



## Article

# Tomato Leaf Disease Classification via Compact Convolutional Neural Networks with Transfer Learning and Feature Selection

Omneya Attallah

Department of Electronics and Communications Engineering, College of Engineering and Technology, Arab Academy for Science, Technology and Maritime Transport, Alexandria 1029, Egypt; o.attallah@aast.edu

**Abstract:** Tomatoes are one of the world's greatest valuable vegetables and are regarded as the economic pillar of numerous countries. Nevertheless, these harvests remain susceptible to a variety of illnesses which can reduce and destroy the generation of healthy crops, making early and precise identification of these diseases critical. Therefore, in recent years, numerous studies have utilized deep learning (DL) models for automatic tomato leaf illness identification. However, many of these methods are based on a single DL architecture that needs a high computational ability to update these hyperparameters leading to a rise in the classification complexity. In addition, they extracted large dimensions from these networks which added to the classification complication. Therefore, this study proposes a pipeline for the automatic identification of tomato leaf diseases utilizing three compact convolutional neural networks (CNNs). It employs transfer learning to retrieve deep features out of the final fully connected layer of the CNNs for more condensed and high-level representation. Next, it merges features from the three CNNs to benefit from every CNN structure. Subsequently, it applies a hybrid feature selection approach to select and generate a comprehensive feature set of lower dimensions. Six classifiers are utilized in the tomato leaf illnesses identification procedure. The results indicate that the K-nearest neighbor and support vector machine have attained the highest accuracy of 99.92% and 99.90% using 22 and 24 features only. The experimental results of the proposed pipeline are also compared with previous research studies for tomato leaf diseases classification which verified its competing capacity.



**Citation:** Attallah, O. Tomato Leaf Disease Classification via Compact Convolutional Neural Networks with Transfer Learning and Feature Selection. *Horticulturae* **2023**, *9*, 149. <https://doi.org/10.3390/horticulturae9020149>

Academic Editor: Jin Hee Lim

Received: 14 December 2022

Revised: 8 January 2023

Accepted: 16 January 2023

Published: 22 January 2023



**Copyright:** © 2023 by the author. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Agriculture has traditionally been the principal source of income for the majority of African and Asian countries. Agriculture's comprehensive commercialization has had a significant impact on our environment. Plant disease detection is one of the most pressing issues in agriculture. Early disease detection aids in the prevention of disease spread among other plants, resulting in considerable economic losses. The consequences of plant illness vary from minor manifestations to the loss of entire plantations, which has a significant impact on the agricultural economy [1].

Tomatoes are among the most substantial and commonly consumed crops on the globe [2]. Based on the latest statistics, the world's tomato production is approximately greater than 180 million metric tons, exporting USD 8.81 billion [3]. Nevertheless, tomato production is declining due to the crop's susceptibility to various diseases [4]. Tomato leaf diseases are a major cause of the loss of tomato crops and the economic losses of farmers. The detection of tomato leaf disease is inseparably linked to agricultural financial activity. It is critical to quickly identify tomato leaf diseases and implement appropriate control actions to guarantee tomato production and farmer profitability. Conventional disease detection methods necessitate a manual investigation of diseased leaves using visual information or chemical analysis of affected regions, which is a time-consuming

procedure and can result in low detection accuracy and poor reliability due to human mistakes [5]. Further exacerbated by farmers' lack of professional experience and the absence of agricultural trained experts who can identify diseases in remote places impeding overarching agricultural production. Irresponsibility in this respect presents a serious threat to global food security whilst also allowing significant losses for tomato production stakeholders. Early disease detection and identification utilizing automated tools and techniques available to farmers could indeed help alleviate all of the issues raised [6].

Inspired by the great success of artificial intelligence (AI) technologies, such as the traditional machine learning (ML) algorithms and the latest deep learning (DL) methods in several domains including medical image analysis [7–10], human disease identification and classification [11–16], healthcare [17–19], industry [20], several AI methods have been adopted to automate the detection and identification of tomato leaf disease in precision agriculture. Numerous resolutions have been developed for plant disease identification using traditional machine-learning approaches. Although conventional ML methods depend greatly on the hand-crafted features provided to the classifiers, these features are obtained manually by the expert, making these methods costly and laborious. On the other hand, DL can avoid these limitations by automatically extracting deep features directly from images reaching better accuracies than traditional ML approaches. Among DL techniques that have been widely used in plant disease classification is the convolutional neural network (CNN) [21–23]. The invention of the CNN model raised the prospect of disease identification in tomato plants based on variations in leaf appearance [24]. As a result, treatment can begin immediately depending on the type of illness. There are several architectures for CNNs, and many of them contain a large number of deep layers with an extremely large number of parameters; therefore, they need a high computational ability to update these hyperparameters leading to a rise in the classification complexity. Transfer learning (TL) is a popular ML methodology that allows for the reuse of an effective DL model to tackle one issue as the preliminary step for another problem in a related field. It has drastically decreased the need for massive computational resources and model construction period [25]. The objective of the current study is to introduce a framework based on lightweight CNNs and TL to detect and classify tomato leaf diseases. The proposed framework is based on three DL models with fewer deep layers and less parameters. Furthermore, the framework blends deep features of the three CNNs. It also performs a hybrid feature selection (FS) procedure to decrease the number of features utilized for the diagnosis process using ML classifiers.

## 2. Literature Review

The amount and quality of tomato plant products are seriously impacted by the timely identification of tomato plant diseases. Earlier studies conducted for tomato disease classification used various classifiers fed with hand-crafted features based on the texture, color, or shape features of tomato leaves [26]. These studies primarily concentrated on a small number of diseases using extreme feature engineering and were frequently limited to restricted environments. Due to the sensitivity of the extracted features to the surroundings located in leaf pictures, methods of ML depended on thorough preprocessing steps, such as manual region of interest cropping, color alteration, resizing, background elimination, and filtering for effective feature extraction. Due to the greater sophistication caused by these preprocessing procedures, conventional ML methodologies could only classify a few diseases from a small dataset, failing to generalize to bigger sizes [3]. The results of a considerable part of previous works were not comparable since they were mainly accomplished on self-curated datasets with few images. This problem was greatly ameliorated when the PlantVillage dataset [27], which contained 54,309 images of 14 different crop species and 26 diseases, was introduced. With the creation of large datasets, such as PlantVillage, recently, several researchers have used DL algorithms, such as CNN for plant leaves disease detection and classification as they are requiring large data to function well.

Recent TL-based approaches using the PlantVillage dataset for leaf disease classification have investigated the performance of different pre-trained CNN models using various hyperparameters in order to reduce reliance on hand-crafted features while improving classification accuracy with large datasets. Among these studies, the authors of [28] presented a DL model for detecting tomato disease using the conditional generative adversarial network (C-GAN) to create artificial photos of the tomato plant leaf. Next, a DenseNet-121 CNN is trained on artificial and real photos using TL to classify tomato leaf images into five, seven, and ten disease subgroups. For tomato leaf image classification into five classes, seven classes, and ten classes, the authors achieved an accuracy of 99.51%, 98.65%, and 97.11%, respectively. Similarly, the study [29] proposed an augmentation approach to generate synthetic data fed to boundary-aware refined network (BARNet) to classify tomato leaf disease into four categories reaching 98.75%.

On the other hand, the research article [30] created a lightweight ResNet-20 CNN with different attention modules to improve model performance in identifying 11 tomato leaf diseases. The highest accuracy of 99.69% was attained with the convolutional block attention module. Similarly, the study [3] designed a compact CNN based on MobileNet and TL to distinguish 10 types of tomato leaf illness achieving an accuracy of 99.3%. Moreover, the research article [31] created a compact customized CNN to distinguish between 10 class labels of tomato leaf diseases reaching 99.7% accuracy, while the study [32] developed an adapted version of Xception CNN and utilized TL to identify 10 categories of tomato leaf diseases of the PlantVillage dataset. The authors employed different optimization algorithms to train the CNN to reach 99.5% accuracy with the ADAM optimizer. Other studies [33,34] employed several CNN architectures individually to distinguish among different tomato leaf diseases. The authors of [33] employed LeNet, while ResNet, VGG16, and Xception were used to classify tomato leaf images into ten categories, with VGG16 achieving a maximum accuracy of 99.25%. Comparably, the authors of [34] compared the performance of five CNN structures involving ResNet-18, ResNet-50, Inception, GoogleNet, and AlexNet and achieved the maximum accuracy of 99.36% using GoogleNet.

