 38(38): 108-115.
42. Elzamly, A., et al. (2017). "Predicting Critical Cloud Computing Security Issues using Artificial Neural Network (ANNs) Algorithms in Banking Organizations." International Journal of Information Technology and Electrical Engineering 6(2): 40-45.
43. Habib, N. S., et al. (2020). "Presence of Amphibian Species Prediction Using Features Obtained from GIS and Satellite Images." International Journal of Academic and Applied Research (IJAAR) 4(11): 13-22.
44. Hadi, A. M. A. and S. S. Abu-Naser (2023). "Forecasting Stock Prices using Artificial Neural Network." International Journal of Engineering and Information Systems (IJEAIS) 7(10): 42-50.
45. Hamad, M. S., et al. (2024). "Harnessing Artificial Intelligence to Enhance Medical Image Analysis." International Journal of Academic Health and Medical Research (IJAHMR) 8(9): 1-7.
46. Hamdan, M. K., et al. (2020). "Clarity of Vision and Its Relationship to the Creative Behavior of NGOs." International Journal of Academic Management Science Research (IJAMSR) 4(4): 55-82.
47. Hamdan, M. K., et al. (2020). "Creative Behavior and Impact on Achieving Lean Strategy in Organizations." International Journal of Academic Accounting, Finance & Management Research (IJAAFMR) 4(6): 66-90.
48. Hamdan, M. K., et al. (2020). "Creative Behavior in Palestinian NGOs between Reality and Expectations." International Journal of Academic Multidisciplinary Research (IJAMR) 4(3): 91-107.
49. Hamdan, M. K., et al. (2020). "Shared Responsibility, Rapid Response and Their Relationship to Developing the Creative Behavior of Organizations." International Journal of Academic Management Science Research (IJAMSR) 4(7): 1-21.
50. Hamdan, M. K., et al. (2020). "Strategic Sensitivity and Its Impact on Boosting the Creative Behavior of Palestinian NGOs." International Journal of Academic Accounting, Finance & Management Research (IJAAFMR) 4(5): 80-102.
51. Hamdan, M. K., et al. (2020). "Taking Action, Rapid Response and Its Role in Improving the Creative Behavior of Organizations." International Journal of Academic Accounting, Finance & Management Research (IJAAFMR) 4(4): 41-62.
52. Hamdan, M. K., et al. (2020). "The Effect of Choosing Strategic Goals and Core Capabilities on the Creative Behavior of Organizations." International Journal of Academic Information Systems Research (IJAISR) 4(4): 56-75.
53. Hamdan, M. K., et al. (2020). "The Reality of Applying Strategic Agility in Palestinian NGOs." International Journal of Academic Multidisciplinary Research (IJAMR) 4(4): 76-103.
54. Hamdan, M. K., et al. (2021). "The Reality of the Practice of Crisis Management in the Union of Health Work Committees in Gaza In Light of the Corona Pandemic." International Journal of Academic Management Science Research (IJAMSR) 5(4): 141-148.
55. Hamdan, M. K., et al. (2021). "The Reality of the Quality of Health Services in the Union of Health Work Committees in Gaza In Light of the Corona Pandemic." International Journal of Academic Accounting, Finance and Management Research(IJAAFMR) 5(4): 97-104.
56. Hamed, M. A. and F. Mohammed (2024). "El-Habib, Raed Z. Sababa, Mones M. Al-Hanjor, Basem S. Abunasser, and Samy S. Abu-Naser."." Artificial Intelligence in Agriculture: Enhancing Productivity and Sustainability.
57. Hamed, M. A., et al. (2024). "Artificial Intelligence in Agriculture: Enhancing Productivity and Sustainability." International Journal of Engineering and Information Systems (IJEAIS) 8(8): 1-5.
58. Harara, F. E., et al. (2022). "Figs Knowledge Based System Disease Diagnosis and Treatment." International Journal of Academic Engineering Research (IJAER) 6(5): 41-45.
