

Article

Not peer-reviewed version

HybridPlantNet23: A Scientific Insight into the Power of Ensemble Modelling using VGG16 and Darknet53 for Plant Disease Classification

[Tahir Cetin AKINCI](#)^{*}, [Sami Ekici](#)^{*}, [Masud Kabir](#), [Alfredo A. Martinez-Morales](#)

Posted Date: 8 July 2024

doi: 10.20944/preprints202407.0556.v1

Keywords: classification; deep learning; pretrained models; plant diseases; ensemble model



Preprints.org is a free multidiscipline platform providing preprint service that is dedicated to making early versions of research outputs permanently available and citable. Preprints posted at Preprints.org appear in Web of Science, Crossref, Google Scholar, Scilit, Europe PMC.

Copyright: This is an open access article distributed under the Creative Commons Attribution License which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Article

HybridPlantNet23: A Scientific Insight into the Power of Ensemble Modelling using VGG16 and Darknet53 for Plant Disease Classification

Masud Kabir ¹, Tahir Cetin Akinci ^{2,*}, Sami Ekici ^{1,*} and Alfredo A. Martinez-Morales ²

¹ Department of Energy Systems Engineering, Firat University, 23119 Elazığ, Turkey

² Center for Environmental Research and Technology, University of California Riverside, Riverside, CA 92507, USA

* Correspondence: tahircetin.akinci@ucr.edu (T.C.A.); sekici@firat.edu.tr (S.E.)

Abstract: This study evaluates the effectiveness of Darknet53 and VGG16 pretrained models to develop a hybrid ensemble model named HybridPlantNet23 deep learning model for effective plant disease classification, using the publicly available dataset 'A Dataset of Leaf Images' from Mendeley webpage. The preprocessing of the dataset involved resizing the images to the respective requirements of the pretrained models to develop a combined datastore which served as the input. All experiments were conducted using MATLAB R2023b deep network designer interface. Model performances were evaluated based on the conventional accuracy, precision, recall, and F1-score. VGG16 achieved the highest accuracy (97.25%), precision (96.30%), recall (96.96%), and F1-score (96.55%), outperforming both Darknet53 and the ensemble HybridPlantNet23 model. Despite the advanced architecture of the ensemble model, its performance (accuracy: 95.37%, precision: 93.56%, recall: 94.20%, F1-score: 93.74%) was not significantly superior, thus highlighting scientific reasons behind this development. Recommendations for future works were provided accordingly.

Keywords: classification; deep learning; pretrained models; plant diseases; ensemble model

1. Introduction

The foundation of human civilization is the agricultural sector, which supplies the resources required for both survival and economic growth [1]. Yet, there are many obstacles this vital sector must overcome, and among the most dangerous are plant diseases. Plant diseases reduce crop production while also compromising food quality and safety, resulting in major economic losses and affecting food security [2–5]. In order to effectively manage and mitigate these illnesses and maintain the stability and productivity of agricultural practices, early and precise detection of these diseases is necessary [6–8].

Traditionally, plant disease diagnosis was mainly reliant on manual inspection by trained agronomists and farmers [9]. Although this method worked well, it was labor-intensive, time-consuming, and frequently subjective, which results in inconsistent disease diagnosis. Furthermore, the swift dissemination of plant illnesses and the introduction of novel pathogens demand a more accurate and scalable diagnostic methodology. An encouraging answer to these problems is the development of deep learning and computer vision technologies, which make automated, precise, and effective plant disease identification through picture analysis possible [10,11]. One major example is the convolutional neural networks (CNNs).

With its exceptional performance in a variety of fields, such as autonomous driving [12], facial recognition [13], and medical imaging [14], CNNs have completely changed the area of image classification. CNNs have demonstrated considerable potential in agriculture for recognizing diseases in plants from photos of its several parts, including leaves, stems, and fruits [15–18]. However, because there are intrinsic trade-offs between model complexity, depth, and feature

extraction capabilities, no single CNN design can be said to be the best for all kinds of image classification applications. This in turn called for the idea of model hybridization.

Deep learning has witnessed a surge in interest in hybrid models in recent times. In order to take advantage of their complimentary qualities and improve overall performance, hybrid models blend several neural network architectures. The reasoning behind this strategy is that certain designs are better at identifying patterns and extracting features in different ways. Hybrid models, by combining these diverse qualities, can achieve higher accuracy and robustness than standalone models [17,19].

In a number of fields, including medical diagnosis, hybrid models have been effectively used. By fusing CNNs and Recurrent Neural Networks (RNNs) to investigate long-term survival of subjects in a lung cancer screening, illness prediction using sequential data has become more accurate [20]. Hybrid models that combine CNNs and Transformers have improved sentiment analysis and text categorization in the field of natural language processing [21].

In the context of agriculture, plant disease detection specifically, Poornim et al. (2023) [22] designed a lightweight ‘VGG-ICNN’ hybrid model by training on PlantVillage dataset, Embrapa dataset and single crop datasets. They obtained a 99.61% accuracy on PlantVillage set, while establishing success for the other datasets utilized. Chin Poo Lee et al. (2023) [19] developed a robust ensemble model called Plant-CNN-Vi by combining four pre-trained models; Vision Transformer, ResNet-50, DenseNet-201, and Xception, respectively. Other relevant literature were summarized and presented as Table 1.

In this study, we investigated the effectiveness of integrating two robust CNN architectures, VGG16 and Darknet53, into HybridPlantNet23, to improve plant disease classification systems. VGG16, created by the Visual Geometry Group at Oxford, is known for its simplicity and depth, with 16 layers that excel in extracting hierarchical elements from images. Darknet53, the backbone of the YOLO (You Only Look Once) object identification framework, is a more complicated and deep network with 53 convolutional layers that is well-known for its resilient feature extraction and exceptional performance in a variety of computer vision tasks. Utilizing the complementing qualities of both architectures is the driving force behind this hybrid strategy. The design of VGG16 makes it easier to extract fine-grained, low-level characteristics, which are important for recognizing small variations in plant disease symptoms. Darknet53’s richer structure allows it to capture high-level, abstract information, which improves the model’s capacity to generalize across disease kinds and environmental situations. We hope to construct a more robust and accurate model capable of classifying a diverse spectrum of plant illnesses by merging these architectures. The study has practical ramifications for farmers, agronomists, and agricultural researchers, making it significant outside the confines of academia. A precise and dependable plant disease classification system can enable prompt and targeted interventions, lowering crop losses and increasing output quality. In addition, the implementation of these systems in farming operations can improve disease surveillance, facilitate precision farming methods, and foster the growth of sustainable agriculture.

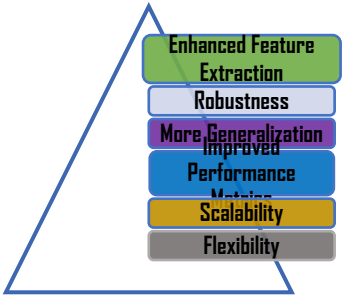


Figure 1. Values of Ensemble Modeling.

Table 1. Summary of relevant literature.