Other studies compared traditional ML algorithms with DL methods. An example of these studies is [35]. The authors used various methods to manually retrieve disease features for ML algorithm implementation. Texture features, such as local binary pattern (LBP) and grey level co-occurrence matrix (GLCM) methods, as well as color features, such as color moment and color histogram methods, are included. SVM and KNN classifiers were fed by these features. The authors used EfficientNet-B0, AlexNet, ResNet-34, VGG16, and MobileNetV2 to implement DL. The findings revealed that DL approaches outperformed ML methods. While other research articles utilized TL to extract deep features from a CNN and then employed ML classifiers to classify tomato leaf illnesses. The literature demonstrated that merging DL algorithms with TL and ML classifiers could boost diagnostic performance. Among these articles, the study [2] combined deep features extracted from MobileNet and NASNet CNNs, and then used principal component analysis (PCA) to diminish the dimensionality of features. Finally, these reduced features were inputs to various ML classifiers where the peak accuracy of 97% is attained with multinomial logistic regression. Furthermore, the authors of [36] created a new feature extraction technique that employed attention-based dilated CNN to retrieve the most important features. Bilateral filtering and Otsu segmentation were used to preprocess the images. The CGAN model is then used to produce synthesized images from the preprocessed images. Finally, pre-processed attributes were merged and classified using a logistic regression (LR) classifier achieving an accuracy of 96.6%, whereas the article [37] implemented a modified version of AlexNet CNN to classify 10 classes of tomato leaf diseases reaching an accuracy of 98%.

Despite the interesting performance attained by these previous methods for tomato leaf disease classification, they still suffer from several shortcomings. First, most of them conduct the classification procedures via individual CNN architecture or deep features obtained from a single CNN; however, merging deep features from several DL models is capable of boosting diagnostic accuracy [7,11,38–40]. Second, some of them are based on

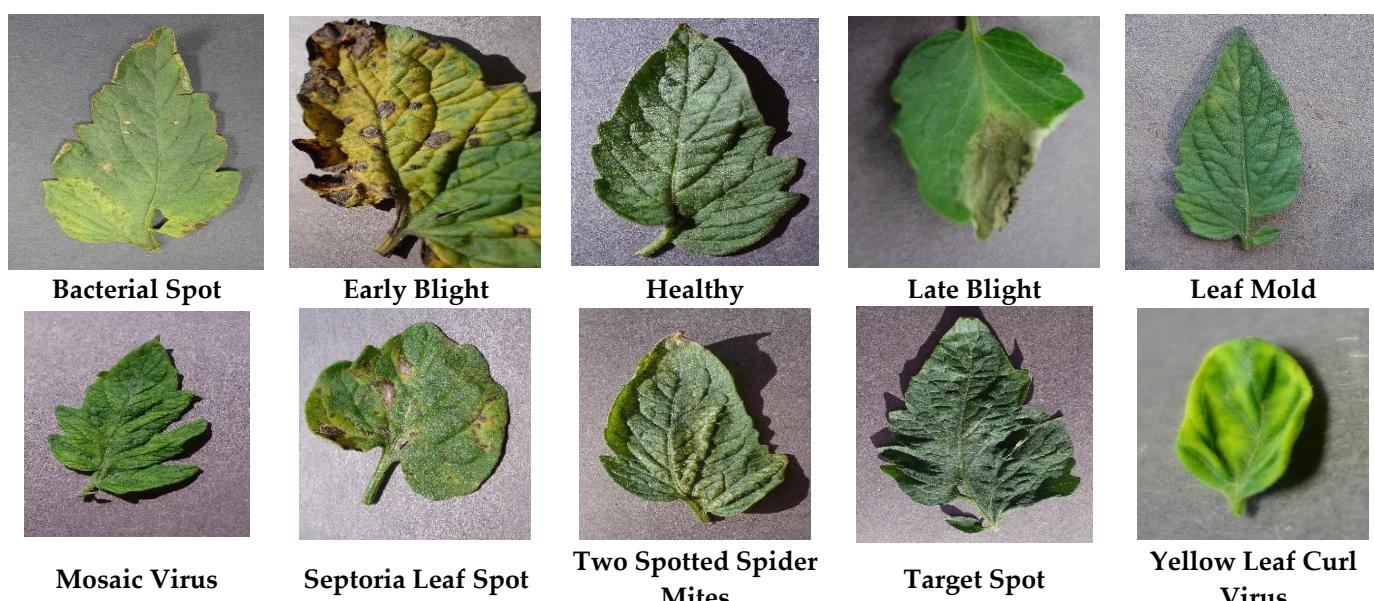
deep features of large dimensions fed to ML classifiers which increase the computational load and duration of the classification procedure. To our knowledge, none of them used FS to choose the highly valuable attributes; thus, lessening the dimensionality of features which correspondingly diminishes the complexity and time of the classification. Third, the majority of them are based on CNN structures that involve a large number of deep layers with an extremely large number of parameters, necessitating a high computational ability to modify these hyperparameters, resulting in an increase in classification ambiguity. To overcome these limitations, this study proposes a novel efficient pipeline for tomato leaf disease classification based on multiple compact CNNs. The novelty and contributions are summarized as follows:

- Employing three compact CNNs with dissimilar structures involving ResNet-18, ShuffleNet, and MobileNet to extract deep features via TL.
- Retrieving deep features out of the final fully connected (FC) layer for each CNN before the softmax layer and obtaining a lower number of deep features from this FC layer compared to earlier layers.
- Blending deep features obtained from the three CNNs to merge benefits of every CNN construction.
- Utilizing a hybrid FS approach for selection among merged deep features diminishes their dimensionality.
- Using three search policies to choose among these combined features results in selecting only the most significant.

### 3. Materials and Methods

#### 3.1. Tomato Diseases Dataset Acquisition

Currently, the PlantVillage dataset is considered as one of the globe's biggest accessible repositories of proficiently curated plant leaf photos for the identification of diseases. It includes 54,309 pictures of normal and diseased leaves from fourteen harvests that were classified by plant pathology professionals. Only tomato crops were considered in this study, which included 9 different leaf illness classes and one normal class, with a sum of 16,011 photos employed in the experiments. This dataset contains samples of leaves infected with various diseases to varying degrees. Figure 1 shows a picture from every class label of tomato leaf illness in the database. Table 1 displays various dataset information, such as the class name of the tomato illnesses and the quantity of pictures for each class.



**Figure 1.** Instances of tomato leaf pictures of the PlantVillage dataset.

**Table 1.** The distribution of pictures among each class label of the PlantVillage dataset.

Class Label	Number of Images
Bacterial Spot	2127
Early Blight	1000
Healthy	1591
Late Blight	1909
Leaf Mold	952
Mosaic virus	373
Septoria Leaf Spot	1771
Two Spotted Spider Mites	1676
Target Spot	1405
Yellow Leaf Curl Virus	3209

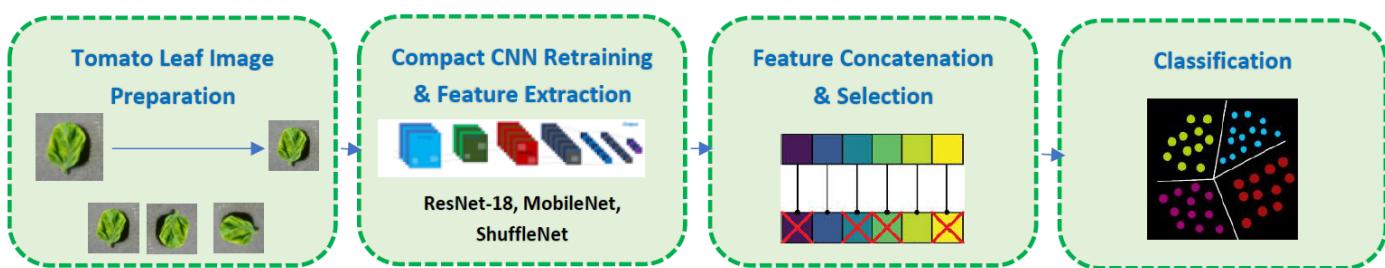
Tomato crops are susceptible to a variety of maladies, including late blight, bacterial spot, leaf mold, early blight, leaf curl, mosaic virus, spider mites, yellow leaf curl virus, and Septoria leaf spot, at various stages of development, which can be caused primarily by weather conditions or provoked by external environmental influences. These illnesses that arise in the plant's leaves have a massive effect on tomato productivity and production [41]. Early blight is a virus that infects the tomato plant and fruit, causing dark circles of concentric rings to appear on older leaves. The infected leaves die prematurely, and the fruit is also harmed. This condition is typically soil-borne and evolves throughout wet weather. Furthermore, the late blight condition is among the most destructive types and had been linked to the Irish famine in the past. It destroys other plants at a rapid rate, causing disfluent grey spots on plants with oily surfaces. Within the winter, the spots have a white periphery, allowing the leaves to become papery and eventually fall out. Moreover, the Septoria leaf spot exhibits effects similar to late blight in that the mature leaves are initially infected before spreading to other areas of the plant. Bacterial Spot is a fungal illness that causes tomatoes to wilt. The disease begins as tiny-shaped spots on fruits that expand larger in size over time. The disorder is soil-borne, and therefore any overripe tomato that comes into contact with the soil becomes susceptible to infection [42]. Similarly, tomato yellow leaf curl virus is a deadly virus that causes tomato illness, which is characterized by plant stunted growth and noticeable wilt leaves that wrap upward. As the leaves age, they get to be wrinkled and fragile. The size of the internodes and nodes has been considerably lowered. Affected leaves appear dull and have more lateral branches, providing them with a bushy look. The affected leaves continue to be stunted. A further viral disease is the tomato mosaic virus. Plants are hampered with mosaic or Fernleaf-like signs when contaminated, based on the strain and maturity level of the plants. Spider Mites is another plant-threatening plant disease. Due to feeding injury, tomato plants struck by mites frequently exhibit a mottled or flecked dull visual effect on the top leaf surfaces. Thereafter, the leaves turn yellow and fall. Vast numbers generate a noticeable thread that can cover the leaves entirely [43].