59. Harara, M. and S. S. Abu-Naser (2023). "Unlocking Literary Insights: Predicting Book Ratings with Neural Networks." International Journal of Academic Information Systems Research (IJAISR) 7(10): 22-27.
60. Harz, H. H., et al. (2020). "Artificial Neural Network for Predicting Diabetes Using JNN." International Journal of Academic Engineering Research (IJAER) 4(10): 14-22.
61. Hasan, W., et al. (2023). "Artificial Neural Network for Predicting COVID 19 Using JNN." International Journal of Academic Engineering Research (IJAER) 7(10): 41-47.
62. Hassanein, R. A. A., et al. (2020). "Artificial Neural Network for Predicting Workplace Absenteeism." International Journal of Academic Engineering Research (IJAER) 4(9): 62-67.
63. Heriz, H. H., et al. (2018). "English Alphabet Prediction Using Artificial Neural Networks." International Journal of Academic Pedagogical Research (IJAPR) 2(11): 8-14.
64. Hissi, H. E.-., et al. (2008). "Medical Informatics: Computer Applications in Health Care and Biomedicine." Journal of Artificial Intelligence 3(4): 78-85.
65. Husien, A. M., et al. (2023). "Predicting Students' end-of-term Performances using ML Techniques and Environmental Data." International Journal of Academic Information Systems Research (IJAISR) 7(10): 19-25.
66. Ibaid, M. R. and S. S. Abu-Naser (2024). "Using Deep Learning to Classify Eight Tea Leaf Diseases." International Journal of Academic Information Systems Research (IJAISR) 8(4): 89-96.
67. Ikram, B., et al. (2022). "Malimg2022: Data Augmentation And Transfer Learning To Solve Imbalanced Training Data For Malware Classification." Journal of Theoretical and Applied Information Technology 101(4).
68. Jaber, A. S., et al. (2020). "Evolving Efficient Classification Patterns in Lymphography Using EasyNN." International Journal of Academic Information Systems Research (IJAISR) 4(9): 66-73.
69. Jamala, M. N. and S. S. Abu-Naser (2018). "Predicting MPG for Automobile Using Artificial Neural Network Analysis." International Journal of Academic Information Systems Research (IJAISR) 2(10): 5-21.
70. Jamala, M. N. and S. S. Abu-Naser (2023). "Knowledge Based System for Diagnosing Lung Cancer Diagnosis and Treatment." International Journal of Academic Information Systems Research (IJAISR) 7(6): 38-45.
71. Jamala, M. N., et al. (2025). "Identifying Fish Species Using Deep Learning Models on Image Datasets." International Journal of Academic Information Systems Research (IJAISR) 9(1): 1-9.
72. Jarghon, A. and S. S. Abu-Naser (2023). "Predicting Kidney Stone Presence from Urine Analysis: A Neural Network Approach using JNN." International Journal of Academic Information Systems Research (IJAISR) 7(9): 32-39.
73. Karaja, M. B. and S. S. Abu-Naser (2024). "Using Deep Learning to Detect the Quality of Lemons." International Journal of Academic Information Systems Research (IJAISR) 8(4): 97-104.
74. Kashf, D. W. A., et al. (2018). "Predicting DNA Lung Cancer using Artificial Neural Network." International Journal of Academic Pedagogical Research (IJAPR) 2(10): 6-13.
75. Kassab, M. K. I., et al. (2017). "An Analytical Study of the Reality of Electronic Documents and Electronic Archiving in the Management of Electronic Documents in the Palestinian Pension Agency (PPA)." EUROPEAN ACADEMIC RESEARCH 6(12): 10052-10102.
76. Kassab, M. K. I., et al. (2019). "The Role of Policies and Procedures for the Electronic Document Management System in the Success of the Electronic Document Management System in the Palestinian
77. Khalil, A. J. and S. S. Abu-Naser (2022). "Diagnosis of Blood Cells Using Deep Learning." International Journal of Academic Engineering Research (IJAER) 6(2): 69-84.