Source	Model	Method	Dataset(s)	Accuracy (%)	Remark
Lee et al. (2015) [23]	Fine-tuned CNN	Feature extraction using CNN, classified using MLP/SVM	MalayaKew Leaf dataset, D1, D2	99.5	Demonstrates the high accuracy achievable with a fine-tuned CNN and MLP combination
Liu et al. (2018) [24]	10-layer CNN (LeNet based)	Feature extraction and classification with data augmentation	Flavia dataset	87.92	Highlights the effectiveness of data augmentation in improving CNN performance
Tan et al. (2018) [25]	D-Leaf	Six-layer CNN, features classified using various machine learning techniques	MalayaKew, Flavia, Swedish Leaf datasets	98.09 (highest)	Shows the strength of combining multiple models for feature extraction
Hu et al. (2018) [26]	MSF-CNN	Multiscale fusion CNN with bilinear interpolation	MalayaKew Leaf dataset (D2)	99.82	Emphasizes the advantages of multiscale feature extraction and fusion
Kaya et al. (2019) [27]	Pre-trained VGG16 and LDA	Various CNN and machine learning combinations	Flavia, Swedish Leaf, UCI datasets	99.11 (highest)	Indicates the benefits of combining pre-trained models with traditional classifiers
Anubha et al. (2019) [28]	VGG16, VGG19, Inception-v3, Inception-ResNet-v2	Feature extraction with pre-trained models, classified using various ML techniques	Leaf12, Folio, Flavia, Swedish Leaf datasets	99.41 (highest)	Highlights the potential of using pre-trained models for feature extraction
Riaz et al. (2020) [29]	MPF-CNN	Multi-path multi-convolutional network	MalayaKew dataset (D2)	98.71	Demonstrates the effectiveness of multi-path architecture in CNN
Litvak et al. (2022) [30]	Various pre-trained CNN models	Evaluated models on Urban Planter dataset	Urban Planter dataset	96	Shows the importance of pre-training on large datasets like ImageNet
Arun and Viknesh (2022) [31]	EfficientNet-B5	Training pre-trained models on leaf images	Plant leaf image dataset (11 plant species)	99.75	Indicates the superior performance of EfficientNet in plant leaf classification
Beikmohammadi et al. (2022) [32]	SWP-LeafNET	Combines S-LeafNET, W-LeafNET, and P-LeafNET	Flavia, MalayaKew datasets	99.81 (highest)	Shows the effectiveness of combining multiple CNN models with a voting mechanism

2. Materials and Methods

2.1. Dataset

The dataset used in this study is A Database of Leaf Images [33] available on the Mendeley Data source. It consists of images of 12 different plant diseases. The dataset’s summary is given in Table 2 while its class samples could be seen in Figure 2. The dataset was chosen for this study owing to previously conducted study justifying its superior quality and high significance for deep learning application, through its GLCM metrics rigorous analyses and concurrence with deep learning train-

test results as in [34]. To better understand the class samples labelled in series, the sequential numbering utilized for this studies is given as in Table 3.

Table 2. Brief of the Utilized Dataset.

Attribute	Description
Source	Mendeley Plant Disease Dataset
Plant Species	Mango, Arjun, Alstonia Scholaris, Guava, Bael, Jamun, Jatropha, Pongamia Pinnata, Basil, Pomegranate, Lemon, Chinar
Total Number of Images	4503 (2278 healthy, 2225 diseased)
Annotation	Plant classes P0-P11 Healthy classes 0000-0011 Diseased classes 0012-0022
Collection Location	Shri Mata Vaishno Devi University, Katra
Collection Period	March to May 2019
Environment	Closed, Wi-Fi enabled
Image Format	JPEG
Image Specifications	24-bit depth, 2 resolution units, 1000-ISO, no flash
Performance Timing	JPEG: 0.58 seconds/frame, RAW+JPEG: 0.63 seconds/frame
Camera	Nikon D5300
Lens	18-55mm
Color Representation	sRGB
Applications	Plant identification, classification, growth monitoring, disease diagnosis

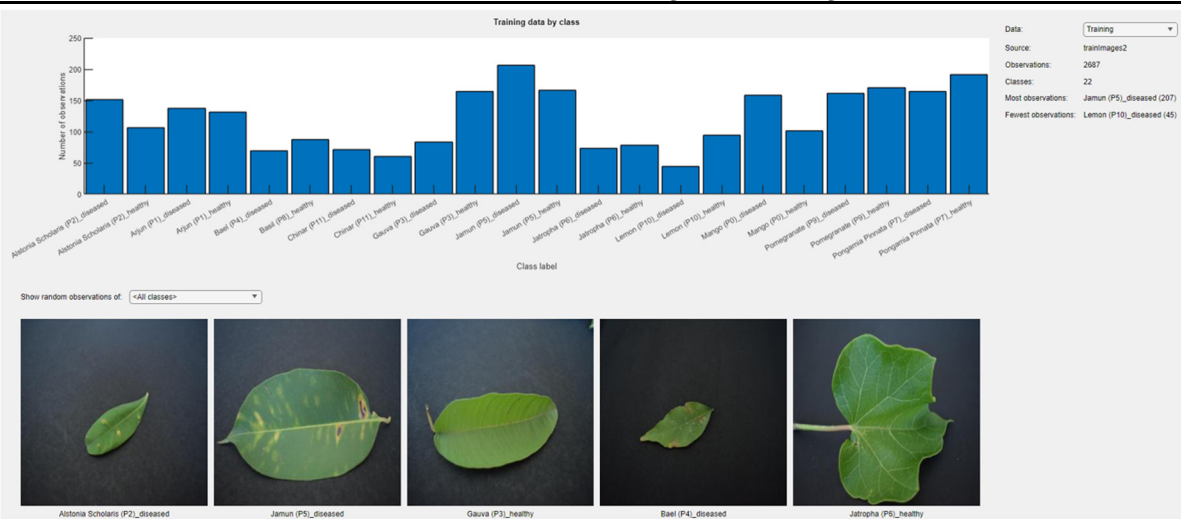


Figure 2. Sample data with class information.

Table 3. Class Labels.

Classes	Labels
1.	Alstonia Scholaris (P2)_diseased
2.	Alstonia Scholaris (P2)_healthy
3.	Arjun (P1)_diseased
4.	Arjun (P1)_healthy
5.	Bael (P4)_diseased
6.	Basil (P8)_healthy

7.	Chinar (P11)_diseased
8.	Chinar (P11)_healthy
9.	Gauva (P3)_diseased
10.	Gauva (P3)_healthy
11.	Jamun (P5)_diseased
12.	Jamun (P5)_healthy
13.	Jatropha (P6)_diseased
14.	Jatropha (P6)_healthy
15.	Lemon (P10)_diseased
16.	Lemon (P10)_healthy
17.	Mango (P0)_diseased
18.	Mango (P0)_healthy
19.	Pomegranate (P9)_diseased
20.	Pomegranate (P9)_healthy
21.	Pongamia Pinnata (P7)_diseased
22.	Pongamia Pinnata (P7)_healthy

2.2. Dataset Preprocessing

The dataset’s raw images were 4000x6000x3 pixels in size. The photographs were scaled to comply with the input size specifications of the pretrained models VGG16 and Darknet53. The input size requirements for VGG16 and Darknet53 are 224x224x3 and 256x256x3, respectively. This scaling was necessary to guarantee that the convolutional layers of these models could process the images efficiently and without experiencing any dimensionality problems.

MATLAB R2023b was used for all preprocessing operations and the ensuing model training. This MATLAB edition is ideal for managing big datasets and intricate neural network designs because it offers sophisticated tools and functions for deep learning, image processing, and integration with pretrained models.

2.3. Model Architecture

The proposed hybrid model combines the VGG16 and Darknet53 architectures, as to be shown. The final layers of both networks are concatenated to form a more comprehensive feature representation, followed by fully connected layers, a softmax layer, and a classification layer. To further understand the concept, the respective models are initially described as follows:

2.3.1. VGG16

VGG16 is a widely used CNN architecture known for its simplicity and depth, consisting of 16 weight layers. It is adept at extracting hierarchical features from images. VGG16 was introduced in the year 2015 as an advanced convolutional neural network model by Simonyan and Zisserman [35] from the University of Oxford. Using ImageNet, a dataset containing over 14 million images divided into 1000 classes, the model attains 92.7% top 5 test accuracy. By sequentially replacing the large kernel-sized filters (11 and 5 in the first and second convolutional layers, respectively) with multiple 3x3 kernel-sized filters, it proves to be an enhanced version of AlexNet. In this study, VGG16 and Darknet53 were selected to develop HybridPlantNet23 due to its superb accuracy that it showed with the same selected dataset herein, confirmed from another study conducted by the authors, which is yet to be published by Springer Nature as a book chapter under the series name “Proceedings of Science and Technology”. A general concept of how VGG-16 works is provided in Figure 3.

