

To study, analysis and implementation of Fruit/Leave Disease Detection using image processing and neural approach

¹Rupesh Kumar, ²Suresh S. Gawande, ³Satyarthi Tiwari

^{1,2,3} Bhabha engineering and research institute (BERI), Bhopal , India

Abstract

The latest generation of convolution neural networks (CNNs) has achieved impressive results in the field of image classification. This paper is concerned with a new approach to the development of fruit/plant disease detection model, based on leaf image processing and classification, by the use of ANN. Novel way of training and the methodology used facilitate a quick and easy system implementation in practice. The developed model is able to recognize 13 different types of plant diseases out of healthy leaves, with the ability to distinguish plant leaves from their surroundings. According to our knowledge, this method for plant disease recognition has been proposed for the first time. All essential steps required for implementing this disease recognition model are fully described throughout the paper, starting from gathering images in order to create a database, assessed by agricultural experts. Caffe, a deep learning framework developed by Berkley Vision and Learning Centre, was used to perform the deep CNN training.

1. Introduction

The problem of efficient plant disease protection is closely related to the problems of sustainable agriculture and climate change [1]. Research results indicate that climate change can alter stages and rates of pathogen development; it can also modify host resistance, which leads to physiological changes of host-pathogen interactions [2, 3]. The situation is further complicated by the fact that, today, diseases are transferred globally more easily than ever before. New diseases can occur in places where they were previously unidentified and, inherently, where there is no local expertise to combat them [4–6]. Inexperienced pesticide usage can cause the development of long-term resistance of the pathogens, severely reducing the ability to fight back. Timely and accurate diagnosis of plant diseases is one of the pillars of precision agriculture [7]. It is crucial to prevent unnecessary waste of financial and

other resources, thus achieving healthier production, by addressing the long-term pathogen resistance development problem and mitigating the negative effects of climate change. In this changing environment, appropriate and timely disease identification including early prevention has never been more important. There are several ways to detect plant pathologies. Some diseases do not have any visible symptoms, or the effect becomes noticeable too late to act, and in those situations, a sophisticated analysis is obligatory. However, most diseases generate some kind of manifestation in the visible spectrum, so the naked eye examination of a trained professional is the prime technique adopted in practice for plant disease detection. In order to achieve accurate plant disease diagnostics a plant pathologist should possess good observation skills so that one can identify characteristic symptoms [8]. Variations in symptoms indicated by diseased plants may lead to an improper diagnosis since amateur gardeners and hobbyists could have more difficulties determining it than a professional plant pathologist. An automated system designed to help identify plant diseases by the plant's appearance and visual symptoms could be of great help to amateurs in the gardening process and also trained professionals as a verification system in disease diagnostics. Advances in computer vision present an opportunity to expand and enhance the practice of precise plant protection and extend the market of computer vision applications in the field of precision agriculture. Exploiting common digital image processing techniques such as colour analysis and thresholding [9] were used with the aim of detection and classification of plant diseases. Various different approaches are currently used for detecting plant diseases and most common are artificial neural networks (ANNs) [10] and Support Vector Machines (SVMs) [11]. They are combined with different methods of image preprocessing in favour of better

feature extraction. In machine learning and cognitive science, ANN is an information-processing paradigm that was inspired by the way biological nervous systems, such as the brain, process information. The brain is composed of a large number of highly interconnected neurons working together to solve specific problems. An artificial neuron is a processing element with many inputs and one output. Although artificial neurons can have many outputs, only those with exactly one output will be considered. Their inputs can also take on any value between 0 and 1. Also, the neuron has weights for each input and an overall bias. The weights are real numbers expressing importance of the respective inputs to the output. The bias is used for controlling how easy the neuron is getting to output 1. For a neuron with really big bias it is easy to output 1, but when the bias is very negative then it is difficult to output 1. The output of the neuron is not 0 or 1. Instead, it is $\alpha \cdot (w \cdot x + b)$, where α is called the transfer function. There are different types of transfer function: step, linear, sigmoid, and so forth. The smoothness of α means that small changes Δw_j in the weights and Δb in the bias will produce small change Δoutput in the output from the neuron. Small output change is approximated by

$$\Delta \text{output} \approx \sum \frac{\partial \text{output}}{\partial w_j} \Delta w_j + \sum \frac{\partial \text{output}}{\partial b} \Delta b.$$

Basically, the small change in weight or bias causes the small corresponding change in the network output (Figure 1). Neural networks, with their outstanding ability to derive meaning from complex or imperfect data, can be applied for extracting patterns and detecting trends that are too difficult to notice by humans or computer techniques. Other advantages of ANNs are adaptive learning, self-organization, real time operations, and so forth. There are two main categories of ANNs when speaking about architecture: feed-forward ANNs where the output of any layer is unlikely to influence that same layer and feedback ANNs where signals travel in both directions by involving loops in the network.

