IAES International Journal of Artificial Intelligence (IJ-AI)
Vol. 9, No. 4, December 2020, pp. 670~683
ISSN: 2252-8938, DOI: 10.11591/ijai.v9.i4.pp670-683

Machine learning for plant disease detection: an investigative comparison between support vector machine and deep learning

Aliyu M. Abdu, Musa M. Mokji, Usman U. Sheikh
School of Electrical Engineering, Universiti Teknologi Malaysia, 81310 Skudai, Johor, Malaysia

Article Info

Article history:

Received Feb 8, 2020
Revised Jul 14, 2020
Accepted Oct 15, 2020

Keywords:

Deep learning
Image processing
Machine learning
Plant disease detection
SVM

ABSTRACT

Image-based plant disease detection is among the essential activities in precision agriculture for observing incidence and measuring the severity of variability in crops. 70% to 80% of the variabilities are attributed to diseases caused by pathogens, and 60% to 70% appear on the leaves in comparison to the stem and fruits. This work provides a comparative analysis through the model implementation of the two renowned machine learning models, the support vector machine (SVM) and deep learning (DL), for plant disease detection using leaf image data. Until recently, most of these image processing techniques had been, and some still are, exploiting what some considered as "shallow" machine learning architectures. The DL network is fast becoming the benchmark for research in the field of image recognition and pattern analysis. Regardless, there is a lack of studies concerning its application in plant leaves disease detection. Thus, both models have been implemented in this research on a large plant leaf disease image dataset using standard settings and in consideration of the three crucial factors of architecture, computational power, and amount of training data to compare the duos. Results obtained indicated scenarios by which each model best performs in this context, and within a particular domain of factors suggests improvements and which model would be more preferred. It is also envisaged that this research would provide meaningful insight into the critical current and future role of machine learning in food security.

Corresponding Author:

Aliyu M. Abdu
School of Electrical Engineering
Universiti Teknologi Malaysia
Johor, Malaysia
Email: aliyu104@yahoo.com

1. INTRODUCTION

Plant disease detection is a crucial part of precision agriculture that primarily deals with the observation of the earliest stages of diseases in plants [1]. As disease outbreaks are also increasingly becoming rampant across the globe, the outcome of early disease detection can be used for disease diagnosis, control, and damage assessment, especially since some are extremely difficult to control and can lead to famine [2-3]. Furthermore, the information can also help with the application of disease-specific remedy or chemical applications such as pesticide and fungicide for improved productivity and avert losses that can range into billions [4]. From literature, plant disease detection is about measuring the disease incidence, its severity, and consequence [4]. Disease incidence is the proportion of plants in a farm or leaves on a diseased plant. Severity, sometimes interchanged with intensity, can be expressed as the rate at which the disease area of the plant manifest (i.e., relative or absolute area damaged by disease). At the same time, the consequence is the outcome in the form of a percentage of yield lost or quality of yield drop in the harvest.

Plant disease detection methods are classified into destructive (serology and molecular methods) and non-destructive (biomarker-based and plant properties image processing-based techniques included) [4]. The use of image-based techniques in precision agriculture has grown tremendously through machine learning-based methods. This can be attributed to the availability of higher-quality measurements, coupled with modern algorithms and an increased possibility to fuse multiple sources of images. These images can either be from satellite imagery, sensors, or even cameras positioned in fields. Earlier imaging techniques were not cost-effective, which include hyperspectral imaging, fluorescence imaging, spectroscopy, infrared, and even x-ray imaging [5-7]. Nowadays, image processing techniques, together with machine learning classifiers, can efficiently identify such diseases in color images at advanced levels with excellent precision [7]. Also, classification based on plant properties can be mimicked through color, shape, and texture features to enhance classification accuracy. The enormous potential for the success of these algorithms has motivated further development in herbicide applications [6]. Fuzzy algorithms based on green color analysis of plants have allowed for the integration of this knowledge into farm management plans and provided disease coverage estimation [6]. Hence, from the applications in precision agriculture listed above, we can easily imagine the future of the role of machine learning in agricultural processes, particularly in this aspect of diseases associated with plant leaves.

The machine learning process can either be supervised or unsupervised. Supervised learning is where the machine is taught and trained using a well-labeled image dataset of diseased pair. That is the data that is already tagged with the correct disease classifications or absence thereof. The higher or larger the dataset, the more accurate the machine learns [5]. In practice, two main classification approaches exist, which are a deep learning approach and a conventional classifier approach [8-9]. In the DL approach, a DL classifier incorporates several layers of information processing stages that are arranged in a hierarchy of neural network layers. These layers are then exploited for feature learning, analysis, and pattern classification [10]. The DL classifiers can automatically learn 1000s of global feature representations from the whole leaf, a region or neighborhood, or a segmented region of interest [10-12]. The examples for these classifiers include convolutional neural networks (CNNs), recurrent neural networks (RNNs), and deep neural networks (DNNs), among which CNN is most popular and has shown excellent performance for object and image classification [9, 13-14]. On the other hand, a conventional classifier analyzes the input labeled data and observes the correlation between feature attributes that are extracted from a region of interest within the image during training. These features are then used as the learned parameters to predict the subsequent unlabeled data during disease classification [6, 15]. There are also many different conventional techniques for building machine learning models such as the K-nearest neighbor (K-NN), naive bayes, support vector machines (SVMs), and linear discriminant analysis (LDA). However, most research in plant disease detection has emphasized the use of the SVM learning algorithm above the other conventional classifiers [7, 16]. SVM has been widely used in solving many well-constrained plant disease detection problems in smaller data. In more often cases than not, it is highly likely to have a trade-off between limited modeling with SVM and representational power with DL. Furthermore, the D-CNN network, even though it has been very much around for decades, only recently has just begun its debut in image processing applications for plant disease detection [9, 17-19].

Most of the current machine learning methods are yet to be robust enough to bridge the gap between the real-world methods of plant disease detection, often due to the choice of process or its specificity to one symptom [5, 17]. One of the most difficult challenges is that plant species have some diseases with a significant degree of similarity and can appear simultaneously on a single plant. This affects not only the image processing and machine learning methods but equally human experts as well. Thus, it is paramount for the choice of method to be relevant for the target disease detection.