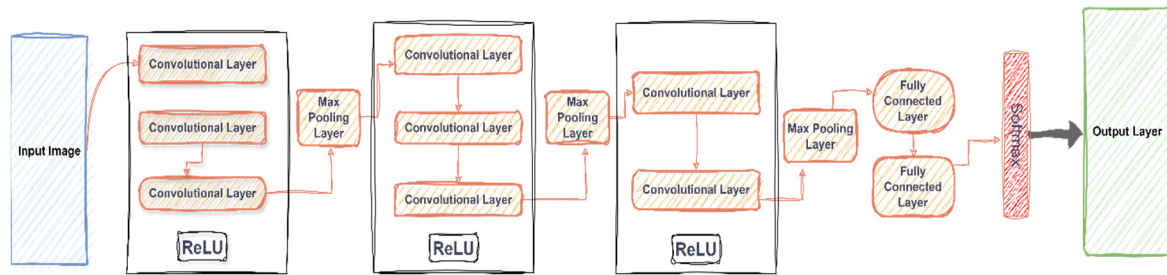


Figure 3. Schematic of VGG-16 Model's architecture.

An RGB image with a fixed size of 224 by 224 is the input for the convolution layer 1. The image is fed through a series of convolutional layers, where filters are applied with a little receptive field of 3x3. This is the smallest size that can effectively capture left/right, up/down, and center concepts. As a linear transformation of the input channels, one of the configurations additionally makes use of 1x1 convolution filters. The spatial padding of the convolution layer input is set to preserve the spatial resolution after the convolution, meaning that for 3x3 convolution layers, the padding is 1 pixel. The convolution stride is fixed at 1 pixel. The five max-pooling layers, which come after parts of the convolutional Layers, handle spatial pooling. Using stride 2, max-pooling is applied to the 2x2 pixel window. The stack of convolutional layers is followed by three Fully-Connected (FC) layers, each with 4096 channels. The third FC layer, which conducts 1000-way ILSVRC classification, has 1000 channels in total. The convolutional layers vary in depth and architecture. The softmax layer comes in last. Every network has the same configurations for the completely connected tiers. Every buried layer has non-linearity in rectification (ReLU). Notedly, all but one of the networks lack Local Response Normalization (LRN); this normalization results in higher computation and memory usage but does not enhance performance on the ILSVRC dataset [36–39].

2.3.2. Darknet53

Darknet-53 is a convolutional neural network (CNN) architecture that serves as the backbone for the YOLOv3 (You Only Look Once, version 3) object detection system. It is an advancement over earlier iterations of YOLO, created especially to improve the precision and effectiveness of object identification activities [40]. The advantages of this model include improved accuracy, efficiency, and versatility. A schematic of the model's general concept is provided in Figure 3.

Darknet-53, as the name implies, has 53 convolutional layers that are deeper than the 19 layers that make up Darknet-19, the foundation of YOLOv2. The network can learn more intricate features and improves detection accuracy with the increased depth. A combination of 1x1 and 3x3 convolutional filters is used by the network. While the 1x1 filters are used for dimension reduction, which lowers computational complexity, the 3x3 filters are utilized to capture spatial characteristics. The residual connections in Darknet-53 are modeled after ResNet (Residual Networks). With a shortcut link that adds the block's input to its output, each residual block is made up of two convolutional layers. By making these connections, the vanishing gradient issue is lessened and the network is able to train more efficiently. Bottleneck layers with 1x1 convolutions are employed within residual blocks to minimize the number of input channels, thus assisting in lowering the computational cost without sacrificing performance [41–43].

By normalizing the output of each layer, batch normalization, when applied after each convolutional layer, aids in stabilizing and speeding up the training process. The network can learn more intricate representations by introducing non-linearity through the use of the Leaky Rectified Linear Unit (Leaky ReLU) activation function. By permitting a little, non-zero gradient while the unit is dormant, the leaky version aids in the prevention of dead neurons. The model employs global average pooling at the final stage of the convolutional layers in place of fully linked layers. As a result, a feature vector for every class is produced, bringing the spatial dimensions down to 1x1. Depending

on the particular application, the resultant feature vector from global average pooling is then processed through fully connected layers to carry out classification or regression tasks [41,42,44].

Darknet-53’s main responsibility in YOLOv3 is to extract features from the input pictures [40]. Remaining connections in the deeper network facilitate the capture of more abstract and detailed data, which are essential for precise object detection.

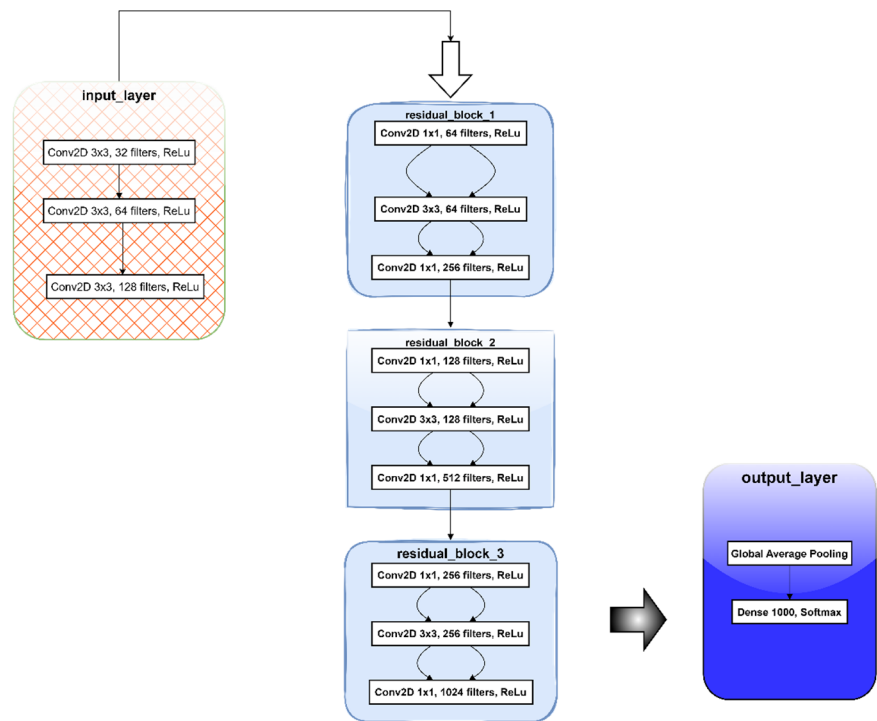


Figure 4. Schematic of Darknet53 Model’s architecture.

Table 4. HybridPlantNet23 model’s architectural description.

Model Components	Functionality
Dropout Layer (drop7)	Reduces overfitting by randomly setting a fraction of input units to 0 at each update during training
Fully Connected Layer (fc8_new)	Maps the learned features to the final output layer
Global Average Pooling Layer (avg1)	Reduces the spatial dimensions of the feature map, keeping only the average of each feature map
Convolutional Layer (conv53_new)	Extracts features from the input images
Concatenation Layer (concat)	Combines the features from both VGG16 and Darknet53
Fully Connected Layer (fc)	Further processes the combined features.
Softmax Layer (softmax)	Converts the output into probability distributions
Classification Layer (classoutput)	Produces the final classification result

2.3.3. HybridPlantNet23

HybridPlantNet23 is an enhanced model ensembled with the purpose of precisely identifying and categorizing plant diseases. In order to attain excellent performance in recognizing plant illnesses, this model carefully integrates the strengths of two potent convolutional neural network (CNN) architectures: Darknet53 and VGG-16. Using a large dataset of pictures of different plant species in both healthy and disease conditions, the VGG-16 and Darknet53 ensemble approach in HybridPlantNet23 trains both models independently. Using its special advantages, each model gains the ability to distinguish different features from the input images. Its major architectural descriptions

are presented and briefly discussed in Table 4. As it could be seen vividly from stage number 4 of the methodology, provided in Figure 5, the layers utilized and the arranged is presented.