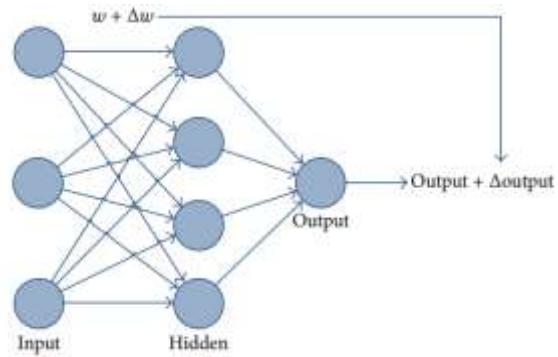


Figure 1: Simple model of ANN. The method described in this paper is a new approach in detecting plant diseases using the deep convolutional neural network trained and fine-tuned to fit accurately to the database of a plant's leaves that was gathered independently for diverse plant diseases. The advance and novelty of the developed model lie in its simplicity; healthy leaves and background images are in line with other classes, enabling the model to distinguish between diseased leaves and healthy ones or from the environment by using deep CNN.

2. Literature Review

Implementing the appropriate management strategies like fungicide applications, disease-specific chemical applications, and vector control through pesticide applications could lead to early information on crop health and disease detection. This could facilitate the control of diseases and improve productivity. In [12], authors present, review, and recognize the demand for developing a rapid, cost-effective, and reliable health-monitoring sensor that facilitates advancements in agriculture. They described the currently used technologies that include spectroscopic and imaging-based and volatile profiling-based plant disease detection methods for the purpose of developing ground-based sensor system to assist in monitoring health and diseases in plants under field conditions. After analysis of their work and analysis presented by the authors of [13-16], it was decided to use image processing disease recognition approach among other approaches commonly used for plant disease diagnostics, for instance, double-stranded ribonucleic acid (RNA) analysis, nucleic acid probes, and microscopy.

Numerous procedures are currently in use for plant disease detection applying computer vision. One of them is disease detection by extracting colour feature as authors in [17] have presented. In this paper YcbCr, HSI, and CIELB colour models were used in the study; as a result, disease spots were successfully detected and remained unaffected by the noise from different sources, such as camera flash. In addition, plant disease detection could be achieved by extracting shape features method. Patil and Bodhe applied this technique for disease detection in sugarcane leaves where they have used threshold segmentation to determine leaf area and triangle threshold for lesioning area, getting the average accuracy of 98.60% at the final experiments [18]. Furthermore, extracting texture feature could be used in detecting plant diseases. Patil and Kumar proposed a model for plant disease detection using texture features such as inertia, homogeneity, and correlation obtained by calculating the gray level cooccurrence matrix on image [19]. Combined with colour extraction, they experimented on detecting diseases on maize leaves. Combination of all these features provides a robust feature set for image improvement and better classification. In [20], the authors have presented a survey of well-known conventional methods of feature extraction. Due to the rapid progress of Artificial Intelligence (AI) science, work in this paper is mainly focused on applying these methodologies and techniques. There are some approaches which apply the feed-forward back propagation of neural networks consisting of one input, one output, and one hidden layer for the needs of identifying the species of leaf, pest, or disease; this model was proposed by the authors in [21]. They developed a software model, to suggest remedial measures for pest or disease management in agricultural crops. Another technique proposed by the authors in [22] incorporates the features extracted by Particle Swarm Optimization (PSO) [23] and forward neural network in direction of determining the injured leaf spot of cotton and improving the accuracy of the system with the final overall accuracy of 95%. Also, detection and differentiation of plant diseases can be achieved using Support Vector Machine algorithms. This technique was implemented for sugar beet diseases and presented in [24], where, depending on the type and stage of disease, the classification accuracy was between 65% and 90%.

Likewise, there are methods that combine the feature extraction and Neural Network Ensemble (NNE) for plant disease recognition. Through training a definite number of neural networks and combining their results after that, NNE offers a better generalization of learning ability [25]. Such method was implemented only for recognizing tea leaf diseases with final testing accuracy of 91% [26]. Another approach based on leaf images and using ANNs as a technique for an automatic detection and classification of plant diseases was used in conjunction with *K*-means as a clustering procedure proposed by the authors in [27]. ANN consisted of 10 hidden layers. The number of outputs was 6 which was the number of classes representing five diseases along with the case of a healthy leaf. On average, the accuracy of classification using this approach was 94.67%. The authors in [28–31] presented the deep learning methods for solving most complex tasks in different areas of research in biology, bioinformatics, biomedicine, robotics, and 3D technologies.

3. Basic System Model

The entire procedure of developing the model for plant disease recognition using deep CNN is described further in detail. The complete process is divided into several necessary stages in subsections below, starting with gathering images for classification process using deep neural networks.