The literature on related works

Several research studies on machine learning for image-based plant disease classification have been reported [1, 7, 20-21]. In summary, a large portion focused on conventional classification techniques such as SVM while others on DL. Kaur *et al*. made a more comprehensive summary of machine learning methods applied in different plant cultures. One of the bases for their observations was to identify the most popular classifier [3]. Work from Camargo and Smith is one of the earliest proposed works on pattern recognition using SVM. Diseased regions such as spots, lesions or stains, and strikes were identified and segmented, and features were later extracted and fed as inputs to the classifier [22]. With a dataset of 117 images, the trained classifier recorded 93% accuracy. They proved the hypothesis of using texture features as valid discriminators for plant disease identification, furthermore, set a precedent for using the extracted features as inputs to the machine learning algorithm to identify plant disease visual symptoms. Camargo and Smith, in separate research, presented the image registration procedure on how preprocessing (filtering and intensity distribution), identification, and segmentation of the disease symptoms regions (ROIs) were made [23]. The

features were obtained from hue saturation and value (HSV) color channels of the ROI using co-occurrence matrix methodology. Bernardes *et al*. also used the SVM for the automatic cotton foliar disease classification, although the wavelet transform was employed as the feature extractor [24]. The image dataset used contained a total number of 420 images of varying scales and intensities. However, the work did not report the methods used for the preprocessing, nor did it specify the use of segmentation algorithms. It is in this regard that despite having a total of 216 feature vectors, the reported classification rate ranged from 80% to 96.2%. Barbedo *et al*. proposed a pair-wise based classification system where the main features were determined using color transformations and relevance histograms [25]. The image dataset was composed of 82 (74 diseases, four pests, and four abiotic disorders) disorders spread across 12 plant species, with only 15% of the total images reported taken under controlled conditions while the rest under real conditions from designated experimental fields. The guided active contour (GAC) method was used for leaf segmentation, which employed two masks generated based on discoloration. The proposed method reported a low average accuracy of 58% across all the species, mainly to the absence of distinct similarities between some diseases, sparse image datasets, and lack of preprocessing on input images. Singh and Misra, in a similar approach, made use of a genetic algorithm (GA) for the segmentation of diseased regions after first performing preprocessing [26]. Both texture and color features were used for the segmentation and subsequent identification of disease presence or absence thereof using the SVM classifier. A dataset of 106 captured leaf images was split into training and testing. The average classification accuracy was reported at 97% when tested on four plant species and five diseases. The research overruled use of shape features such as extent and circularity due to shape variation of as diseases evolve into severe stages. The proposed method tailored the segmentation and feature extraction process to be less compromising. However, the lack of using a proper image dataset reduced the relevance of the work.

Although works are still reported on the application of the SVM, ensuring simplicity and robustness of the segmentation process for diseased ROI seem to be the priority. Dhingra *et al*. presented a neutrosophic approach that utilizes the CIELab color space to detect color homogeneity changes produced by the disease symptoms [27]. Also, Barbedo presented a less complicated image processing approach by exerting changes to the RGB color channel to form 4 binary masks and then applying Boolean operations on the masks to obtain cut-off thresholds [24-25]. Wu *et al*. proposed an effective segmentation method for RGB diseased leaf images based on color transformation, using linear discriminant analysis (LDA) combined with different color separation models (Lab, YCbCr, I1I2I3, RGB, and HSV) [28]. The image data (total of 100 images) comprises of pictures taken from a farm field captured using a high-resolution digital camera. Based on results obtained (average accuracy of 91%), they concluded that the RGB, YCbCr, I1I2I3, and HSV in that order were suitable models to employ for proper segmentation of diseased pixels from the images. Xu *et al*., in their proposed work, incorporated a synergetic method by combining basic image processing algorithms such as edge detection and morphology for segmentation and extracting features for wheat rust disease detection [29]. Conclusively, the ROI segmentation process is exceedingly important when it comes to the SVM use case.

The process of segmentation, however, is not a necessary step when using a DL network. Mohanty *et al*. and Sladojevic *et al*. both used DL network-based approach for the disease classification using transfer learning and gathered images that formed large datasets [30-31]. The former presented perhaps the first application of DL to plant disease detection and classification on a relatively comprehensive PlantVillage image dataset [30]. It constituted over 54,000 images of 14 different crop species (tomato, potato, apple, pepper, etc.) and 26 diseases (healthy and unhealthy). In both works, a deep convolution neural network (DCNN) model was developed and trained to perform the disease classification. Using two different sets of training which are transfer learning (using AlexNet and GoogLeNet [32]) and training from scratch, they achieved over 95% accuracy on both accounts using the dataset. The process of segmentation is overrated in these approaches. Another reported work by Xu *et al*. using the same PlantVillage dataset showed a test accuracy of 90.4% using a VGG-16 model trained with transfer learning [29]. Fuentes *et al*. also applied the DL architecture for real-time implementation on tomato crops [33]. All the listed works on DL were trained using powerful GPUs, and, except for [30], all employed only transfer learning.

In reference to the reviewed literature, conventional machine learning classifiers are still in consistent use [21]. Out of all the research articles, they summarized that 41% had employed the use of SVM while the next popular was at 17% and 14% for NN and K-nearest neighbor (K-NN), respectively. The rest of the classifiers combined deep learning included were at 28%. Thus, this wide margin proves the popularity of SVM among all other conventional classifiers. Deep convolution neural network gave rise to DL, and its current popularity in this context is fast being considered as a benchmark. Hence there is some confusion on the type of model most preferred for application in plant disease detection. This often plagues researchers to choose among the two, particularly with regards to the amount of computational power, training data and required memory for experimental implementations. Despite the progress being made in machine learning

algorithms, particularly the DL, it is yet to be as popular in plant disease detection research. However, why is that? Are there complications or limitations associated with the DL network? Or, are there certain conditions where the use of it is not preferable? If so, what could they be? If DL is superior, why do some researchers still prefer to use conventional classifiers such as SVM for plant disease detection? Can the two be combined for better classification accuracy? It may be challenging to examine and answer all these questions analytically. Nevertheless, we can experimentally compare the conventional method SVM and the DL network towards determining the extent or conditions both systems perform best in a plant disease image dataset. A comparison of the results could also highlight which will be best suited in specific scenarios. There has been a separate quantitative analysis for both networks, particularly SVM. However, to the best of our knowledge, no studies systematically investigate their comparisons in this domain. Furthermore, there is no consensus among researchers regarding which machine learning method can be applied in different conditions to detect plant disease [25]. Thus, this paper analyzed the two exceptional methods, SVM and DL, with the objectives to confirm how competitive a shallow approach is (SVM) comparing to the more complex one (DL). This goes a long way in providing suggestions on the choice of the method depending on specific scenarios. The analysis focuses on disease-pair (healthy and unhealthy) detection but with more emphasis on classification.

2. RESEARCH METHOD

2.1. Experimental set-up and dataset

All experiments were conducted using MATLAB software incorporated with DL and neural network Toolbox (MATLAB®, 2017). The experiments were carried out on an HP® Laptop; Intel Core i7-7500U processor, a clock rate of 2.70 GHz and 4 Mb Cache; incorporated with NVIDIA® GEFORCE 940MX GPU running on Windows 10.

Data acquisition is the first important step in every ML-based plant disease detection method that involves acquiring and registering or preprocessing of the leaf images to form an image dataset. However, most recent reported works have used available labeled datasets that have undergone preprocessing at various levels. These include plantvillage (PV) and digipathos [25, 30]. Their advantages over self-acquired images include thousands of labeled image data, multiple varieties of crop species, and different aspects of plant diseases at various levels of severity. This warrants the adoption of available datasets by several studies as a benchmark image database in this context [34-36]. The PV dataset collects images of diseased and healthy plant leaves spread across 38 assigned labels, each with disease pair (diseased or healthy). This dataset of images showing early and late blight diseases was considered due to their significant degree of symptom similarity.