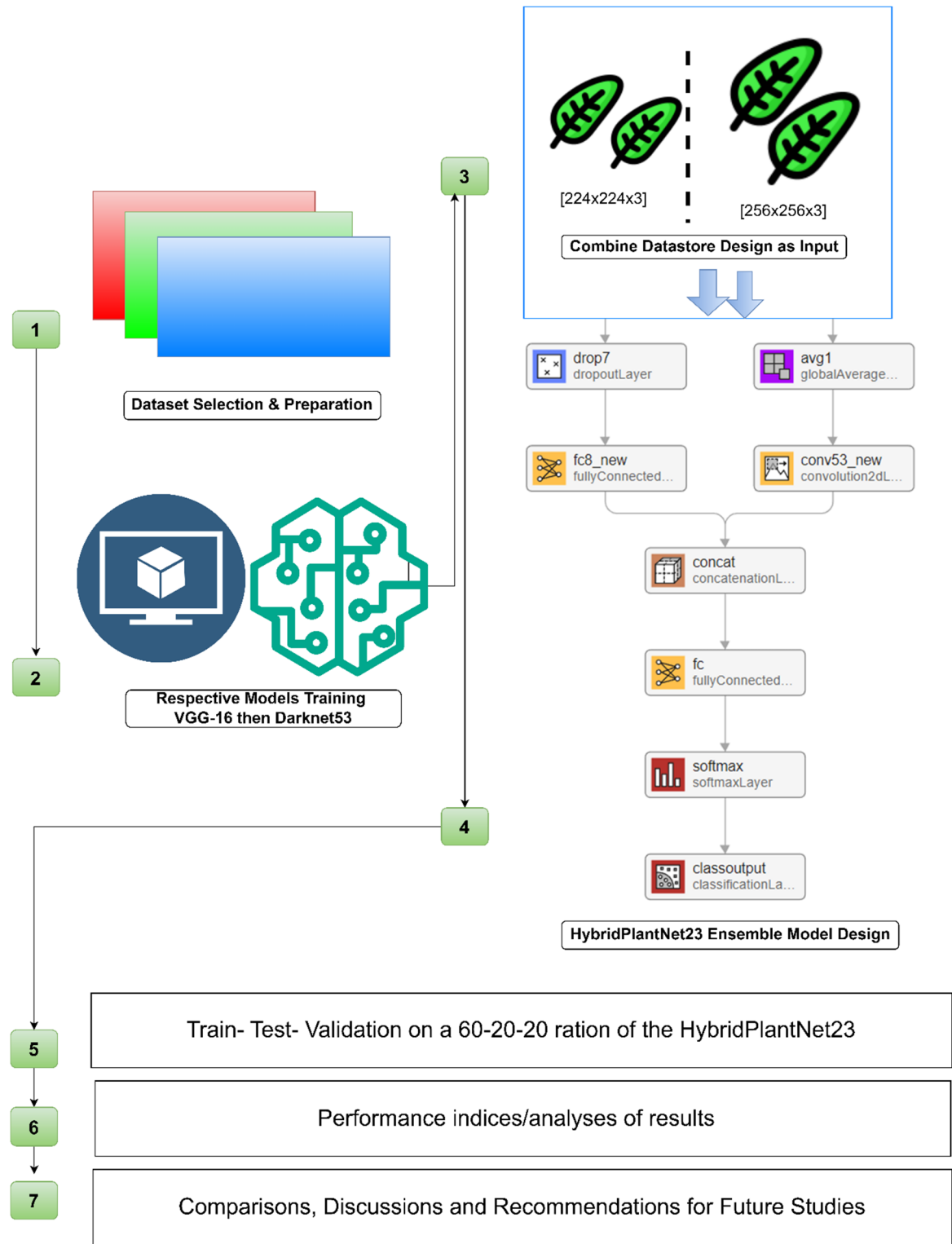


Figure 5. Hybrid Model Development’s Methodology.

2.4. Combined Datastore

To guarantee that each model received images of the right size, separate datastores were established for VGG16 and Darknet53. For this, MATLAB’s imageDatastore function was utilized, which effectively manages big image collections. Preprocessing pipelines were included in each

datastore to resize images to the necessary sizes of 256x256x3 for Darknet53 and 224x224x3 for VGG16, respectively. As is often done in image preparation, these pipelines also normalized pixel values by removing the mean and dividing by the standard deviation.

To create a merged feature set, the feature vectors that were extracted from VGG16 and Darknet53’s penultimate layers were concatenated. With VGG16 being particularly good at capturing fine-grained textures and Darknet53 being particularly good at larger-scale features, this combination potentially made the most of both models’ strengths.

2.5. Training

The hybrid model is trained using 2687 images (60%) of the dataset with an initial learning rate of 0.01. The categorical cross-entropy loss function is used to handle the multi-class classification problem. The model’s selected training parameters show a carefully calculated trade-off between model accuracy and training speed. Given the depth and complexity of the combined VGG-16 and Darknet53 architectures, the model’s convergence is accelerated by the relatively high initial learning rate.

To minimize the risk of overfitting, the model was trained with 30 epochs, which is enough to teach it all the complex properties needed to detect plant diseases accurately. The conventional selection of 32 mini-batch sizes offers a fair balance between the computational efficiency of the training process and the stability of gradient estimations. For every epoch, the data was shuffled and validated to verify that the model generalizes effectively and its performance is periodically reviewed, allowing for fast modifications supposing the model begins to overfit.

Figure 6 provides the training progress of the ensemble model. The top graph displays the training and validation accuracy over respective iterations. The blue line is representing the training accuracy. Dashed black line visible in the figure is a representation of the validation accuracy. The bottom graph further displays the training and validation loss over iterations where red line presents the training loss. The validation loss is represented by the dashed black line.

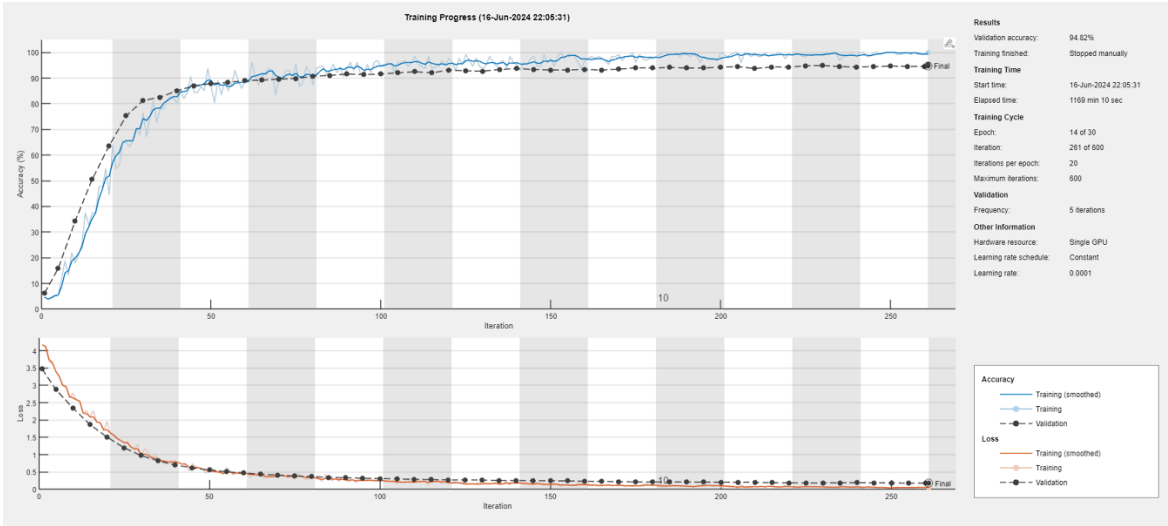


Figure 6. Training progress of HybridPlantNet23.

The model’s validation accuracy was recorded as 94.82%. This high validation accuracy shows how well it performs on unseen data, hence indicating effective learning and generalization. The training process was halted manually because the model had already reached satisfactory performance levels, a recommended by experts and professionals in the field. A significant computational effort is indicated by the total training time elapsed, which is typically necessary for training deep learning models with large datasets. The model was trained for 14 epochs out of the planned 30, suggesting early stopping due to reaching satisfactory performance. Furthermore, it only completed 261 iterations out of the maximum 600 planned iterations.