The aforementioned crop diseases pose serious risks to the agricultural production of food for the subsistence and preservation of the world's most important species [42]. When leaf maladies are detected earlier, provisions can be taken to boost tomato production and avoid losses [41]. However, these diseases require the implementation of infrastructure and computer programs to automate their detection, prevention, and diagnosis [42]. The development of deep learning algorithms for automatic plant disease diagnosis has improved malady accuracy of classification. These methods can be used to increase tomato production in both size and quality. Lately, they are employed to identify diseases that affect plants in the agricultural area using only cameras to capture the image. These techniques significantly simplify the task of tomato leaf identification [41]. Therefore, in this study, an

automated tool based on several deep learning models is presented to detect and diagnose these threatening tomato leaf diseases in order to enhance tomato yield and production.

### 3.2. Proposed Tomato Leaf Disease Classification Pipeline

The proposed pipeline for tomato leaf disease classification is based on four consecutive steps which are tomato leaf image preparation, compact CNN training and feature extraction with TL, feature concatenation and selection, and finally classification. Initially, tomato leaf photos are augmented, and their dimensions are changed in the tomato leaf image preparation step. Thereafter, three compact CNNs including ResNet-18, ShuffleNet, and MobileNet are retrained with TL. Next, TL is utilized to extract deep features from the three CNNs. These features are then concatenated, and a hybrid FS approach is applied to these features to select a reduced set of significant features. Finally, six ML classifiers are employed to classify tomato leaf photos into one of the 10 classes of tomato images. Figure 2 demonstrates the workflow of the proposed pipeline.



**Figure 2.** The workflow of the proposed pipeline for tomato leaf image classification.

#### 3.2.1. Tomato Leaf Image Preparation

Tomato leaf images of the PlantVillage dataset dimensions are primarily modified to be acceptable to feed the input layers of the three compact CNNs. As these layers only admit images of specific sizes; thus, the new size of the images is  $224 \times 224 \times 3$  for the three CNNs. Thereafter, these images are augmented using various data augmentation approaches. The effectiveness of CNN models is strongly dependent upon the training dataset. On adequately large datasets, these models show improved results and high generalizability. The datasets currently available for tomato plant disease typically lack sufficient images in a variety of conditions, which is required for developing high-accuracy models. Given the small size of the dataset, the model could overfit and perform poorly on real-world test data. Hereafter, numerous data augmentation methods, such as flipping, rotation, shearing, and scaling are being employed to boost the amount of images utilized to train the CNNs; thus, improving training performance and preventing overfitting.

#### 3.2.2. Compact CNNs Retraining and Feature Extraction with TL

As previously mentioned, three compact CNNs are used in the experiments including ResNet-18, ShuffleNet, and MobileNet. These CNNs were formerly trained on the large ImageNet dataset. Then, these pre-trained models are alerted in order to be used for classifying the tomato leaf images of the PlantVillage dataset. First, TL is employed to adjust the number of FC layers to correspond to ten which is the number of tomato image categories of the PlantVillage dataset rather than the 1000 class labels of ImageNet. Next, a few hyperparameters are tweaked which is going to be illustrated later in the experiment setting section. Thereafter, these CNNs are retrained with the tomato leaf images of the PlantVillage dataset. Subsequently, when the retraining procedure is completed, deep features are retrieved using TL from the last FC layer before the softmax classification layer. The CNN consists of various deep layers and the initial layers learn basic components from an image. On the other hand, later deep layers acquire high-level detailed features from the image. For this reason, the last FC layer prior to the softmax layer is selected to extract the deep features. One more reason is that the deeper layer acquires features with lower

dimensions and since we aim to propose an efficient pipeline with a few number of features and compact CNNs with fewer deep layers and many parameters, the FC layer is the right option. The length of the feature vector obtained from each CNN is 10.

### 3.2.3. Feature Concatenation and Selection

Features retrieved in the previous step are concatenated to merge the benefits of features obtained from each CNN construction. Thereafter, a hybrid FS approach is applied to the merged features to select the most noteworthy features that have a greater impact on the classification performance and reduce the merged features' dimensionality. FS is a critical step in selecting the most important attributes located in the feature space in order to narrow its dimension, which enhances detection ability and prevents overfitting [44–47]. There are three types of FS methods: Filter, wrapper, and hybrid [48]. In the former filter FS methodology, attributes are ranked according to a specific criterion. This method is straightforward and speedy, but it is independent of the classification procedure. Alternatively, the wrapper methods take into consideration the FS procedures during the classification process but at the cost of speed. Filter and wrapper methods are combined in hybrid FS. This type brings together the advantages of the preceding FS types. As a result, in the present study, a hybrid FS method is proposed and utilized.

The proposed pipeline's hybrid FS step unites the GR-Filter FS method along with the wrapper FS approach, which uses multiple search methods. GR is a bias-reduction alteration to the information gain FS approach. When selecting an attribute, GR considers the number and size of branches. It corrects the information gained by taking into account the intrinsic information of a split. Intrinsic information provides the entropy required to identify which branch an instance belongs to. The value of an attribute decreases as the amount of intrinsic information increases [49]. The steps for calculating the gain ratio are as follows. First, calculate the entropy using Equation (1):

$$\text{Entropy}(S) = \sum_i^C -p_i \log_2 p_i \quad (1)$$

where  $C$  is the number of the class label,  $p_i$  is the ratio between the amount of instances in class  $i$  and the amount of instances in the entire dataset, and  $S$  is the sample or instance.

Then, calculate the information gain value utilizing Equation (2):

$$\text{Gain}(S, A) = \text{Entropy} - \sum_{v \in \text{values}(A)} \frac{|S_v|}{S} \text{entropy}(S_v) \quad (2)$$

Thereafter, calculate the value of split information using Equation (3):

$$\text{SplitInfo}(S, A) = -\frac{\sum_{j=1}^c |S_j|}{S} \log_2 \frac{|S_j|}{S} \quad (3)$$

Finally, the  $GR$  is determined via Equation (4):

$$GR = \frac{\text{Gain}(S, A)}{\text{SplitInfo}(S, A)} \quad (4)$$

Using the  $GR$  filter FS, the proposed FS method first sorts features acquired from the triple DL architectures. Then, within the wrapper FS approach, it uses this ranking to mentor the three attribute search techniques. Backward, forward, and bidirectional search methods are examples of these search approaches. The first method begins by including the entire number of attributes and afterward recursively disregards attributes with lesser scores. Conversely, the forward methodology starts with the feature with the highest score and keeps adding the next attributes one at a time. On the other hand, the bidirectional strategy switches among forward/backward approaches. It is worth noting that for each

of the triple schemes, only the attributes that are capable of boosting the classification performance are retained, whereas others are removed.

### 3.2.4. Classification

Six ML classifiers are used to classify tomato leaves into ten classes. These classifiers include Naïve Bayes (NB), K-nearest neighbor (KNN), decision tree (DT), the linear discriminant classifier (LDA), support vector machine (SVM), and quadratic discriminant analysis (QDA). For the KNN classifier, the number of neighbors is 3 and the distance metric is Euclidean. On the other hand, for the SVM classifier, a linear kernel function is utilized. For the DT classifier, the J48 algorithm is implemented. The five-cross validation approach is utilized to access the performance of the proposed pipeline.

The classification procedure of the proposed pipeline is implemented in three contexts. Each classifier is fed with the features acquired from each CNN separately in the first context. Whereas, in the second context, concatenated features retrieved from the three CNNs are employed as inputs to the six ML classifiers. Finally, in the third context, features selected utilizing the three search methodologies of the hybrid FS approach are utilized to train the six ML classifiers.

## 4. Pipeline Evaluation and Networks' Hyperparameters Setting

### 4.1. Pipeline Results Evaluation

Precision, specificity, F1-score, accuracy, sensitivity, and Mathew correlation coefficient (MCC) are among the evaluation criteria used to assess the efficiency of the proposed pipeline. These metrics are calculated using the formulae that follow (Equations (5)–(10)). Furthermore, the receiving operating characteristic (ROC) and confusion matrix are calculated.

$$\text{Accuracy} = \frac{TP + TN}{TN + FP + FN + TP} \quad (5)$$

$$\text{Sensitivity} = \frac{TP}{TP + FN} \quad (6)$$

$$\text{Precision} = \frac{TP}{TP + FP} \quad (7)$$

$$\text{MCC} = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \quad (8)$$

$$\text{F1-Score} = \frac{2 \times TP}{(2 \times TP) + FP + FN} \quad (9)$$

$$\text{Specificity} = \frac{TN}{TN + FP} \quad (10)$$

where true positive ( $TP$ ) denotes the fraction of instances correctly identified as positive, false negative ( $FN$ ) denotes the fraction of samples incorrectly classified as negative, true negative ( $TN$ ) indicates the number of cases that are correctly interpreted as negative, and false positives ( $FP$ ) represent samples that have been misclassified as positives.