Furthermore, both diseases show the same symptoms across the vegetable species (potato, tomato, pepper, and eggplant) [37-38]. Early blight and late blight are some of the most severe (destructive) diseases that reduce the overall yield of the potato crop, affecting both home gardeners and large productions [37]. The late blight symptoms are quite like those of early blight, but far more severe as the entire garden or farm filed can be lost within a fortnight [38]. Relevance to this, the utilized crop was potato images labeled as either "Early blight," "Late blight," or "Healthy." A total of 2,152 images were used, with some of them being of the same leaf taken at different (augmented) orientations to allow for real-world scenarios. The dataset used was in color (RGB) format, representing various degrees of disease severity.

The images were of the same scale size of 256×256 pixels used for the model optimization and predictions of both the machine learning algorithms. Furthermore, all the experiments were performed on the version of the potato dataset with a segmented background to allow for a proper basis for comparative analysis with SVM and DL. Figure 1 shows one example of each image class. The whole dataset was split randomly for each class (late blight - LB, early blight - EB, and healthy - HL) into the train-optimize-test (T-O-T) scenario to allow for proper assessment of classification accuracy on unseen data. Also considered were the augmented samples, which were separately split as well. Those splits were 50-20-30 (50% train, 20% optimize, and 30% test), and 60-20-20 (60% train, 20% optimize, and 20% test) for both SVM and DL as shown in Table 1. This is to allow for experiments on the effect of the amount of training data on their accuracy. At each period of training interval, i.e., at the end of every 12 iterations (epoch), mean precision, mean recall, and mean F1 score, which is the measure of the test accuracy, are computed.

Table 1. T – O – T dataset showing the number of samples for each split

EXPERIMENT	1			2		
CLASS	T-O-T (SVM & DL) 50 – 20 – 30			T-O-T (SVM & DL) 60 – 20 – 20		
LB (sample images)	500	200	300	600	200	200
EB (sample images)	500	200	300	600	200	200
HL (sample images)	76	30	46	91	30	31
TOTAL			2,152	TOTAL		2,152

Figure 1. Example of potato leaf image samples from the segmented version of the PlantVillage dataset. From right: healthy, early blight, and late blight

2.2. Segmentation and feature extraction

The PlantVillage dataset images used were already registered. Thus, this had conveniently taken care of the first stage and narrowed down the second. For the DL network training, the images were fed directly as input. Representative disease features needed to be extracted from the images and fed as input to the multiclass SVM architecture model with a default linear kernel. Additionally, the RBF kernel classifier was also implemented. The most commonly associated feature categories are color, texture, and shape, most distinct for each particular plant disease symptom [22]. Though minimal distinctions are observed between early and late blight, this, in turn, makes determining the best method for diseased region (ROI) segmentation to be quite tricky. Furthermore, the symptoms do not have well-defined edges. Instead, they gradually fade into healthy tissue.

In an attempt to deal with these complex situations for ROI segmentation, various approaches have been developed through computer-aided segmentation categorized as 1) manual when the user has to tune one or several parameters in the software to achieve the desired segmentation, 2) semi-automatic where the user manually initializes the segmentation which then proceeds automatically, or 3) automatic where no intervention by the user other than submitting the image to be analyzed. The automatic segmentation method used in this study had earlier been presented [39]. The three channels of RGB color space were utilized to generate four binary masks that, in turn, are combined into a single segmentation mask. This method works well in segmenting the ROI for better feature extraction with precision. Figure 2 shows a sample of the segmentation result.

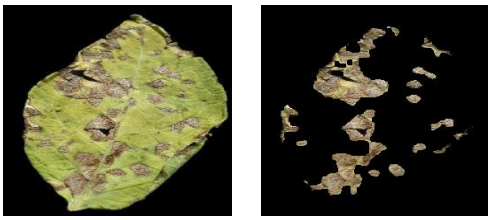


Figure 2. Sample input image (left) and its segmented diseased region (right)

Extracting the features can be done statistically using different image processing algorithms depending on what features best describe the disease symptoms. While most methods would select color and texture, some may choose texture and shape, and others may prefer all three. It is all dependent on what kind of features the diseases in focus exhibit. Furthermore, each feature category has its own sets of values, usually working well for those specific diseases but likely lacking sufficient generality to be extended to other symptom types. It is in this effect that the combination of nine fundamental feature values for color and texture commonly used for the cases of blight was adopted for this research. The gray-level co-occurrence matrix (GLCM) methodology [40] was employed for the extraction of textural features in the ROI. At the same time, the color moment was used for the color feature extraction. Two parameters are usually employed in GLCM computation, the relative distance between the pixel pair d measured in the number of pixels and their relative orientations θ. For an ROI image R, let m denote the pixels (x, y) gray levels and n the gray level of pixels oriented at θ_0 and θ_1 with L level of gray tones. Where $0 \leq x \leq M-1, 0 \leq y \leq N-1$, and $0 \leq m, n \leq L-1$. From these representations, the GLCM $C_{m,n}$ for distance d and direction θ can be given as (1) [40]:

$$C_{m,n,\emptyset} = \sum_x \sum_y P\{R(x,y) = m \,\&\, R(x + d\emptyset_0, y - d\emptyset_1) = n\} \quad (1)$$

Where $P\{.\} = 0$ if the argument is not true and $P\{.\} = 1$ otherwise. The extracted feature values include contrast, correlation, energy, entropy, homogeneity, variance as texture features, and mean, skewness, and kurtosis as color features. Refer to (2-7) for the mathematical formulae for the texture feature [41].

$$contrast = \sum_{i,j=0}^{N-1} (i,j)^2 C(i,j) \quad (2)$$

$$Correlation = \sum_i^{N-1} \sum_j^{N-1} (ij) C(i,j) - \mu_m \mu_n / \sigma_m \sigma_n \quad (3)$$

$$Energy = \sum_{i,j=0}^{N-1} C(i,j)^2 \quad (4)$$

$$Homogeneity = \sum_{i,j=0}^{N-1} C(i,j)/(1 + (i,j)^2) \quad (5)$$

$$Entropy = -\sum_{i,j=0}^{N-1} C(i,j) \log C(i.j) \quad (6)$$

$$Variance = \sum_i^{N-1} \sum_j^{N-1} (i - \mu)^2 C(i,j) \quad (7)$$

where N is the number of gray levels, $C(i,j)$ is the (i,j)th entry in $C_{m,n}$. μ is the mean of $C(i,j)$ where $\mu_m, \mu_n, \sigma_m,$ and σ_n are the average standard deviation of C_m and C_n respectively.

2.3. Classification

The performance of both SVM and DL architectures on the PlantVillage potato dataset was analyzed by training the SVM model on the extracted features on one side and then the DL based on transfer learning on the other side.

Support vector machine (SVM)

SVM is a linear classifier, a set of related supervised learning methods. The input data is linearly mapped to non-linearly separated data in some high dimensional space providing excellent classification performance [42]. Specifically, it analyzes the input labeled data and observes the attributes (training data) to classify subsequent unlabeled data (training data). It does so in a way to maximize the marginal distance (so-called functional margin) between the two or more intra classes, and those points closest to the marginal line are known as the support vectors [42]. Figure 3 shows the underlying architecture of the SVM [43]. The division of classes is carried out with different kernels, such as linear and radial basis functions (RBF). From Figure 1, the two classes (class 1 and 2) represented in a 2D input space (x and y) are separated with a linear kernel as the optimal hyperplane in-between the three support vectors. Like neural networks, the computational complexity of SVMs does not depend on the dimensionality of the input space, but in SVM, it is often difficult to understand the learned function [42]. Though SVM was initially designed to work with only two classes, multiclass SVM classification has become widely applicable. Several two-class SVMs can be implemented to provide a valid prediction, either by using one-versus-all or one-versus-one. The winning class is then determined by the highest output function or the maximum votes, respectively.