To ensure frequent monitoring and adjustment of the training process, the model’s performance was validated in every 5 iterations. Single GPU and constant learning rate (0.0001) were utilized straight throughout the training phase of the model. The relatively low learning rate is necessary for careful weight adjustments, which helps in achieving fine-tuned model performance.

2.6. Performance Evaluation

Total images used to test and validate the model were 908 and 907 images, representing a 20% and 20%, respectively, of the total images. The model’s performance is evaluated using standard metrics such as accuracy, precision, recall, and F1-score, which calculated using equations 1- 4 respectively. The results are compared with those of standalone VGG16 and Darknet53 models, earlier trained before the ensemble model, to demonstrate the effectiveness of the hybrid approach.

Accuracy = (True Positives + True Negatives) /Total Observations (1)

Recall = True Positives / (True Positives + False Negatives) (2)

Precision = True Positives / (True Positives – False Positives) (3)

F1 – Score = 2 x (Precision x Recall) / (Precision + Recall) (4)

3. Results

This section provides a concise and precise description of the results and their interpretation.

Table 5. Model Testing Performance Comparison.

Model	Accuracy	Precision	Recall	F1-Score
Darknet53	0.94934	0.92415	0.93523	0.92547
VGG16	0.97247	0.96304	0.96958	0.96550
HybridPlantNet23	0.95374	0.93556	0.94204	0.93742

4. Discussion

The training, test and performance metrics observed in this study indicate a scientific progress and reliability of these models in plant disease classification. To further digest the results, the following itemizations will be utilized:

4.1. Single Models Classification

The performance across different classes is displayed in the Darknet53 model’s confusion matrix (Figure 7). Although there are some noticeable instances where the model falls short, overall accuracy is quite great. As an example, class 15 (lemon) exhibits a notable decline in accuracy, which is reflected in the performance metrics and the confusion matrix.

Class 15, where the accuracy of the model is only 31.2%, is where it struggles the most. This implies that the model has trouble differentiating between other classes and lemons. In the dataset, class 15 had the fewest photos, suggesting that there is a class imbalance. Higher misclassification rates may result from the model’s under fitting of this class as a result of this imbalance. Moreover, Class 7 performs better with an accuracy of 87.5%, while class 8 has an accuracy of 85.7%. These classes still exhibit some misclassification, but the effect is less severe than in class 15. This suggests that even with a certain amount of misclassification, the model works fairly well on these classes overall.

Compared to Darknet53, the VGG16 confusion matrix (Figure 8) displays a more balanced performance across classes, as well as higher overall accuracy, precision, recall, and F1-score.

Although class 15’s accuracy is 75%, it is still significantly lower than Darknet53, indicating a considerable issue in accurately categorizing lemons. This improvement implies that VGG16’s architecture may be able to manage the class imbalance marginally better than Darknet53’s, but the

problem still exists because of the dataset’s inherent imbalance. Furthermore, Class 8 accuracy of 95.2% and class 7 accuracy of 87.5% demonstrate the robust performance of VGG16. The resilience of VGG16 in managing these classes is demonstrated by these results, which are probably attributable to its deeper network and superior feature extraction skills when compared to Darknet53.

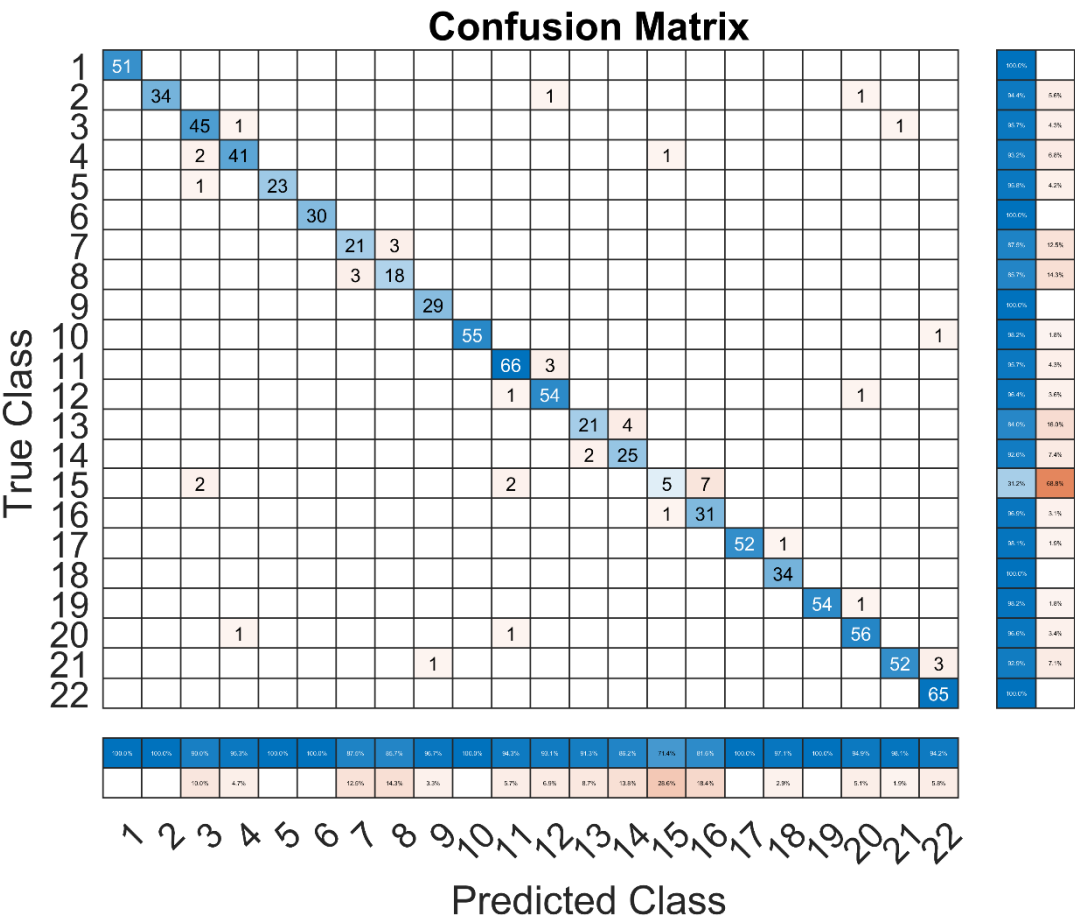


Figure 7. Darknet53 (single) confusion matrix.

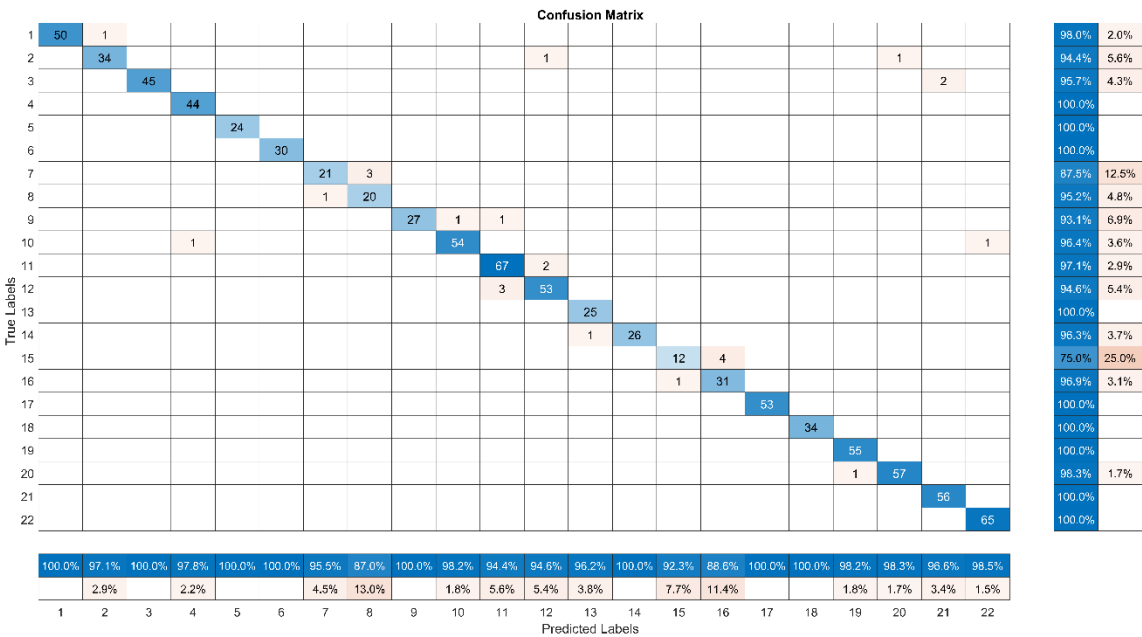


Figure 8. VGG16 (Single) Confusion Matrix.