### 4.2. Networks Hyperparameters Setting

For retraining the CNNs, the frequency of the epochs is set to 20. Furthermore, the learning rate and minimum batch size have been adapted to 0.001 and 10. In addition, the frequency of validation is adjusted to 1120 in order to compute the CNNs output error one time per epoch. The three CNNs were constructed using stochastic gradient descent with a momentum technique. Other hyperparameters remained constant. The introduced pipeline is executed utilizing MATLAB R2020a and Weka Data Mining Tool [50].

## 5. Results

The experimental performance of the three classification contexts of the proposed pipeline is displayed and discussed in this section. Initially, the classification results

for the six ML classifiers constructed with each feature set extracted from every CNN individually are presented. Next, the classification results of the ML classifier fed with features obtained after concatenation of the attributes attained from the three CNNs are demonstrated. Lately, the ML results accomplished with the features selected using the three search methodologies are illustrated.

### 5.1. Context 1 Classification Performance

The results achieved by the six ML of context 1 are displayed in Table 2. Table 2 indicates that using ResNet-18 deep features, the accuracy varies between 97.3% and 99.34%, sensitivity ranges from 97.3% to 99.31%, specificity fluctuates from 97.0% to 99.9%, precision alters between 97.4% and 99.3%, the F1-score changes between 97.3% and 99.3%, and the MCC ranges from 96.9% to 99.3%. Herein, the highest performance is attained with KNN and linear SVM classifiers and the least performance is achieved with the DT classifier.

**Table 2.** The classification results (%) obtained by the individual deep features of ResNet-18, ShuffleNet, and MobileNet.

	Accuracy	Sensitivity	Specificity	Precision	F1-Score	MCC
<b>ResNet-18 Features</b>						
<b>NB</b>	98.5	98.2	99.8	98.5	98.5	98.3
<b>LDA</b>	97.3	97.3	97.0	97.4	97.3	96.9
<b>QDA</b>	98.9	98.9	99.9	98.9	98.9	98.8
<b>LSVM</b>	99.34	99.3	99.9	99.3	99.3	99.3
<b>KNN</b>	99.34	99.3	99.9	99.3	99.3	99.3
<b>DT</b>	99.05	99.1	99.1	99.1	98.9	99.6
<b>ShuffleNet Features</b>						
<b>NB</b>	99.21	99.2	99.9	99.2	99.2	99.1
<b>LDA</b>	99.33	99.3	99.9	99.3	99.3	99.2
<b>QDA</b>	99.48	99.5	99.9	99.5	99.5	99.4
<b>LSVM</b>	99.53	99.5	99.9	99.5	99.5	99.5
<b>KNN</b>	99.60	99.6	100	99.6	99.6	99.6
<b>DT</b>	99.17	99.2	99.9	99.2	99.2	99.1
<b>MobileNet Features</b>						
<b>NB</b>	99.28	99.3	99.9	99.3	99.3	99.2
<b>LDA</b>	99.24	99.3	99.9	99.3	99.3	99.2
<b>QDA</b>	99.47	99.5	99.9	99.5	99.5	99.4
<b>LSVM</b>	99.56	99.6	99.9	99.6	99.6	99.5
<b>KNN</b>	99.67	99.7	100	99.7	99.7	99.6
<b>DT</b>	99.18	99.2	99.9	99.2	99.2	99.1

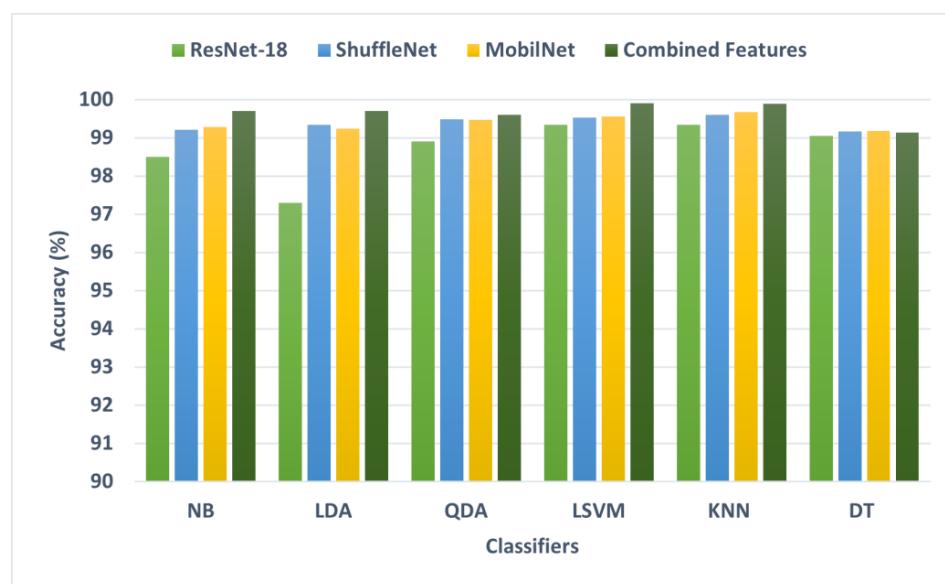
On the other hand, the ML trained with ShuffleNet features attained higher performance as the accuracy diverges between 99.17% and 99.60%, sensitivity, precision, and F1-score varies from 99.2% to 99.6%, and the MCC alters between 99.1% and 99.6%. Similarly, the KNN and linear SVM realize the best performance while DT shows the least performance.

Similarly, the performance achieved by the six ML constructed with MobileNet deep features is greater than the ResNet-18 deep features. This is due to the fact that the accuracy differs between 99.18% and 99.67%, sensitivity, precision, and F1-score fluctuate from 99.2%

to 99.7%, and the MCC changes between 99.1% and 99.6%. Moreover, the KNN and linear SVM accomplish the maximum performance while DT has the lowest performance.

### 5.2. Context 2 Classification Performance

The accuracy after the feature concatenation step is compared with that attained using individual deep feature sets of the three CNNs as shown in Figure 3. The results in Figure 3 verify that combining features obtained from different CNN architectures are capable of boosting the classification performance. This is clear as the classification accuracies attained after feature concatenation have increased with respect to individual deep feature sets for all classifiers except for the DT classifier, which is almost the same. These classification accuracies are 99.7%, 99.7%, 99.6%, 99.9%, 99.89%, and 99.13% for the NB, LDA, QDA, linear SVM, KNN, and DT classifiers. Table 3 shows other performance metrics attained with the six ML classifiers fed with the combined features. The table indicates the peak sensitivity, specificity, precision, F1-score, and MCC of 99.9%, 100%, 99.9%, 99.9%, and 99.9% reached by the KNN. Similar performance is accomplished by the linear SVM classifier, but the MCC is 99.8%. Note that these results are attained with 30 deep features.



**Figure 3.** The classification accuracy (%) of the six ML classifiers after feature concatenation.

**Table 3.** The classification results (%) for the six ML classifiers fed with the combined features after the concatenation step.

	Sensitivity	Specificity	Precision	F1-Score	MCC
<b>NB</b>	99.7	100	99.7	99.7	99.7
<b>LDA</b>	99.7	100	99.7	99.7	99.7
<b>QDA</b>	99.6	99.9	99.6	99.6	99.5
<b>LSVM</b>	99.9	100	99.9	99.9	99.8
<b>KNN</b>	99.9	100	99.9	99.9	99.9
<b>DT</b>	99.1	99.9	99.1	99.1	99.0

### 5.3. Context 3 Classification Performance

The results attained after the three search methodologies of the hybrid FS approach are demonstrated in this section. Table 4 reveals a comparison between the results obtained using each search strategy for each classifier and the number of features used in each methodology. The table specifies that FS has improved the classification performance for

DT, NB, LDA, and QDA classifiers, while for KNN and linear SVM classifiers the same performance is achieved but with a lower number of features. For the forward search methodology, the highest performance is accomplished with 22 and 24 features using KNN and SVM classifiers, respectively, where the accuracy, sensitivity, specificity, precision, F1-score, and MCC for the KNN classifiers are 99.9%, 99.9%, 100%, 99.9%, 99.9%, and 99.9%. The same performance is attained with the SVM classifier except for the MCC, which is 99.8%.