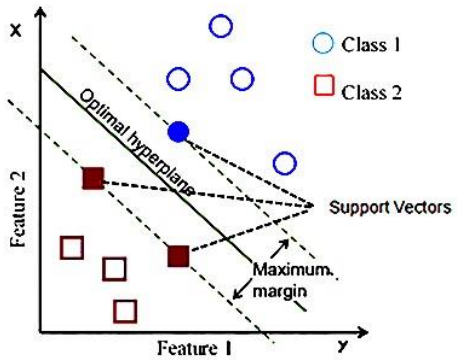


Figure 3. The basic architecture of the SVM

Deep learning (DL) or deep convolution neural network (DCNN)

D-CNN are being applied in diverse domains of plant disease detection as part of automated end-to-end learning [12]. The words "deep learning" (DL) refer to a class of machine learning techniques where several layers of information processing stages are arranged in a hierarchy of nodes to be exploited for unsupervised feature learning, analysis, and pattern classification. The nodes as shown in Figure 4 (a), also known as neurons in the DNN, are mathematical functions that take numerical values for inputs and as incoming edges and numerical output values as an outgoing edge [12]. Each neuron in the hidden layer has weight connections $(w_1, w_2, w_3, \dots, w_n)$ and an activation function $f(\dots)$ as shown in Figure 4 (b), adding layers means more interconnections and weights between and within the layers [44]. DL computes the features of the observational data hierarchically so that the higher-level features are preserved in the deeper layers. They are also defined in terms of those from the surface or low-level layers. Such a hierarchy of features is referred to as the deep architecture [45]. An observational data of an image, for example, can be represented in many ways (e.g., an array of pixels), but some features make it easier to learn specific tasks of interest (e.g., is it an image of a tomato leaf?) from samples. Research in this area attempts to define what makes better representations and the best way to learn them.

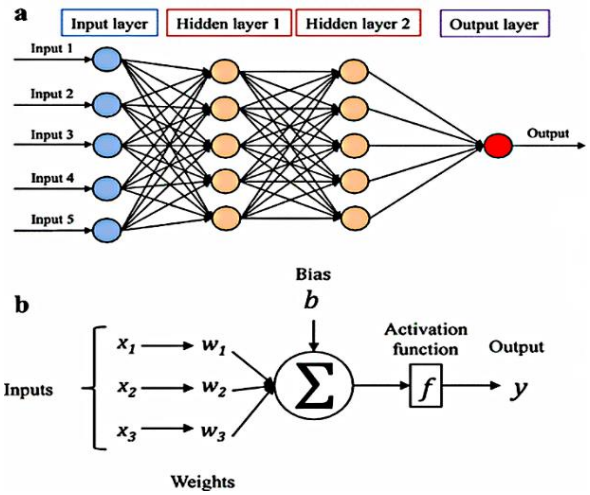


Figure 4. The basic architecture of a deep learning convolution network (a) layer configuration and (b) activations

There are two methods of training a DL network for feature learning: training from scratch and transfer learning. Training a DL from "scratch" is often a complex and time-consuming process due to the deep architecture design [9, 46]. Transfer learning, on the other hand, involves the application of an already established architecture that has been successful in other computer vision domain problems and can adapt to the problem under consideration [30]. Most popular transfer learning architectures include AlexNet [47], VGG Net [48], ResNet [49], and GoogleNet [32], among others. In most cases, the process of transfer learning with the AlexNet has become more favorable for optimizing feature representation and reducing the limitation of architecture complexity [6]. Hence, this network is used in this research.

Transfer learning with AlexNet

In the AlexNet architecture, the variants or local receptive fields usually are arranged convolution layers succeeded by one or more fully connected layers [32]. Those same convolution layers can also have normalization and pooling layers right after them, and traditionally all the layers are initiated or activated using the Rectifying linear unit or function (ReLU). The ReLU applies a transformation to the output of each neuron and then maps the output to the highest possible value or zero if negative. It adaptively learns the parameters of rectifiers, thereby improving accuracy at a negligible extra computational cost. ReLu is given as (8).

$$f(z_i) = \max(0, z_i) \tag{8}$$

Where z_i represents the input of the nonlinear activation function f on the i^{th} channel.

The pooling layer further transforms the output of the activation step by reducing the dimensionality of the features map considering the output of the small region of neurons into a single output. The AlexNet comprises of 5 convolution layers and three fully connected layers with a softMax layer as the final layer. Each convolution layer C_{Li} has maps of the same size for the two directions of x & y of the image given as $M_{ix}\ and\ M_{iy}$ with kernel sizes $k_{ix}\ and\ k_{iy}$ respectively. Then, given the number of pixels to skip during the traverse at both directions, denoted as $S_{ix}\ and\ S_{iy}$. The final output map size could be given as (9-10) [50].

$$M_{ix}^{L} = \frac{M_{ix}^{L-1} - k_{ix}^{L}}{S_{ix}^{L}+1} + 1 \quad (9)$$

$$M_{iy}^{L} = \frac{M_{iy}^{L-1} - k_{iy}^{L}}{S_{iy}^{L}+1} + 1 \quad (10)$$

Where L denotes the layers.

The last fully-connected layer right before the SoftMax (fc7) layer tagged "fc8" has three outputs in this adopted version equaling the number of classes or labels. The outputs are fed as input to the softMax layer, which exponentially normalizes them, thereby giving out a distribution of values across the three classes that add up to 1. Figure 5 shows the AlexNet architecture [51].

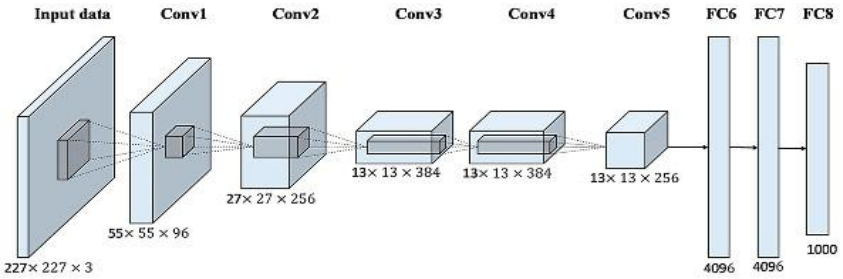


Figure 5. The AlexNet architecture

The weights of the final fully connected layer fc8 of the AlexNet network were re-initialized. In summary, a total of 2 experimental configurations were conducted on each model with T-O-T 1 (50%-20%-30%) and T-O-T 2 (60%-20%-20%). The SVM was implemented with the two kernel functions, linear and RBF. For DL, each experimental run was set to 12 iterations per epochs. Each epoch is the number of iterations for the neural network that completes a full-pass over the whole training data. Twelve epochs were substantially enough for the learning process to converge well-enough for classification. Furthermore, to allow for a valid, fair comparison with SVM, the DL was also constrained to a TOT of 50-20-30, and standardized training parameters were used as follows:

- Solver type: stochastic gradient descent (sgd),
- Base learning rate: 0.0001,
- Learning rate policy: step (decrease by a factor of 10),
- Weight decay: 0.004,
- Gamma: 0.9,
- Batch size: 100

3. RESULTS

It was observed that samples with early symptoms of the late blight disease were those that largely lead to the slight misclassification in both models. The SVM model was first trained and optimized on the 60-20-20 TOT distribution. The training data was mapped using the dot or linear kernel space with sequential minimal optimization (SMO) for separating the hyperplane. The same procedure was repeated for 50-20-30 distribution. The classification confusion matrixes are shown in Figure 6. The overall time taken for training and validation on the CPU was averagely 40 secs for both cases. The training achieved an average recall rate of 92.5% and a mean precision of 92.3%. Extracting the features is the most tasking process for the SVM model implementation, but the training and validation processes were comparably fast. Overall, SVM 60-20-20 achieved an F1-score of 88% and 99% for EB class, 90% and 92% for LB class with linear and RBF

kernels. While for SVM 50-20-30, 84% and 97% (EB class), 92% and 92% (LB class) with linear and RBF kernels, respectively. 100% rates were recorded for HL class all through.

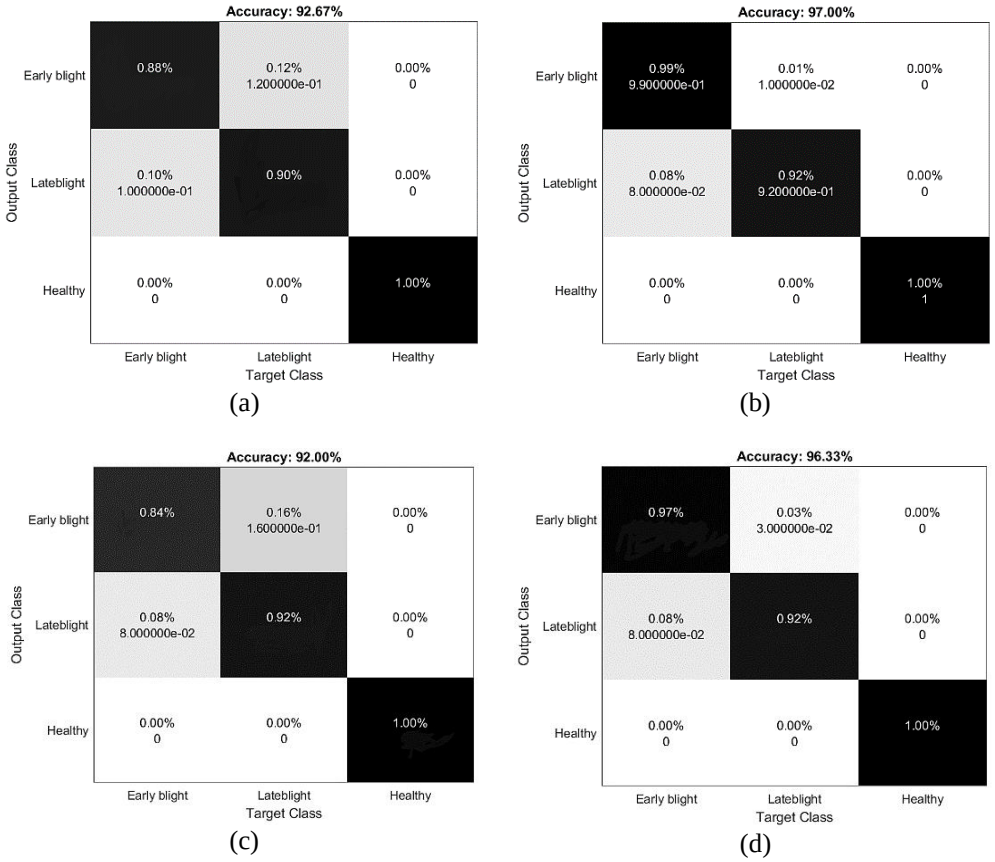


Figure 6. (a) Confusion Matrix for the SVM classifier on T-O-T 60-20-20 linear, (b) RBF, (c) T-O-T 50-20-30 linear, (d) RBF

Figure 7 shows the training chart for the DL model for the training period of 12 epochs (white and grey bars), which shows the two processes seemingly converge between the 6th and the 7th epoch. Despite the training and optimization running on the GPU, the running time was moderately fast given the amount of data, precisely 6 mins 46 secs (50-20-30) and 8 mins 12 secs (60-20-20). Regardless, both the TOT validation accuracies were at 98.14% (in the case of 60-20-20) and 97.44% (in the case of 50-20-30). Classification accuracy for the DL as shown in Figure 8 60-20-20 varied from 98.7% (EB class), 99.7% (LB class), to 95.7% (HL class). Accuracy for 50-20-30 from 88.2% (HL class) to 100% (for both EB and LB classes). Mean F1-scores were 96.1% 60-20-20 and 98.92% 50-20-30. The former F1-score is lower due to the resulting lower accuracy of HL classification, as shown in the confusion matrix in Figure 8.

In addition to T-O-T segments, the SVM was further evaluated on the quality and number of features using k-fold cross-validation on 60-20-20 distribution. The correlation of all features to the target was first computed. The features were then ranked following their correlation weight: the higher the correlation weight of a feature, the more exceptional quality it has as a discriminant for improved performance. Then the first subset of all raked features was first used to build and evaluate the classifier, and at each subsequent round, the least ranked feature is removed, and the process repeated until only the two most ranked features remained. The results are displayed in Table 2.