4.2. HybridPlantNet23 Classification

The training accuracy of the ensemble model rapidly increases from the start and begins to plateau around 90-100 iterations. As the validation accuracy also follows a similar trend, this implies that the model generalizes effectively well to unseen data. After 200 iterations, both the training and validation accuracies stabilized around 95%, thus indicating the efficient and excellent performance of the model. From the training and validation loss over iterations observed earlier in Figure 6, the training loss decreases rapidly at the beginning and then gradually levels off, thereby indicating the model’s learning capability and how it adjusts its weights effectively. The validation loss, as well, follows a similar trait, mirroring the training loss closely, thereby suggesting absence of overfitting in the model.

The provided confusion matrix in Figure 9 presents a detailed view of the performance of the ensemble HybridPlantNet23 model in the act of classifying plant diseases as trained. Normally, a confusion matrix allows stakeholders to assess the model’s accuracy and identify any misclassifications accordingly. From the matrix, it could be realized that the majority of class entries are concentrated along the diagonal, which indicates the correct predictions for most classes. The vivid dark blue squares along the diagonal represent the number of correct predictions for each class. From a Class-Wise Accuracy point of view, 49 correct predictions, representing 96.1% accuracy was achieved for Class 1. In addition, 2 incorrect predictions were misclassified as other classes. Class 2 recorded 33 correct predictions, equivalent to 91.7% accuracy with only 3 misclassifications. Furthermore, class 3 correctly predicted 46 classes which is 97.9% accurate and only 1 misclassification. Moreover, class 4 has 42 correct predictions representing a 95.5% accuracy and 2 misclassifications respectively. Where class 5, class 17, class 18, and class 22, respectively recorded absolutely correct predictions with ideal (100%) accuracies, as this pattern continues for all classes, with accuracy rates ranging from 85.3% to 100%, the robustness of HybridPlantNet23 across various categories is established.

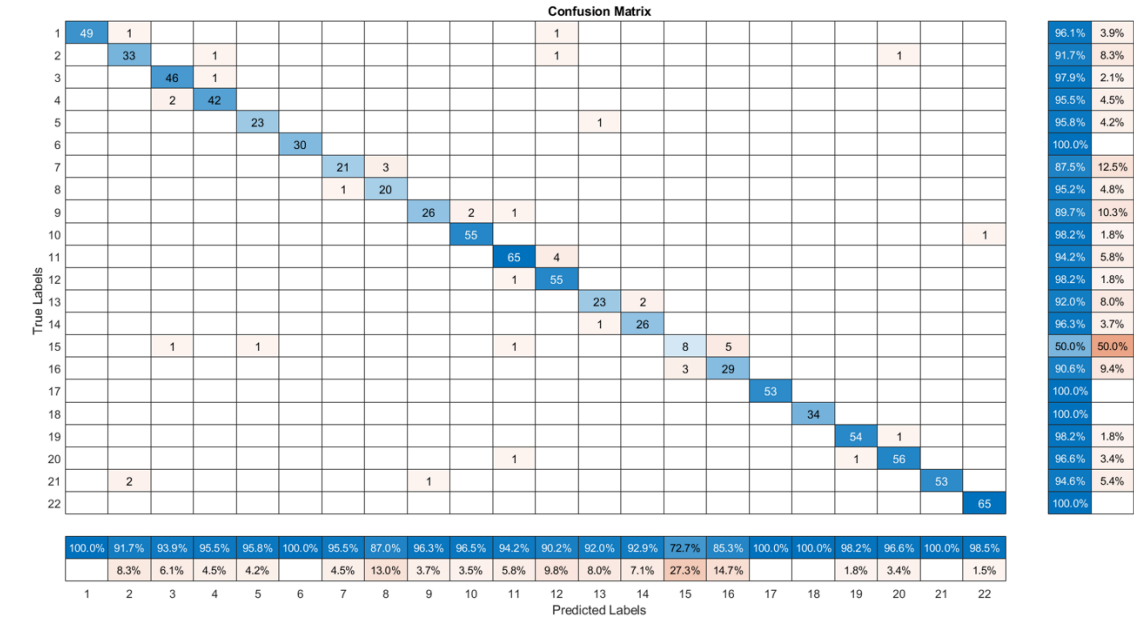


Figure 9. Confusion Matrix for HybridPlantNet23 Model.

Misclassifications of the model are relatively low and scattered across different classes. With a 50% accuracy rate and 50% misclassified classes, Class 15 stand as the most significant misclassification occurs with the model. This occurs as the model struggles with this particular class, likely due to overlapping features with other classes and, more precisely, inadequate training data of the class. Furthermore, Misclassified classes frequently have misclassifications with nearby classes, suggesting that these classes may be similar. This could be true for classes 8 and 7, with 85.3% and 87.0% classification accuracy, respectively.

The overall high accuracy for most classes, in essence, indicates the model’s effectiveness in distinguishing between different plant diseases. The small number of incorrect classifications points to potential areas for model improvement—possibly through dataset expansion or architectural optimization. For the lower accuracy observed in class 15, 8 and 7, respectively, increasing their sample sizes, examining their features, and possibly implementing additional preprocessing strategies to enhance feature differentiation is recommendable for improved classification results. Moreover, the high variability within Class 7 images, is making it challenging for the model to learn a consistent pattern thus the need for some better and more standardized guidelines in plant disease datasets preparation and utilization.

4.2. Comparative Analysis and Insights

With superior accuracy of 97.25%, VGG16 appears to be the most reliable model in terms of correct predictions as seen in Table 5. HybridPlantNet23 overtakes Darknet53 even though it falls short of VGG16 thus suggesting that: while ensemble technique improves performance, VGG16’s standalone capability is superior. Furthermore, VGG16 leads with the highest precision (96.30%), indicating the least false positives. Since HybridPlantNet23 model improves upon Darknet53’s precision, this showcases the advantage of combining features from both models. In addition, VGG16’s recall (96.96%) being the highest suggests its effectiveness in identifying true positives. HybridPlantNet23, as well, demonstrated substantial progress over Darknet53, thus indicating a better classification of actual disease cases. Even though HybridPlantNet23 improves over Darknet53 in terms of balanced performance, the F1-Score in VGG16 identifies it as the highest (96.55%). This could somewhat justify that the ensemble approach benefits from the strengths of both models.

From the results of this study, it might seem counterintuitive for single model (VGG16) to outperform HybridPlantNet23, since ensemble models are generally expected to utilize the powers of their component models to deliver higher performance. Technically, in the methodology and dataset preprocessing stage, the images were resized to the requirements of the respective pretrained models (VGG16 and Darknet53). While this effort ensures compatibility, it may introduce slight inconsistencies in feature extraction when the combined datastore is formed. This would make it necessary to select single models that are having equal or similar image size requirements thus avoiding potential inconsistencies in feature extraction. Furthermore, the ensemble model relies fully on the combined datastore, which must balance the feature extraction methods of both VGG16 and Darknet53. Hence, any misalignment or bias in feature representation may have the potential to affect the HybridPlantNet23’s performance.

For cases of ensemble modelling where one model significantly outperforms the other (as VGG16 does), the ensemble has high tendency to be dragged down by the weaker model, which in turn could displace the overall performance gain expected from ensembling. This could highlight vividly the nature and complexity of ensemble models like HybridPlantNet23. In addition, the ensemble model’s architecture plays a decisive role, such as how it weighs the classifications from each single model. On this note, it could be iterated that where the combination strategy of HybridPlantNet23 does not optimally utilize the strengths of VGG16 and Darknet53, the ensemble’s performance might not surpass VGG16.