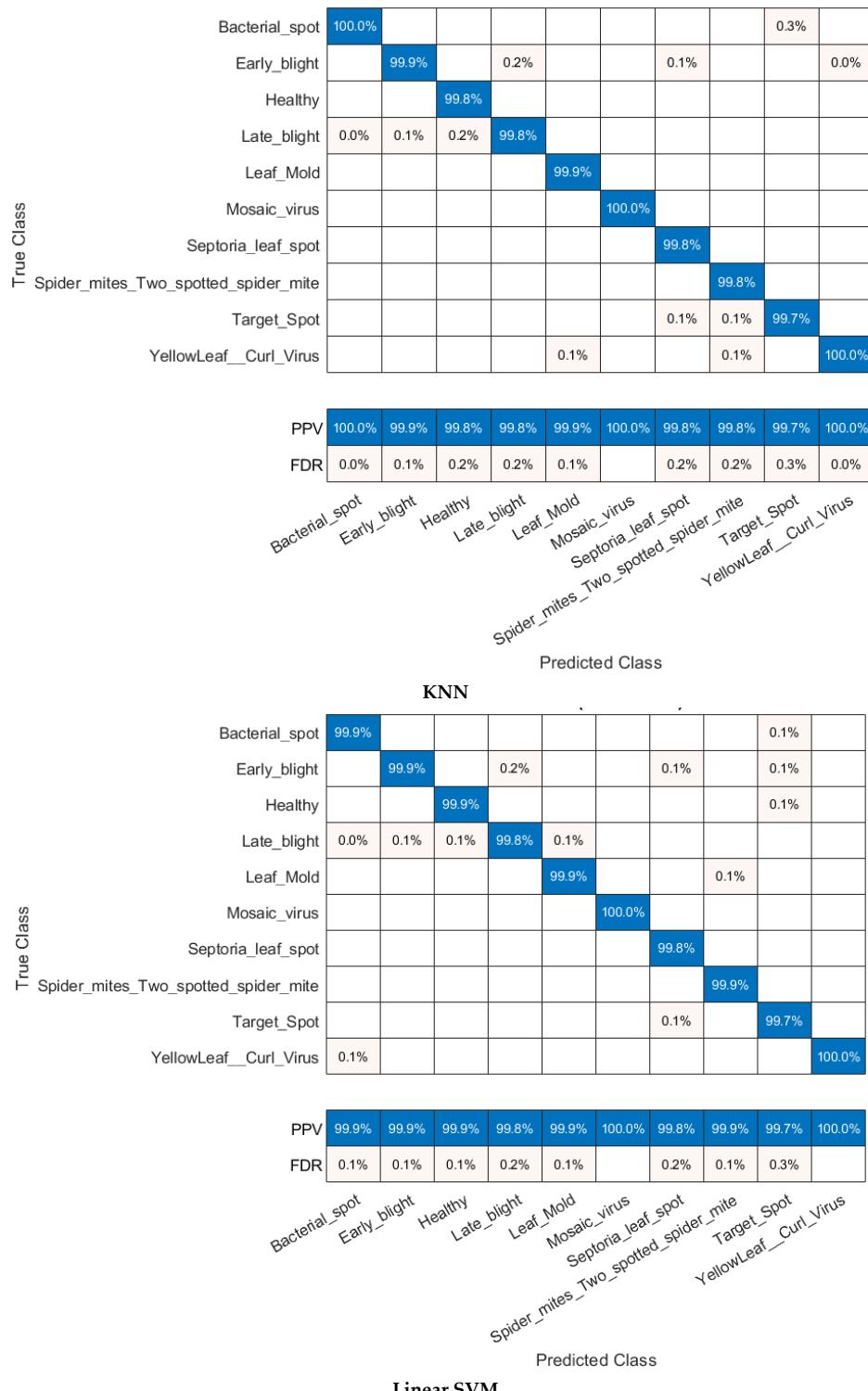
**Table 4.** A comparison between the results (%) obtained using each search strategy for each classifier and the number of features used in each methodology.

	# Features	Accuracy	Sensitivity	Specificity	Precision	F1-Score	MCC
<b>Forward Search Methodology</b>							
<b>NB</b>	21	99.80	99.8	100	99.8	99.8	99.8
<b>LDA</b>	21	99.80	99.8	100	99.8	99.8	99.8
<b>QDA</b>	15	99.80	99.8	100	99.8	99.8	99.8
<b>LSVM</b>	24	99.90	99.9	100	99.9	99.9	99.8
<b>KNN</b>	22	99.90	99.9	100	99.9	99.9	99.9
<b>DT</b>	14	99.34	99.3	99.9	99.3	99.3	99.3
<b>Backward Search Methodology</b>							
<b>NB</b>	19	99.81	99.8	100	99.8	99.8	99.8
<b>LDA</b>	24	99.81	99.8	100	99.8	99.8	99.8
<b>QDA</b>	16	99.92	99.9	100	99.9	99.9	99.8
<b>LSVM</b>	16	99.8	99.8	100	99.8	99.8	99.8
<b>KNN</b>	24	99.91	99.9	100	99.9	99.9	99.9
<b>DT</b>	14	99.34	99.3	99.9	99.3	99.3	99.3
<b>Bidirectional Search Methodology</b>							
<b>NB</b>	21	99.81	99.8	100	99.8	99.8	99.8
<b>LDA</b>	19	99.81	99.8	100	99.8	99.8	99.8
<b>QDA</b>	16	99.92	99.9	100	99.9	99.9	99.9
<b>LSVM</b>	24	99.90	99.9	100	99.9	99.9	99.8
<b>KNN</b>	22	99.92	99.9	100	99.9	99.9	99.9
<b>DT</b>	14	99.36	99.4	99.9	99.4	99.4	99.3

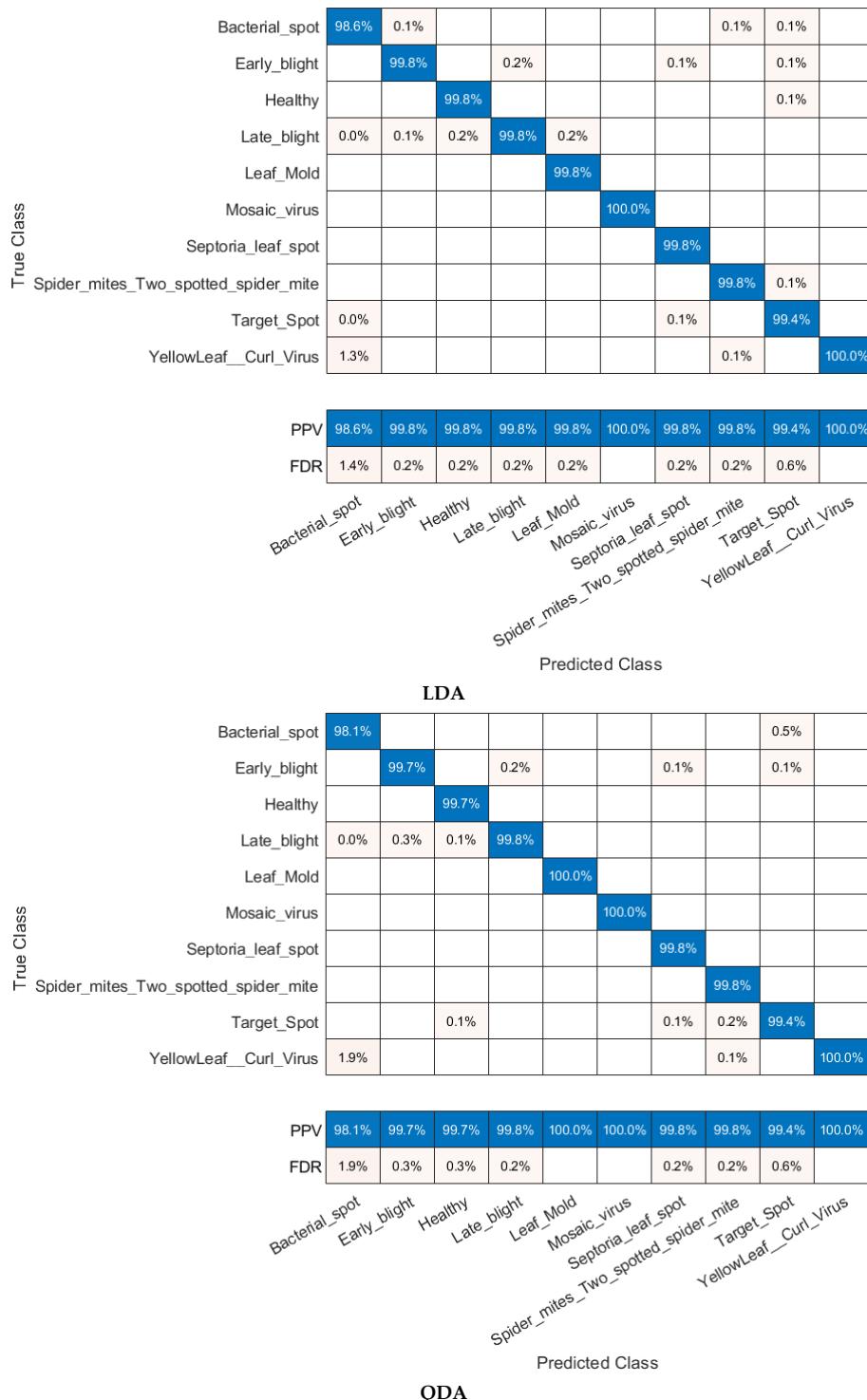
In the case of the backward strategy, the QDA classifier achieves higher accuracy than the forward strategy with 16 features, whereas the KNN with 24 features attains the same performance as the forward strategy. Meanwhile, the NB, LDA, and DT classifiers accomplish similar classification results but with 19, 24, and 14 features. However, for the SVM classifiers, the performance is lower than the forward strategy. In contrast to the bidirectional search methodology, the DT with 14 features reaches a slightly higher accuracy than the forward and backward strategies. Similarly, the KNN with 22 features has slightly better accuracy than both forward and backward strategies with fewer features. The QDA, LDA, and NB attain the same performance as the backward strategy but with 16, 19, and 21 features. For the SVM classifier, the classification accuracy attained is higher than the backward strategy with 24 features. The confusion matrices for the KNN, linear SVM, LDA, and QDA are displayed in Figure 4.