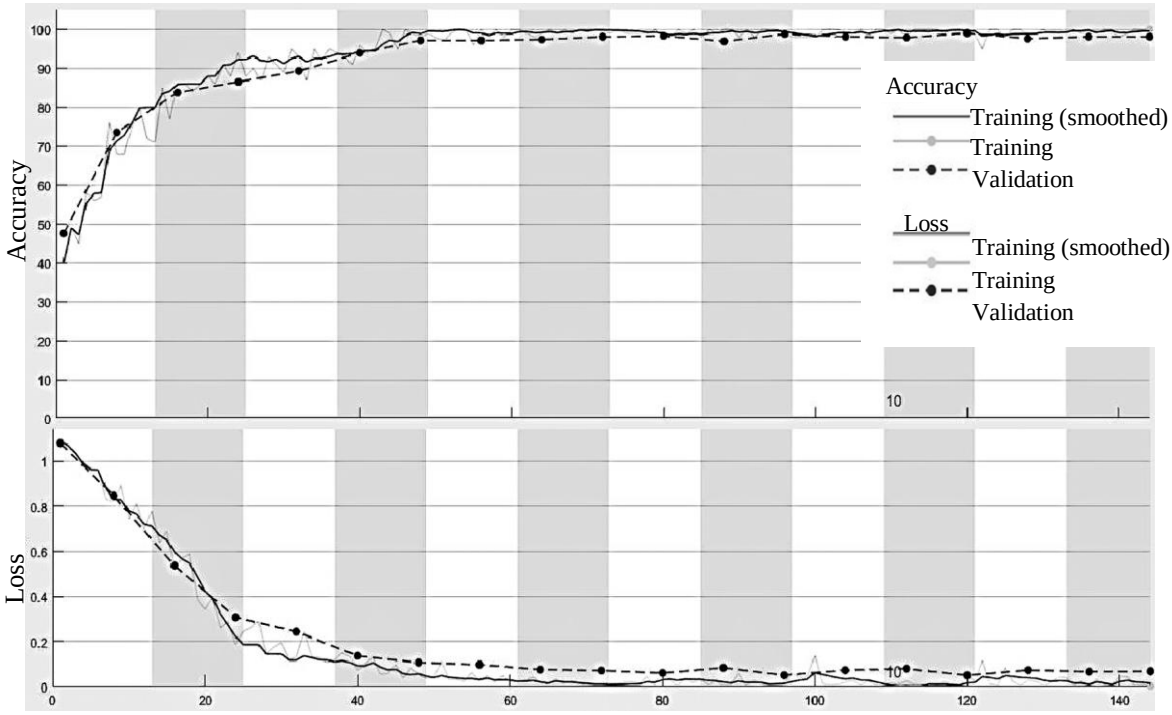
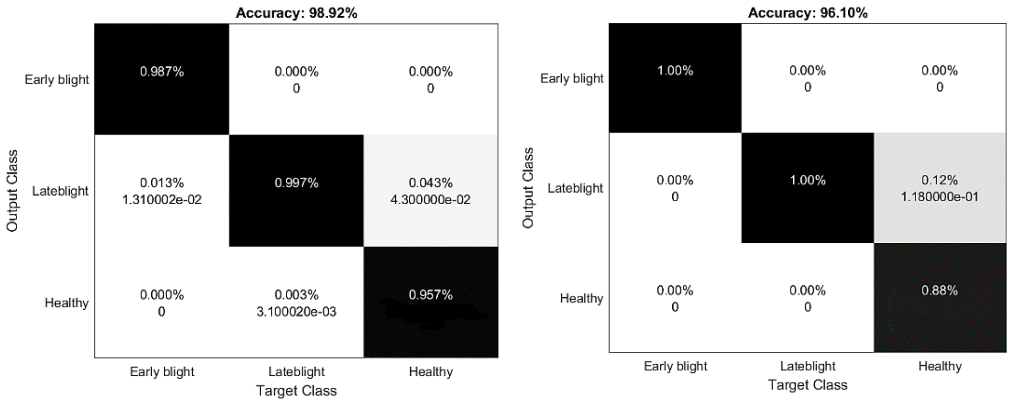


Figure 7. Training and validation progression for the DL network



(a) (b)

Figure 8. DL network Confusion Matrix for TOT: 50-20-30 (left), and TOT: 60-20-20 (right)

Table 2. SVM performance on different combination and number of feature subsets

Feature sub-set	*Accuracy - RBF (100%)*	*Accuracy - linear (100%)*
All 9	95.78	94.93
Best 8	95.78	94.99
Best 7	95.56	94.87
Best 6	96.67	96.21
Best 5	97.00	96.33
Best 4	96.22	96.01
Best 3	95.78	95.22
Best 2	95.56	94.96

4. ANALYSIS AND DISCUSSION

As results indicated, DL is superior in terms of performance and disease classification accuracy in this set-up due to its advanced architecture. However, with less training data (about 16% drop), the classification accuracies of EB and LB dropped by 2.8%. These results confirmed the DL and training data

relationship for optimum performance, which is the more massive the training samples, the more efficient and accurate the DL classifier becomes. The SVM recorded slightly lower classification accuracy than DL. It also showed a slight drop by 0.67% in the overall accuracy with reduced training data, which indicated a negligible effect compared to DL. A higher result margin was recorded between linear and RBF kernels. Accuracy significantly increased by about 5% with the RBF kernel, which indicated its superiority in separating the hyperplane. The SVM has a more simplified architecture design and much less computationally expensive in both training and classification. This is reflected in the different number of feature subsets' results, as shown in Table 2.

With a small limited amount of 5 qualitative features, the accuracies were higher at 96.33% and 97% for both linear and RBF kernels, respectively, and even closer to that of DL. Both SVM and DL rely on the quality of input data. However, SVM relies more on the crafted or pre-defined features, while the DL on the extensiveness of the data. It is essential to highlight that the classification between the EB and LB is the true measure of each classifier's performance. Both have achieved 100% classification accuracy in detecting whether a leaf is diseased or not. This indicates the relevance of quality characterization features to classifying disease symptoms with a high degree of similarity. It would also suggest that the concept of image processing-based plant disease detection must not just be about adjusting function weighs, settings, or kernels, generally tweaking of some parameters of architecture or model. Instead, engineering the features, no matter how complicated it may seem, often provides the best possible solutions for the diagnosis. This is inevitable as constant changes in natural phenomena such as temperature and humidity may virally cause spawn or the evolution of new plant disease symptoms [37]. Furthermore, handcrafted features allow users to properly observe the machine learning (model) behavior towards new patterns, particularly during training and optimization of robust models. Thus, many researchers continue to use conventional models such as SVM.

In summary, the results have shown DL requires tremendous building time, training time, higher computational power, and training data. When all these requirements are no longer considered as issues of limitation, then it would perhaps be the time to give more emphasis than currently being given to the DL. Hence, if there is a lack of the necessary computational power, i.e., a GPU, and a well-constrained data (few hundred to a thousand), then one might be better off with using the shallow SVM machine learning algorithm. On the contrary, however, DL would be a better choice. This experiment has also confirmed that among the power of the deep networks, the practical automatic feature extraction of the DL stores the features in one of the lower layers. Thus, the preserved features can be harnessed for use with shallow machine learning classifiers, such as the SVM. This perhaps answers the question of possibly combining, at a certain level, the DL and the SVM for better performance. As an extension of this work, it would be interesting to assess these features for classification.

Research focus and improvement suggestions

While the DL remains the focus of research as of late, its current application within this context is mostly towards future applications such as in smartphones, as recent studies would indicate [34]. Even with transfer learning, constant fine-tuning is required, such as batch normalization and learning rate stabilization. Lack of substantial training data and required computational power are what hinders its practical usage the most. Adding more training data and incorporating cloud computing would simplify some of these limitations. The complexity of the SVM model is found to be influenced by the segmentation and feature extraction processes, and it may well be a complicated and tedious process, particularly with extensive data (as encountered in this experiment). Lack of visible boundary edges that determines the separation boundary limit between color variant symptom lesions from healthy green tissue is what influence the effectiveness of segmentation result and the quality of the engineered features. In this regard, there has not been a unanimously acceptable ground truth for the use of segmentation methods. However, with supporting literature, best practice suggests color channel manipulation through thresholding is the least complex and adaptable to all crop species. The use of GLCM for extraction also weighs in some of the computational cost.