Table 6. Some class performances over different models.

	Class 15	Class 8	Class 7
Darknet19	80	100	73.33
VGG16	75	95.2	87.5
Darknet53	31.2	85.7	87.5
HybridPlantNet23	50	85.3	87.0

One important point to note is the effect of data distribution and class imbalance from the dataset utilized in this study. For example, with the same dataset utilized in this study, the authors trained a Darknet19 model with a balanced dataset of 77 images each from all classes, choosing 77 as the

minimal number of images in the whole dataset which was a representation from class 15 (lemon) [34]. Despite variations with different models and data sizes, class 15 always poses a threat to the overall performance of the corresponding model (Table 6). The significant imbalance in the number of samples per class skews the learning process. Models tend to be biased towards classes with more samples, leading to higher performance for those classes and lower performance for minority classes like class 15. Despite being an ensemble model designed to leverage the strengths of VGG16 and Darknet53, HybridPlantNet23 also suffers from this problem. Class 15's performance is particularly poor, which drags down the overall model performance. Furthermore, the overall performance of the model in the former study is significantly lower than that attained herein, despite the fact that the former study used a 'balanced' dataset, whereas herein the whole dataset was used, implying that the total number of images used in this study is greater. This suggests that for effective deep learning applications, the total number of images in datasets is more significant than the class balance.

5. Conclusions

This study thoroughly assessed the results of a hybrid ensemble model called HybridPlantNet23, Darknet53, VGG16, and other deep learning models on a job of classifying plant species. It was clear from a thorough investigation of confusion matrices and a rigorous analysis of performance indicators including accuracy, precision, recall, and F1-score that VGG16 consistently beat Darknet53 and the ensemble model. Due to its better architecture, VGG16 was able to manage the dataset's complexity more skillfully, which raised its classification accuracy in the majority of classes. Class imbalance was still a problem, though, and it continued to have an impact on students' performance, especially in underrepresented classes like class 15.

The study's findings emphasize how crucial it is to solve problems with data distribution and class imbalance in order to enhance model performance. Although VGG16 showed resilience in both feature extraction and classification, the ensemble model's total performance did not improve as anticipated, where scientific insights were discussed. Future research should concentrate on dataset balancing methods such sophisticated data augmentation and synthetic data synthesis. Furthermore, investigating more complex architectures and utilizing transfer learning may improve model robustness and classification accuracy even more. These findings highlight the necessity of balanced datasets and strong model architectures, and they provide the groundwork for further advances in the classification of plant species using deep learning.

Author Contributions: Conceptualization, M.K. and S.E.; methodology, M.K., S.E., and T.C.A. A - validation, M.K., S.E., and T.C.A.; resources M.K., and S.E.; data curation, M.K. and S.E.; writing—original draft preparation, M.K., S.E., and T.C.A. All authors have read and agreed to the published version of the manuscript.

Funding: This study was supported by Firat University, ADEP Project No. 23. 23.

Data Availability Statement: The data can be shared up on request.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. M. Berners-Lee, C. Kennelly, R. Watson, C.N. Hewitt, Current global food production is sufficient to meet human nutritional needs in 2050 provided there is radical societal adaptation, *Elementa*. 6 (2018). <https://doi.org/10.1525/ELEMENTA.310/112838>.
2. S. Singh, B. Singh, A.P. Singh, Nematodes: A Threat to Sustainability of Agriculture, *Procedia Environ. Sci.* 29 (2015) 215–216. <https://doi.org/10.1016/j.proenv.2015.07.270>.
3. R.N. Strange, P.R. Scott, Plant Disease: A Threat to Global Food Security, *Annu. Rev. Phytopathol.* 43 (2005) 83–116. <https://doi.org/10.1146/annurev.phyto.43.113004.133839>.
4. J. Liu, X. Wang, Plant diseases and pests detection based on deep learning: a review, *Plant Methods*. 17 (2021) 1–18. <https://doi.org/10.1186/s13007-021-00722-9>.
5. R.K. Horst, Plant Diseases and Their Pathogens, *Westcott's Plant Dis. Handb.* (2001) 65–530. https://doi.org/10.1007/978-1-4757-3376-1_3.

6. R. Calderón, J.A. Navas-Cortés, P.J. Zarco-Tejada, Early detection and quantification of verticillium wilt in olive using hyperspectral and thermal imagery over large areas, *Remote Sens.* 7 (2015) 5584–5610. <https://doi.org/10.3390/rs70505584>.
7. K.P. Ferentinos, Deep learning models for plant disease detection and diagnosis, *Comput. Electron. Agric.* 145 (2018) 311–318. <https://doi.org/10.1016/j.compag.2018.01.009>.
8. A. Bouguettaya, H. Zarzour, A. Kechida, A.M. Taberkit, A survey on deep learning-based identification of plant and crop diseases from UAV-based aerial images, *Cluster Comput.* (2022) 1–21. <https://doi.org/10.1007/s10586-022-03627-x>.
9. I. Buja, E. Sabella, A.G. Monteduro, M.S. Chiriaco, L. De Bellis, A. Luvisi, G. Maruccio, Advances in plant disease detection and monitoring: From traditional assays to in-field diagnostics, *Sensors.* 21 (2021) 1–22. <https://doi.org/10.3390/s21062129>.
10. S.S. Valle, J. Kienzle, *Agriculture 4.0 - Agricultural robotics and automated equipment for sustainable crop production*, Food and Agriculture Organization of the United Nations, 2020. <https://www.fao.org/3/cb2186en/CB2186EN.pdf> (accessed August 17, 2023).
11. FAO, International Year of Plant Health 2020 | FAO | Food and Agriculture Organization of the United Nations, FAO. (2020). <https://www.fao.org/plant-health-2020/home/en/> (accessed August 14, 2022).
12. A. Ghobadpour, L. Boulon, H. Mousazadeh, A.S. Malvajerdi, S. Rafiee, State of the art of autonomous agricultural off-road vehicles driven by renewable energy systems, in: M. Becherif, H.S. Ramadan, M.T. Benchouia, M. Khaled, M. Ramadan, M. Hilaret (Eds.), *Emerg. Renew. ENERGY Gener. Autom., ELSEVIER SCIENCE BV, SARA BURGERHARTSTRAAT 25, PO BOX 211, 1000 AE AMSTERDAM, NETHERLANDS*, 2019: pp. 4–13. <https://doi.org/10.1016/j.egypro.2019.04.002>.
13. Y.-X. Yang, C. Wen, K. Xie, F.-Q. Wen, G.-Q. Sheng, X.-G. Tang, Face Recognition Using the SR-CNN Model, *Sensors.* 18 (2018) 4237. <https://doi.org/10.3390/s18124237>.
14. A. Mikołajczyk, M. Grochowski, Data augmentation for improving deep learning in image classification problem, in: 2018 Int. Interdiscip. PhD Work. IIPHDW 2018, Institute of Electrical and Electronics Engineers Inc., 2018: pp. 117–122. <https://doi.org/10.1109/IIPHDW.2018.8388338>.
15. M.N. Tibdewal, Y.M. Kulthe, A. Bharambe, A. Farkade, A. Dongre, Deep Learning Models for Classification of Cotton Crop Disease Detection, *Zeichen J.* 8 (2022).
16. R. Sachan, S. Kundra, A. Kumar Dubey, Paddy Leaf Disease Detection using Thermal Images and Convolutional Neural Networks, in: *Proc. Int. Conf. Comput. Intell. Sustain. Eng. Solut. CISES 2022*, Institute of Electrical and Electronics Engineers Inc., 2022: pp. 471–476. <https://doi.org/10.1109/CISES54857.2022.9844413>.
17. M.A.F. Azlah, L.S. Chua, F.R. Rahmad, F.I. Abdullah, S.R.W. Alwi, Review on techniques for plant leaf classification and recognition, *Computers.* 8 (2019). <https://doi.org/10.3390/computers8040077>.
18. P. A, B.K. S, D. Murugan, Paddy Leaf diseases identification on Infrared Images based on Convolutional Neural Networks, (2022). <https://doi.org/10.48550/arxiv.2208.00031>.
19. C.P. Lee, K.M. Lim, Y.X. Song, A. Alqahtani, Plant-CNN-ViT: Plant Classification with Ensemble of Convolutional Neural Networks and Vision Transformer, *Plants.* 12 (2023) 2642. <https://doi.org/10.3390/plants12142642>.
20. Y. Lu, S. Aslani, A. Zhao, A. Shahin, D. Barber, M. Emberton, D.C. Alexander, J. Jacob, A hybrid CNN-RNN approach for survival analysis in a Lung Cancer Screening study, *Heliyon.* 9 (2023) e18695. <https://doi.org/10.1016/j.heliyon.2023.e18695>.
21. Y. Chen, A Sentiment Analysis of Medical Text Based on Deep Learning, (2024). <http://arxiv.org/abs/2404.10503> (accessed June 26, 2024).
22. P.S. Thakur, T. Sheorey, A. Ojha, VGG-ICNN: A Lightweight CNN model for crop disease identification, *Multimed. Tools Appl.* 82 (2023) 497–520. <https://doi.org/10.1007/s11042-022-13144-z>.
23. S.H. Lee, C.S. Chan, P. Wilkin, P. Remagnino, Deep-plant: Plant identification with convolutional neural networks, in: *Proc. - Int. Conf. Image Process. ICIP*, 2015: pp. 452–456. <https://doi.org/10.1109/ICIP.2015.7350839>.
24. J. Su, C. Liu, M. Coombes, X. Hu, C. Wang, X. Xu, Q. Li, L. Guo, W.H. Chen, Wheat yellow rust monitoring by learning from multispectral UAV aerial imagery, *Comput. Electron. Agric.* 155 (2018) 157–166. <https://doi.org/10.1016/j.compag.2018.10.017>.
25. J.W. Tan, S.W. Chang, S. Abdul-Kareem, H.J. Yap, K.T. Yong, Deep Learning for Plant Species Classification Using Leaf Vein Morphometric, *IEEE/ACM Trans. Comput. Biol. Bioinforma.* 17 (2020) 82–90. <https://doi.org/10.1109/TCBB.2018.2848653>.
26. J. Hu, Z. Chen, M. Yang, R. Zhang, Y. Cui, A multiscale fusion convolutional neural network for plant leaf recognition, *IEEE Signal Process. Lett.* 25 (2018) 853–857. <https://doi.org/10.1109/LSP.2018.2809688>.
27. A. Kaya, A.S. Keceli, C. Catal, H.Y. Yalic, H. Temucin, B. Tekinerdogan, Analysis of transfer learning for deep neural network based plant classification models, *Comput. Electron. Agric.* 158 (2019) 20–29. <https://doi.org/10.1016/j.compag.2019.01.041>.