During testing, the ROC curve performances achieved utilizing the KNN classifier after feature selection using the bidirectional search methodology are depicted in Figure 5a–f. The x-axis in the figures represents the false-positive rate, while the y-axis represents the false-negative rate. Each of these ROC curves is plotted assuming that one of the tomato

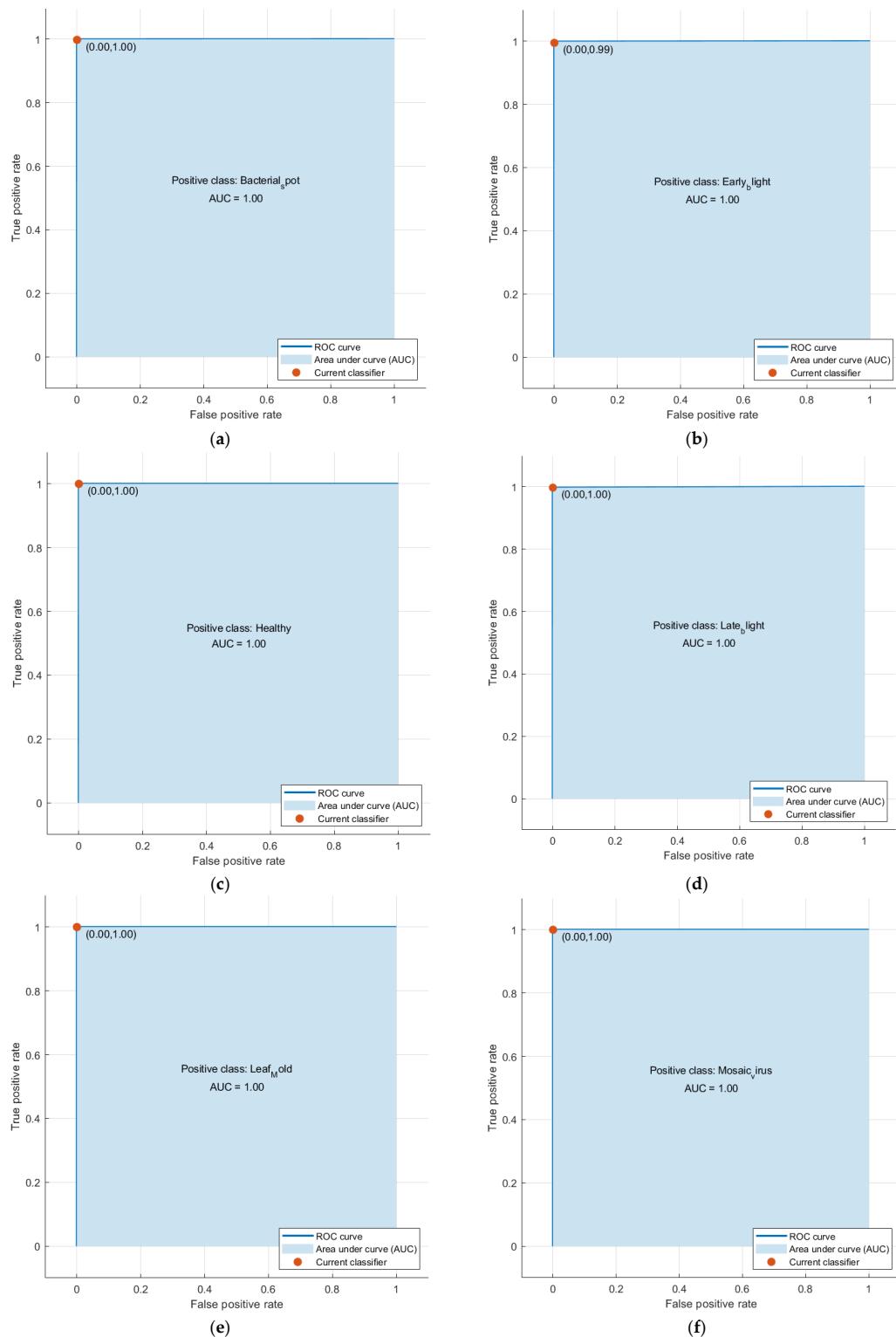
diseases is the positive class label. Figure 5a–c displays that the positive class is bacterial spot disease, early blight disease, healthy, late blight illness, leaf mold illness, and mosaic virus. Our proposed model performs similarly in terms of ROC across the dataset's classes. For the entire dataset's classes, the proposed model obtained 1 area under ROC (AUC), which indicates that the proposed model is capable of differentiating between different tomato diseases accurately.



**Figure 4. Cont.**



**Figure 4.** The confusion matrices of the KNN and linear SVM classifiers after feature selection using the bidirectional search methodology.



**Figure 5.** The ROC curves for the KNN classifier after feature selection using the bidirectional search methodology. (a) Bacterial Spot. (b) Early Blight. (c) Healthy. (d) Late Blight. (e) Leaf Mold. (f) Mosaic Virus.

## 6. Discussion

This study proposed a pipeline for the automatic detection and diagnosis of tomato leaf disease. Opposing the methods used in the literature which are based on individual CNN models that involve a large number of deep layers with an extremely large number of

parameters, the proposed pipeline utilized three compact CNNs of different architectures merging the advantages of each composition. The proposed pipeline is formed of four steps which are tomato leaf image preparation, compact CNN training and feature extraction with TL, feature concatenation and selection, and finally classification. First, the dimension of tomato leaf images is altered followed by an augmentation process. Then, these preprocessed images were utilized to train three CNNs involving ResNet-18, ShuffleNet, and MobileNet with TL. Next, TL was used to obtain high-level features from the final FC layer of each CNN. Thereafter, these features were concatenated, and a hybrid FS approach was employed to select a comprehensive set of deep features of lower dimension. Finally, six ML classifiers were used to classify these images.

The classification procedure of the proposed pipeline was conducted in three contexts. The first context employed each deep feature set acquired from every CNN to train the six ML classifiers separately. In the second context, the ML classifiers were fed with the concatenated deep features of the three CNNs. Lately, in context 3, the features chosen by the hybrid FS were applied as inputs to the six ML classifiers. The results of context 2 showed that utilizing features from several CNNs of distinct structures can enhance the classification performance. Furthermore, the results of context 3 verified that the presented hybrid FS could enhance or at least achieve the same classification performance but with a lower number of features which decreases the complexity of the training procedure of the ML classifiers. The KNN and linear SVM classifiers reached an accuracy of 99.92% and 99.90% with only 22 and 24 features.

### 6.1. Comparative Analysis

The effectiveness of the proposed pipeline was compared to models previously investigated by other researchers. Several studies used the tomato leaf infection PlantVillage dataset along with various deep CNN architectures. Table 5 performs a comparison among the experimental results of a variety of DL constructions on the PlantVillage database. It could be figured from Table 5 that the proposed pipeline has higher accuracy than previous models for tomato leaf disease classification. The reason can be seen in Table 5, as most earlier methods are based on individual DL models. These methods did not apply any FS approaches to choose the most significant features. Conversely, the proposed pipeline combined deep features from multiple CNNs which enhanced performance. It also utilized a hybrid FS approach for the selection of a lowered group of comprehensive features.

### 6.2. Limitations and Forthcoming Directions

Notwithstanding the promising performance of the pipeline presented for the recognition of tomato leaf diseases, the research has some shortcomings. To begin, this study only examines ten different tomato leaf illnesses; other categories of tomato leaves diseases as well as other crop illnesses were not examined. In the future, the author will attempt to enhance the presented model to be applicable to other tomato infections and additional crop leaf diseases. Future work will consider more crops including potatoes, apples, grapes, pepper, oranges, etc. Other features that could be added involve using deep learning segmentation and detection approaches including Yolo, U-Net, regional convolutional neural networks (R-CNN), fast R-CNN, and faster CNN in order to detect and differentiate between crops and facilitate the identification and differentiation among diseases that have similar patterns. In addition, using image enhancement approaches to ease the training procedure of the deep learning models in order to discriminate between different crop diseases. Furthermore, more features could be added when working with other cultures, such as collecting images of different species of each crop planted in various countries. Then, learning the deep learning models with these numerous species to increase their generalizability and increase their capability to classify crops of different species acquired from distinct cultures.

Additionally, the proposed pipeline has another main disadvantage of utilizing images that have been captured in a laboratory environment. Nevertheless, the introduced pipeline

could be enhanced to endorse a real-world integrative plant disease identification system. Regrettably, more experiments are required for advancing the presented pipeline to be capable of identifying increasing groups of crop infections and automatically recognizing the disease's many stages in natural settings. In addition, this study did not consider identifying the severity of tomato leaf diseases. Forthcoming work will consider addressing this issue.