Nevertheless, with the incorporation of plant pathological inference and fusion of relevant features extracted individually from multiple disease symptom regions, machine learning classification can be improved. For SVM, the segmentation process can further be enhanced by utilizing disease region expansion. Expanding the ROI to cover the blurred region, a part of the healthy tissue right before where disease symptoms start to show, capture features corresponding to disease progression information, which can be leveraged for improved disease detection. This may also go a long way in capturing the vivid anatomy of a disease symptom progression, leading to disease severity and yield loss prediction.

5. CONCLUSION

The objectives of this study were to investigate through experimental approach the performance of two popular machine learning models, support vector machine (SVM) and deep learning (DL). The yielded results provided suggestions as to which of the two should be preferred for image processing-based plant leaf disease detection within the constraints of architectural complexity, computational power, and training data. The investigation has concluded that the DL model outperforms the SVM by a somewhat substantial margin in terms of classification accuracy with its advanced architecture using deep layers of convolution neural network. The observations from this study suggest that the DL is preferred for higher result accuracy when large training samples and GPU fitted computers are available with compromise on reducing computational complexity. On the contrary, SVM is most preferred for classification using small data and less or moderate computation power with no GPU requirement. Furthermore, SVM with an RBF kernel is best preferred, particularly when new features need to be observed, tuned, or crafted with less architecture complexity for optimum results. This work has offered clarification on the continuous usage of the machine learning model, SVM, and at the same time, opened several questions that need further investigation. Further work needs to be done to fully establish whether the method of DL on transfer learning should be an alternative to the handcrafted features to reduce the burden of manual extraction for the SVM.

ACKNOWLEDGMENT

The authors thank the Ministry of Education Malaysia and Universiti Teknologi Malaysia (UTM) for their support under the Flagship University Grant, grant number Q.J130000.2451.04G71.

REFERENCES

[1] A. S. Tripathy and D. K. Sharma, "Image Processing Techniques Aiding Smart Agriculture," in *Modern Techniques for Agricultural Disease Management and Crop Yield Prediction*: IGI Global, pp. 23-48, 2020.
[2] P. S. Ojiambo, J. Yuen, F. van den Bosch, and L. V. Madden, "Epidemiology: Past, Present, and Future Impacts on Understanding Disease Dynamics and Improving Plant Disease Management-A Summary of Focus Issue Articles," *Phytopathology*, vol. 107, no. 10, pp. 1092-1094, 2017/10/01 2017.
[3] S. Kaur, S. Pandey, and S. Goel, "Plants Disease Identification and Classification Through Leaf Images: A Survey," *Archives of Computational Methods in Engineering*, journal article January 19 2018.
[4] Y. Fang and R. P. Ramasamy, "Current and Prospective Methods for Plant Disease Detection," (in eng), *Biosensors*, vol. 5, no. 3, pp. 537-561, 2015.
[5] J. G. A. Barbedo, "A review on the main challenges in automatic plant disease identification based on visible range images," *Biosystems Engineering*, vol. 144, pp. 52-60, 2016.
[6] T. U. Rehman, M. S. Mahmud, Y. K. Chang, J. Jin, and J. Shin, "Current and future applications of statistical machine learning algorithms for agricultural machine vision systems," *Computers and Electronics in Agriculture*, vol. 156, pp. 585-605, 2019/01/01/ 2019.
[7] S. Sankaran, A. Mishra, R. Ehsani, and C. Davis, "A review of advanced techniques for detecting plant diseases," *Computers and Electronics in Agriculture*, vol. 72, no. 1, pp. 1-13, 2010/06/01/ 2010.
[8] J. G. A. Barbedo, "Digital image processing techniques for detecting, quantifying and classifying plant diseases," *SpringerPlus*, vol. 2, no. 1, p. 660, 2013.
[9] A. K. Singh, B. Ganapathysubramanian, S. Sarkar, and A. Singh, "Deep learning for plant stress phenotyping: trends and future perspectives," *Trends in plant science*, vol. 23, no. 10, pp. 883-898, 2018.
[10] A. Kamilaris and F. X. Prenafeta-Boldú, "Deep learning in agriculture: A survey," *Computers and electronics in agriculture*, vol. 147, pp. 70-90, 2018.
[11] R. Karthik, M. Hariharan, S. Anand, P. Mathikshara, A. Johnson, and R. Menaka, "Attention embedded residual CNN for disease detection in tomato leaves," *Applied Soft Computing*, p. 105933, 2019.
[12] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *nature*, vol. 521, no. 7553, p. 436, 2015.
[13] K. Mochida *et al.*, "Computer vision-based phenotyping for improvement of plant productivity: a machine learning perspective," *GigaScience*, vol. 8, no. 1, p. giy153, 2018.
[14] Y. Toda and F. Okura, "How convolutional neural networks diagnose plant disease," *Plant Phenomics*, vol. 2019, 2019.
[15] A. Singh, B. Ganapathysubramanian, A. K. Singh, and S. Sarkar, "Machine Learning for High-Throughput Stress Phenotyping in Plants," *Trends in Plant Science*, vol. 21, no. 2, pp. 110-124, 2016/02/01/ 2016.
[16] P. Kaur, S. Singla, and S. Singh, "Detection and classification of leaf diseases using integrated approach of support vector machine and particle swarm optimization," *INTERNATIONAL JOURNAL OF ADVANCED AND APPLIED SCIENCES*, vol. 4, no. 8, pp. 79-83, 2017.
[17] J. G. Barbedo, "Factors influencing the use of deep learning for plant disease recognition," *Biosystems engineering*, vol. 172, pp. 84-91, 2018.
[18] P. K. Gikunda and N. Jouandeau, "Modern CNNs for IoT Based Farms," in *International Conference on Information and Communication Technology for Development for Africa*, 2019, pp. 68-79: Springer.