28. S. Anubha Pearline, V. Sathiesh Kumar, S. Harini, A study on plant recognition using conventional image processing and deep learning approaches, in: *J. Intell. Fuzzy Syst.*, IOS Press, 2019: pp. 1997–2004. <https://doi.org/10.3233/JIFS-169911>.
29. S.A. Riaz, S. Naz, I. Razzak, Multipath Deep Shallow Convolutional Networks for Large Scale Plant Species Identification in Wild Image, in: *Proc. Int. Jt. Conf. Neural Networks*, 2020. <https://doi.org/10.1109/IJCNN48605.2020.9207113>.
30. M. Litvak, S. Divekar, I. Rabaev, Urban Plants Classification Using Deep-Learning Methodology: A Case Study on a New Dataset, *Signals*. 3 (2022) 524–534. <https://doi.org/10.3390/signals3030031>.
31. Y. Arun, G.S. Viknesh, Leaf Classification for Plant Recognition Using EfficientNet Architecture, in: *Proc. IEEE 2022 4th Int. Conf. Adv. Electron. Comput. Commun. ICAECC 2022*, 2022. <https://doi.org/10.1109/ICAEECC54045.2022.9716637>.
32. A. Beikmohammadi, K. Faez, A. Motallebi, SWP-LeafNET: A novel multistage approach for plant leaf identification based on deep CNN, *Expert Syst. Appl.* 202 (2022). <https://doi.org/10.1016/j.eswa.2022.117470>.
33. S.S. Chouhan, A. Kaul, U.P. Singh, S. Jain, A Database of Leaf Images: Practice towards Plant Conservation with Plant Pathology, (2020). <https://doi.org/10.17632/hb74ynkjc4>.
34. M. Kabir, F. Unal, T.C. Akinci, A.A. Martinez-Morales, S. Ekici, Revealing GLCM Metric Variations across a Plant Disease Dataset: A Comprehensive Examination and Future Prospects for Enhanced Deep Learning Applications, *Electron. 2024*, Vol. 13, Page 2299. 13 (2024) 2299. <https://doi.org/10.3390/ELECTRONICS13122299>.
35. K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, in: *3rd Int. Conf. Learn. Represent. ICLR 2015 - Conf. Track Proc.*, International Conference on Learning Representations, ICLR, 2015. <https://arxiv.org/abs/1409.1556v6> (accessed June 28, 2024).
36. A. Ahmad, A. El Gamal, D. Saraswat, Toward Generalization of Deep Learning-Based Plant Disease Identification Under Controlled and Field Conditions, *IEEE Access*. 11 (2023) 9042–9057. <https://doi.org/10.1109/ACCESS.2023.3240100>.
37. C. Bi, S. Xu, N. Hu, S. Zhang, Z. Zhu, H. Yu, Identification Method of Corn Leaf Disease Based on Improved Mobilenetv3 Model, *Agronomy*. 13 (2023) 300. <https://doi.org/10.3390/agronomy13020300>.
38. G. Wang, Y. Sun, J. Wang, Automatic Image-Based Plant Disease Severity Estimation Using Deep Learning, *Comput. Intell. Neurosci.* 2017 (2017). <https://doi.org/10.1155/2017/2917536>.
39. F. Sarwar, A. Griffin, S.U. Rehman, T. Pasang, Detecting sheep in UAV images - ScienceDirect, *Comput. Electron. Agric.* 187 (2021) 106219. <https://doi.org/10.1016/j.compag.2021.106219>.
40. J.H. Won, D.H. Lee, K.M. Lee, C.H. Lin, An Improved YOLOv3-based Neural Network for De-identification Technology, in: *34th Int. Tech. Conf. Circuits/Systems, Comput. Commun. ITC-CSCC 2019*, Institute of Electrical and Electronics Engineers Inc., 2019. <https://doi.org/10.1109/ITC-CSCC.2019.8793382>.
41. H. Wang, F. Zhang, L. Wang, Fruit classification model based on improved darknet53 convolutional neural network, in: *Proc. - 2020 Int. Conf. Intell. Transp. Big Data Smart City, ICITBS 2020*, Institute of Electrical and Electronics Engineers Inc., 2020: pp. 881–884. <https://doi.org/10.1109/ICITBS49701.2020.00194>.
42. N. Rachburee, W. Punlumjeak, An assistive model of obstacle detection based on deep learning: YOLOv3 for visually impaired people, *Int. J. Electr. Comput. Eng.* 11 (2021) 3434–3442. <https://doi.org/10.11591/ijece.v11i4.pp3434-3442>.
43. M.H. Hamidon, T. Ahamed, Detection of Tip-Burn Stress on Lettuce Grown in an Indoor Environment Using Deep Learning Algorithms, *Sensors*. 22 (2022) 7251. <https://doi.org/10.3390/s22197251>.
44. S. Vasavi, N.K. Priyadarshini, K. Harshavaradhan, Invariant Feature-Based Darknet Architecture for Moving Object Classification, *IEEE Sens. J.* 21 (2021) 11417–11426. <https://doi.org/10.1109/JSEN.2020.3007883>.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.