**Table 5.** Comparison of performance with cutting-edge models for tomato leaf disease classification based on the PlantVillage dataset.

Article	# Diseases	Model	Features	Accuracy
[51]	10	ResNet-50	Features of ResNet-50	97.0%
[28]	10	U-Net	ResNet-50	97.11%
[3]	10	Customized CNN	Customized CNN	99.3%
[21]	10	Customized CNN	Customized CNN	98.70%
[52]	4	Fine-tuned MobileNet	Features of MobileNet	90.3%
[53]	10	Spatial attention with CNN	Fully connected layer	95.20%
[54]	7	VGG16	Features of VGG16	96.19%
[2]	6	Multinomial Logistic regression	MobileNetV2 or NASNetMobile	97%
[55]	10	EfficientNet-B0	EfficientNet-B0	98.60%
Proposed	10	KNN	Fully connected layer (MobileNet + ShuffleNet + ResNet-18) + hybrid FS	99.92%

## 7. Conclusions

Recently, the agricultural sector has faced numerous challenges. Rapid and precise detection of leaf diseases could indeed assist in meeting the constantly expanding requirement for food production. In plant disease detection, DL-based approaches have yielded promising results. In this regard, this study proposed a robust and precise DL-based pipeline for the automatic detection and identification of tomato diseases. Rather than utilizing a single CNN model which requires excessive computational capacity and resources utilized in the majority of the previous studies, the proposed pipeline employed three compact CNN structures with fewer deep layers and low parameters. Furthermore, it employed a hybrid FS approach to select a comprehensive feature set of lesser dimensions which is not the case in most of the previous studies which relied on large features. The presented pipeline acquired high-level features from the final FC layer of each CNN and then merged them. Next, it applied the hybrid FS to choose the most significant features. Six ML classifiers were utilized to classify tomato leaf photos. The results of the proposed pipeline indicated that combining features obtained from CNNs of different structures improved performance. Moreover, the presented hybrid FS approach successfully selected a reduced set of significant deep features. These chosen features lowered the complexity of the training models and achieved at least the same accuracy as the full set of combined features or even slightly higher. Furthermore, the results proved the superiority of the experimental performance of the presented pipeline when compared with other earlier research conducted for tomato leaf diseases.

**Funding:** This research received no external funding.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Data are accessed through the following link on 1 October 2022.

**Conflicts of Interest:** The author declares no conflict of interest.

## References

1. Savary, S.; Ficke, A.; Aubertot, J.-N.; Hollier, C. Crop Losses Due to Diseases and Their Implications for Global Food Production Losses and Food Security. *Food Secur.* **2012**, *4*, 519–537. [[CrossRef](#)]
2. Al-gaashani, M.S.; Shang, F.; Muthanna, M.S.; Khayyat, M.; Abd El-Latif, A.A. Tomato Leaf Disease Classification by Exploiting Transfer Learning and Feature Concatenation. *IET Image Process.* **2022**, *16*, 913–925. [[CrossRef](#)]
3. Ahmed, S.; Hasan, M.B.; Ahmed, T.; Sony, M.R.K.; Kabir, M.H. Less Is More: Lighter and Faster Deep Neural Architecture for Tomato Leaf Disease Classification. *IEEE Access* **2022**, *10*, 68868–68884. [[CrossRef](#)]
4. Panno, S.; Davino, S.; Caruso, A.G.; Bertacca, S.; Crnogorac, A.; Mandić, A.; Noris, E.; Matić, S. A Review of the Most Common and Economically Important Diseases That Undermine the Cultivation of Tomato Crop in the Mediterranean Basin. *Agronomy* **2021**, *11*, 2188. [[CrossRef](#)]
5. Azlah, M.A.F.; Chua, L.S.; Rahmad, F.R.; Abdullah, F.I.; Wan Alwi, S.R. Review on Techniques for Plant Leaf Classification and Recognition. *Computers* **2019**, *8*, 77. [[CrossRef](#)]
6. Li, L.; Zhang, S.; Wang, B. Plant Disease Detection and Classification by Deep Learning—A Review. *IEEE Access* **2021**, *9*, 56683–56698. [[CrossRef](#)]
7. Attallah, O. CoMB-Deep: Composite Deep Learning-Based Pipeline for Classifying Childhood Medulloblastoma and Its Classes. *Front. Neuroinform.* **2021**, *15*, 663592. [[CrossRef](#)]
8. Attallah, O. MB-AI-His: Histopathological Diagnosis of Pediatric Medulloblastoma and Its Subtypes via AI. *Diagnostics* **2021**, *11*, 359. [[CrossRef](#)]
9. Attallah, O. A Computer-Aided Diagnostic Framework for Coronavirus Diagnosis Using Texture-Based Radiomics Images. *Digit. Health* **2022**, *8*, 20552076221092544. [[CrossRef](#)]
10. Attallah, O.; Samir, A. A Wavelet-Based Deep Learning Pipeline for Efficient COVID-19 Diagnosis via CT Slices. *Appl. Soft Comput.* **2022**, *128*, 109401. [[CrossRef](#)]
11. Attallah, O. An Intelligent ECG-Based Tool for Diagnosing COVID-19 via Ensemble Deep Learning Techniques. *Biosensors* **2022**, *12*, 299. [[CrossRef](#)] [[PubMed](#)]
12. Attallah, O. GabROP: Gabor Wavelets-Based CAD for Retinopathy of Prematurity Diagnosis via Convolutional Neural Networks. *Diagnostics* **2023**, *13*, 171. [[CrossRef](#)] [[PubMed](#)]
13. Attallah, O. DIAROP: Automated Deep Learning-Based Diagnostic Tool for Retinopathy of Prematurity. *Diagnostics* **2021**, *11*, 2034. [[CrossRef](#)] [[PubMed](#)]
14. Attallah, O. ECG-BiCoNet: An ECG-Based Pipeline for COVID-19 Diagnosis Using Bi-Layers of Deep Features Integration. *Comput. Biol. Med.* **2022**, *2022*, 105210. [[CrossRef](#)] [[PubMed](#)]
15. Attallah, O. A Deep Learning-Based Diagnostic Tool for Identifying Various Diseases via Facial Images. *Digit. Health* **2022**, *8*, 20552076221124430. [[CrossRef](#)]
16. Attallah, O. RADIC: A Tool for Diagnosing COVID-19 from Chest CT and X-Ray Scans Using Deep Learning and Quad-Radiomics. *Chemom. Intell. Lab. Syst.* **2023**, *233*, 104750. [[CrossRef](#)]
17. Attallah, O. An Effective Mental Stress State Detection and Evaluation System Using Minimum Number of Frontal Brain Electrodes. *Diagnostics* **2020**, *10*, 292. [[CrossRef](#)]
18. Attallah, O.; Zaghloul, S. AI-Based Pipeline for Classifying Pediatric Medulloblastoma Using Histopathological and Textural Images. *Life* **2022**, *12*, 232. [[CrossRef](#)]
19. Agarwal, K.; Jalali, S. Classification of Retinopathy of Prematurity: From Then till Now. *Community Eye Health* **2018**, *31*, S4.
20. Attallah, O.; Morsi, I. An Electronic Nose for Identifying Multiple Combustible/Harmful Gases and Their Concentration Levels via Artificial Intelligence. *Measurement* **2022**, *199*, 111458. [[CrossRef](#)]
21. Agarwal, M.; Gupta, S.K.; Biswas, K.K. Development of Efficient CNN Model for Tomato Crop Disease Identification. *Sustain. Comput. Inform. Syst.* **2020**, *28*, 100407. [[CrossRef](#)]
22. Kamarlis, A.; Prenafeta-Boldú, F.X. Deep Learning in Agriculture: A Survey. *Comput. Electron. Agric.* **2018**, *147*, 70–90. [[CrossRef](#)]
23. Hassan, S.M.; Maji, A.K.; Jasiński, M.; Leonowicz, Z.; Jasińska, E. Identification of Plant-Leaf Diseases Using CNN and Transfer-Learning Approach. *Electronics* **2021**, *10*, 1388. [[CrossRef](#)]
24. Karthik, R.; Hariharan, M.; Anand, S.; Mathikshara, P.; Johnson, A.; Menaka, R. Attention Embedded Residual CNN for Disease Detection in Tomato Leaves. *Appl. Soft Comput.* **2020**, *86*, 105933.
25. Kaya, A.; Keceli, A.S.; Catal, C.; Yalic, H.Y.; Temucin, H.; Tekinerdogan, B. Analysis of Transfer Learning for Deep Neural Network Based Plant Classification Models. *Comput. Electron. Agric.* **2019**, *158*, 20–29. [[CrossRef](#)]
26. Kaur, S.; Pandey, S.; Goel, S. Plants Disease Identification and Classification through Leaf Images: A Survey. *Arch. Comput. Methods Eng.* **2019**, *26*, 507–530. [[CrossRef](#)]
27. Hughes, D.; Salathé, M. An Open Access Repository of Images on Plant Health to Enable the Development of Mobile Disease Diagnostics. *arXiv* **2015**, arXiv:1511.08060.
28. Abbas, A.; Jain, S.; Gour, M.; Vankudothu, S. Tomato Plant Disease Detection Using Transfer Learning with C-GAN Synthetic Images. *Comput. Electron. Agric.* **2021**, *187*, 106279. [[CrossRef](#)]
29. Li, M.; Zhou, G.; Chen, A.; Yi, J.; Lu, C.; He, M.; Hu, Y. FWDGAN-Based Data Augmentation for Tomato Leaf Disease Identification. *Comput. Electron. Agric.* **2022**, *194*, 106779. [[CrossRef](#)]