[19] S. H. Lee, H. Goëau, P. Bonnet, and A. Joly, "New perspectives on plant disease characterization based on deep learning," *Computers and Electronics in Agriculture*, vol. 170, p. 105220, 2020/03/01/ 2020.
[20] Z. Iqbal, M. A. Khan, M. Sharif, J. H. Shah, M. H. ur Rehman, and K. Javed, "An automated detection and classification of citrus plant diseases using image processing techniques: A review," *Computers and Electronics in Agriculture*, vol. 153, pp. 12-32, 2018/10/01/ 2018.
[21] S. S. Chouhan, U. P. Singh, and S. Jain, "Applications of Computer Vision in Plant Pathology: A Survey," *Archives of Computational Methods in Engineering*, pp. 1-22, 2019.
[22] A. Camargo and J. Smith, "Image pattern classification for the identification of disease causing agents in plants," *Computers and Electronics in Agriculture*, vol. 66, no. 2, pp. 121-125, 2009.
[23] A. Camargo and J. Smith, "An image-processing based algorithm to automatically identify plant disease visual symptoms," *Biosystems engineering*, vol. 102, no. 1, pp. 9-21, 2009.
[24] A. A. Bernardes *et al.*, "Identification of Foliar Diseases in Cotton Crop," in *Topics in Medical Image Processing and Computational Vision*, J. M. R. S. Tavares and R. M. Natal Jorge, Eds. Dordrecht: Springer Netherlands, 2013, pp. 67-85.
[25] J. G. A. Barbedo, L. V. Koenigkan, and T. T. Santos, "Identifying multiple plant diseases using digital image processing," *Biosystems Engineering*, vol. 147, pp. 104-116, 2016.
[26] V. Singh and A. Misra, "Detection of plant leaf diseases using image segmentation and soft computing techniques," *Information Processing in Agriculture*, vol. 4, no. 1, pp. 41-49, 2017.
[27] G. Dhingra, V. Kumar, and H. D. Joshi, "A novel computer vision based neutrosophic approach for leaf disease identification and classification," *Measurement*, vol. 135, pp. 782-794, 2019/03/01/ 2019.
[28] N. Wu, M. Li, L. Chen, Y. Yuan, and S. Song, "A LDA-based segmentation model for classifying pixels in crop diseased images," in *Control Conference (CCC), 2017 36th Chinese*, 2017, pp. 11499-11505: IEEE.
[29] P. Xu, G. Wu, Y. Guo, H. Yang, and R. Zhang, "Automatic wheat leaf rust detection and grading diagnosis via embedded image processing system," *Procedia Computer Science*, vol. 107, pp. 836-841, 2017.
[30] S. P. Mohanty, D. P. Hughes, and M. Salathé, "Using deep learning for image-based plant disease detection," *Frontiers in plant science*, vol. 7, p. 1419, 2016.
[31] S. Sladojevic, M. Arsenovic, A. Anderla, D. Culibrk, and D. Stefanovic, "Deep neural networks based recognition of plant diseases by leaf image classification," *Computational intelligence and neuroscience*, vol. 2016, 2016.
[32] O. Russakovsky *et al.*, "Imagenet large scale visual recognition challenge," *International Journal of Computer Vision*, vol. 115, no. 3, pp. 211-252, 2015.
[33] A. Fuentes, S. Yoon, S. C. Kim, and D. S. Park, "A robust deep-learning-based detector for real-time tomato plant diseases and pests recognition," *Sensors*, vol. 17, no. 9, p. 2022, 2017.
[34] P. Sharma, Y. Paul Singh Berwal, and W. Ghai, "Performance Analysis of Deep Learning CNN Models for Disease Detection in Plants using Image Segmentation," *Information Processing in Agriculture*, 2019/11/18/ 2019.
[35] J. G. A. Barbedo, "Plant disease identification from individual lesions and spots using deep learning," *Biosystems Engineering*, vol. 180, pp. 96-107, 2019.
[36] M. A. Khan *et al.*, "An Optimized Method for Segmentation and Classification of Apple Diseases Based on Strong Correlation and Genetic Algorithm Based Feature Selection," *IEEE Access*, vol. 7, pp. 46261-46277, 2019.
[37] R. Chaerani and R. E. Voorrips, "Tomato early blight (Alternaria solani): the pathogen, genetics, and breeding for resistance," *Journal of general plant pathology*, vol. 72, no. 6, pp. 335-347, 2006.
[38] M. Nowicki, E. U. Kozik, and M. R. Foolad, "Late blight of tomato," *Translational genomics for crop breeding*, vol. 1, pp. 241-265, 2013.
[39] A. M. Abdu, M. M. Mokji, U. U. Sheikh, and K. Khalil, "Automatic Disease Symptoms Segmentation Optimized for Dissimilarity Feature Extraction in Digital Photographs of Plant Leaves," in *2019 IEEE 15th International Colloquium on Signal Processing & Its Applications (CSPA)*, 2019, pp. 60-64.
[40] M. M. Mokji and S. A. R. A. Bakar, "Gray Level Co-Occurrence Matrix Computation Based On Haar Wavelet," in *Computer Graphics, Imaging and Visualisation (CGIV 2007)*, 2007, pp. 273-279.
[41] S. Arivazhagan, R. N. Shebiah, S. Ananthi, and S. V. Varthini, "Detection of unhealthy region of plant leaves and classification of plant leaf diseases using texture features," *Agricultural Engineering International: CIGR Journal*, vol. 15, no. 1, pp. 211-217, 2013.
[42] M. A. Hearst, S. T. Dumais, E. Osuna, J. Platt, and B. Scholkopf, "Support vector machines," *IEEE Intelligent Systems and their applications*, vol. 13, no. 4, pp. 18-28, 1998.
[43] "Learn Support Vector Machine from Scratch," (in English), *Web access "https://www.datasciencecentral.com/profiles/blogs/learn-support-vector-machine-svm-from-scratch-in-r"*, Online source 2018.
[44] MATLAB, "Introducing Deep Leraning with MATLAB," (in English), Book pp. 1-15, 2017.
[45] I. Goodfellow, Y. Bengio, A. Courville, and Y. Bengio, *Deep learning*. MIT press Cambridge, 2016.
[46] J. G. A. Barbedo, "Impact of dataset size and variety on the effectiveness of deep learning and transfer learning for plant disease classification," *Computers and electronics in agriculture*, vol. 153, pp. 46-53, 2018.
[47] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097-1105.
[48] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.
[49] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770-778.

[50] D. C. Ciresan, U. Meier, J. Masci, L. Maria Gambardella, and J. Schmidhuber, "Flexible, high performance convolutional neural networks for image classification," in *IJCAI Proceedings-International Joint Conference on Artificial Intelligence*, 2011, vol. 22, no. 1, p. 1237: Barcelona, Spain.
[51] X. Han, Y. Zhong, L. Cao, and L. Zhang, "Pre-Trained AlexNet Architecture with Pyramid Pooling and Supervision for High Spatial Resolution Remote Sensing Image Scene Classification," *Remote Sensing*, vol. 9, no. 8, p. 848, 2017.

BIOGRAPHIES OF AUTHORS

Aliyu Muhammad Abdu received his B.Eng. in Electrical Engineering from Kano University of Science and Technology (KUST) in 2009 and his M. Eng. degree in Electrical Engineering from Universiti Teknologi Malaysia (UTM) in 2014. His research interest includes Digital Image Processing, Machine Learning, Computer Vision and Pattern Recognition. His current research is in the application of these models to plant disease detection. Aliyu is currently a Ph. D student, member of the Computer Vision, Video, and Image Processing (CVVIP) research group of the department of School of Electrical Engineering, UTM.

Dr. Musa Mohd Mokji received his B.Eng. in Electrical Engineering from Universiti Teknologi Malaysia. Then he received his M.Eng. and Ph.D. degrees specializing in Image Processing from the same university in 2001 and 2008, respectively. He is currently a senior lecturer at the Faculty of Engineering, UTM with a research interest in Signal and Image Processing, Pattern Recognition and Datamining. He is also interested in the application of these models to agriculture, surveillance system, document processing, and medical. Dr. Musa is the head of the Digital Signal and Image Processing research group at the Universiti Teknologi Malaysia. He gives undergraduate and postgraduate lectures on signal processing and image processing at Universiti Teknologi Malaysia.

Dr. Usman Ullah Sheikh received his Ph.D. degree (2009) in image processing and computer vision from Universiti Teknologi Malaysia. His research work is mainly on computer vision and embedded systems design. He is currently a Senior Lecturer at Universiti Teknologi Malaysia, Malaysia.