30. Bhujel, A.; Kim, N.-E.; Arulmozhi, E.; Basak, J.K.; Kim, H.-T. A Lightweight Attention-Based Convolutional Neural Networks for Tomato Leaf Disease Classification. *Agriculture* **2022**, *12*, 228. [[CrossRef](#)]
31. Özbilge, E.; Ulukök, M.K.; Toygar, Ö.; Ozbilge, E. Tomato Disease Recognition Using a Compact Convolutional Neural Network. *IEEE Access* **2022**, *10*, 77213–77224. [[CrossRef](#)]
32. Thangaraj, R.; Anandamurugan, S.; Kaliappan, V.K. Automated Tomato Leaf Disease Classification Using Transfer Learning-Based Deep Convolution Neural Network. *J. Plant Dis. Prot.* **2021**, *128*, 73–86. [[CrossRef](#)]
33. Kumar, A.; Vani, M. Image Based Tomato Leaf Disease Detection. In Proceedings of the 2019 10th International Conference on Computing, Communication and Networking Technologies (ICCCNT), Kanpur, India, 6–8 July 2019; IEEE: New York, NY, USA, 2019; pp. 1–6.
34. Maeda-Gutiérrez, V.; Galvan-Tejada, C.E.; Zanella-Calzada, L.A.; Celaya-Padilla, J.M.; Galván-Tejada, J.I.; Gamboa-Rosales, H.; Luna-Garcia, H.; Magallanes-Quintanar, R.; Guerrero Mendez, C.A.; Olvera-Olvera, C.A. Comparison of Convolutional Neural Network Architectures for Classification of Tomato Plant Diseases. *Appl. Sci.* **2020**, *10*, 1245. [[CrossRef](#)]
35. Tan, L.; Lu, J.; Jiang, H. Tomato Leaf Diseases Classification Based on Leaf Images: A Comparison between Classical Machine Learning and Deep Learning Methods. *AgriEngineering* **2021**, *3*, 542–558. [[CrossRef](#)]
36. Islam, M.S.; Sultana, S.; Farid, F.A.; Islam, M.N.; Rashid, M.; Bari, B.S.; Hashim, N.; Husen, M.N. Multimodal Hybrid Deep Learning Approach to Detect Tomato Leaf Disease Using Attention Based Dilated Convolution Feature Extractor with Logistic Regression Classification. *Sensors* **2022**, *22*, 6079. [[CrossRef](#)]
37. Chen, H.-C.; Widodo, A.M.; Wisnijati, A.; Rahaman, M.; Lin, J.C.-W.; Chen, L.; Weng, C.-E. AlexNet Convolutional Neural Network for Disease Detection and Classification of Tomato Leaf. *Electronics* **2022**, *11*, 951. [[CrossRef](#)]
38. Amin, J.; Sharif, A.; Gul, N.; Anjum, M.A.; Nisar, M.W.; Azam, F.; Bukhari, S.A.C. Integrated Design of Deep Features Fusion for Localization and Classification of Skin Cancer. *Pattern Recognit. Lett.* **2020**, *131*, 63–70. [[CrossRef](#)]
39. Amrani, M.; Hammad, M.; Jiang, F.; Wang, K.; Amrani, A. Very Deep Feature Extraction and Fusion for Arrhythmias Detection. *Neural Comput. Appl.* **2018**, *30*, 2047–2057. [[CrossRef](#)]
40. Zhang, Q.; Li, H.; Sun, Z.; Tan, T. Deep Feature Fusion for Iris and Periocular Biometrics on Mobile Devices. *IEEE Trans. Inf. Secur.* **2018**, *13*, 2897–2912. [[CrossRef](#)]
41. Nandhini, S.; Ashokkumar, K. Improved Crossover Based Monarch Butterfly Optimization for Tomato Leaf Disease Classification Using Convolutional Neural Network. *Multimed. Tools Appl.* **2021**, *80*, 18583–18610. [[CrossRef](#)]
42. Gadekallu, T.R.; Rajput, D.S.; Reddy, M.; Lakshmanna, K.; Bhattacharya, S.; Singh, S.; Jolfaei, A.; Alazab, M. A Novel PCA–Whale Optimization-Based Deep Neural Network Model for Classification of Tomato Plant Diseases Using GPU. *J. Real-Time Image Process.* **2021**, *18*, 1383–1396. [[CrossRef](#)]
43. Tian, K.; Zeng, J.; Song, T.; Li, Z.; Evans, A.; Li, J. Tomato Leaf Diseases Recognition Based on Deep Convolutional Neural Networks. *J. Agric. Eng.* **2022**, in press. [[CrossRef](#)]
44. Alhenawi, E.; Al-Sayyed, R.; Hudaib, A.; Mirjalili, S. Feature Selection Methods on Gene Expression Microarray Data for Cancer Classification: A Systematic Review. *Comput. Biol. Med.* **2022**, *140*, 105051. [[CrossRef](#)]
45. Fei, H.; Fan, Z.; Wang, C.; Zhang, N.; Wang, T.; Chen, R.; Bai, T. Cotton Classification Method at the County Scale Based on Multi-Features and Random Forest Feature Selection Algorithm and Classifier. *Remote Sens.* **2022**, *14*, 829. [[CrossRef](#)]
46. Chandrashekhar, G.; Sahin, F. A Survey on Feature Selection Methods. *Comput. Electr. Eng.* **2014**, *40*, 16–28. [[CrossRef](#)]
47. Attallah, O.; Karthikesalingam, A.; Holt, P.J.; Thompson, M.M.; Sayers, R.; Bown, M.J.; Choke, E.C.; Ma, X. Using Multiple Classifiers for Predicting the Risk of Endovascular Aortic Aneurysm Repair Re-Intervention through Hybrid Feature Selection. *Proc. Inst. Mech. Eng. Part H J. Eng. Med.* **2017**, *231*, 1048–1063. [[CrossRef](#)]
48. Attallah, O.; Karthikesalingam, A.; Holt, P.J.; Thompson, M.M.; Sayers, R.; Bown, M.J.; Choke, E.C.; Ma, X. Feature Selection through Validation and Un-Censoring of Endovascular Repair Survival Data for Predicting the Risk of Re-Intervention. *BMC Med. Inform. Decis. Mak.* **2017**, *17*, 115–133. [[CrossRef](#)]
49. Pasha, S.J.; Mohamed, E.S. Ensemble Gain Ratio Feature Selection (EGFS) Model with Machine Learning and Data Mining Algorithms for Disease Risk Prediction. In Proceedings of the 2020 International Conference on Inventive Computation Technologies (ICICT), Coimbatore, India, 26–28 February 2020; pp. 590–596.
50. Hall, M.; Frank, E.; Holmes, G.; Pfahringer, B.; Reutemann, P.; Witten, I.H. The WEKA Data Mining Software: An Update. *ACM SIGKDD Explor. Newsl.* **2009**, *11*, 10–18. [[CrossRef](#)]
51. Wspanialy, P.; Moussa, M. A Detection and Severity Estimation System for Generic Diseases of Tomato Greenhouse Plants. *Comput. Electron. Agric.* **2020**, *178*, 105701. [[CrossRef](#)]
52. Zaki, S.Z.M.; Zulkifley, M.A.; Stofa, M.M.; Kamari, N.A.M.; Mohamed, N.A. Classification of Tomato Leaf Diseases Using MobileNet V2. *IAES Int. J. Artif. Intell.* **2020**, *9*, 290. [[CrossRef](#)]
53. Mohanty, S.P.; Hughes, D.P.; Salathé, M. Using Deep Learning for Image-Based Plant Disease Detection. *Front. Plant Sci.* **2016**, *7*, 1419. [[CrossRef](#)] [[PubMed](#)]

54. Rangarajan, A.K.; Purushothaman, R.; Ramesh, A. Tomato Crop Disease Classification Using Pre-Trained Deep Learning Algorithm. *Procedia Comput. Sci.* **2018**, *133*, 1040–1047. [[CrossRef](#)]
55. Bir, P.; Kumar, R.; Singh, G. Transfer Learning Based Tomato Leaf Disease Detection for Mobile Applications. In Proceedings of the 2020 IEEE International Conference on Computing, Power and Communication Technologies (GUCON), Greater Noida, 2–4 October 2020; IEEE: New York, NY, USA, 2020; pp. 34–39.

**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.