3.1. Dataset. Appropriate datasets are required at all stages of object recognition research, starting from training phase to evaluating the performance of recognition algorithms. All the images collected for the dataset were downloaded from the Internet, searched by disease and plant name on various sources in different languages, such as Latin, English, German, Serbian, and Hungarian. Images in the dataset were grouped into fifteen different classes. Thirteen classes represented plant diseases which could be visually determined from leaves. In order to distinguish healthy leaves from diseased ones, one more class was added in the dataset. It contains only images of healthy leaves. An extra class in the dataset with background images was beneficial to get more accurate classification. Thus, deep neural network could be trained to differentiate the leaves from the surrounding. The background images were taken from the Stanford background

dataset [32]. In this stage, all duplicated images taken from different sources were removed by developed python script applying the comparing procedure. The script removed the duplicates by comparing the images' metadata: name, size, and the date. After the automated removal, images were assessed by human experts in much iteration. Next step was to enrich the dataset with augmented images. The main goal of the presented study is to train the network to learn the features that distinguish one class from the others. Therefore, when using more augmented images, the chance for the network to learn the appropriate features has been increased. Finally, a database containing 30880 images for training and 2589 images for validation has been created. The augmentation process is described in Section 3.3. all supported diseases together with the number of original images and number of augmented images for every class used as training and validation dataset for the disease classification model.

3.2. Image Preprocessing and Labelling. Images downloaded from the Internet were in various formats along with different resolutions and quality. In order to get better feature extraction, final images intended to be used as dataset for deep neural network classifier were preprocessed in order to gain consistency. Furthermore, procedure of image preprocessing involved cropping of all the images manually, making the square around the leaves, in order to highlight the region of interest (plant leaves). During the phase of collecting the images for the dataset, images with smaller resolution and dimension less than 500 px were not considered as valid images for the dataset. In addition, only the images where the region of interest was in higher resolution were marked as eligible candidates for the dataset. In that way, it was ensured that images contain all the needed information for feature learning. Images used for the dataset were image resized to 256×256 to reduce the time of training, which was automatically computed by written script in Python, using the OpenCV framework [33]. Many resources can be found by searching across the Internet, but their relevance is often unreliable. In the interest of confirming the accuracy of classes in the dataset, initially grouped by a keywords search, agricultural experts examined leaf images and labelled all the images with appropriate disease acronym. As it is known, it is important to use

accurately classified images for the training and validation dataset. Only in that way may an appropriate and reliable detecting model be developed. In this stage, duplicated images that were left after the initial iteration of gathering and grouping images into classes described in Section 3.1 were removed from the dataset.

3.3. Augmentation Process. The main purpose of applying augmentation is to increase the dataset and introduce slight distortion to the images which helps in reducing overfitting during the training stage. In machine learning, as well as in statistics, overfitting appears when a statistical model describes random noise or error rather than underlying relationship [34]. The image augmentation contained one of several transformation techniques including affine transformation, perspective transformation, and simple image rotations. Affine transformations were applied to express translations and rotations (linear transformations and vector addition, resp.) [35] where all parallel lines in the original image are still parallel in the output image. To find a transformation matrix, three points from the original image were needed as well as their corresponding locations in the output image. For perspective transformation, a 3×3 transformation matrix was required. Straight lines would remain straight even after the transformation. For the augmentation process, simple image rotations were applied, as well as rotations on the different axis by various degrees. Transformations applied in augmentation process are illustrated in Figure 2, where the first row represents resulting images obtained by applying affine transformation on the single image; the second row represents images obtained from perspective transformation against the input image and the last row visualizes the simple rotation of the input image. The process of augmentation was chosen to fit the needs; the leaves in a natural environment could vary in visual perspective. For this stage, in order to automate the augmentation process for numerous images from the dataset, particular application was developed in C++ using the OpenCV library [36], with possibility of changing the parameters of transformation during the run-time, which improves flexibility.

3.4. Neural Network Training. Training the deep convolution neural network for making an image

classification model from a dataset described in Section 3.1 was proposed

Each convolutional layer has M maps of equal size, M_x and M_y , and a kernel of size K_x , and K_y is shifted over the certain region of the input image. The skipping factors S_x and S_y define how many pixels the filter/kernel skips in x - and y -direction between subsequent convolutions [46]. The size of the output map could be defined as

$$M_x^n = \frac{M_x^{n-1} - K_x^n}{S_x^n + 1} + 1, \quad (2)$$

$$M_y^n = \frac{M_y^{n-1} - K_y^n}{S_y^n + 1} + 1,$$

where n indicates the layer. Each map in layer L^n is connected to most M^{n-1} maps in layer L^{n-1} .

Rectified Linear Units (ReLU) are used as substitute for saturating nonlinearities. This activation function adaptively learns the parameters of rectifiers and improves accuracy at negligible extra computational cost [47]. It is defined as

$$f(z_i) = \max(0, z_i), \quad (3)$$

where z_i represents the input of the nonlinear activation function f on the i th channel.

Deep CNN with ReLUs trains several times faster. This method is applied to the output of every convolutional and fully connected layer. Despite the output, the input normalization is not required; it is applied after ReLU nonlinearity after the first and second convolutional layer because it reduces top-1 and top-5 error rates. In CNN, neurons within a hidden layer are segmented into “feature maps.” The neurons within a feature map share the same weight and bias. The neurons within the feature map search for the same feature. These neurons are unique since they are connected to different neurons in the lower layer. So for the first hidden layer, neurons within a feature map will be connected to different regions of the input image. The hidden layer is segmented into feature maps where each neuron in a feature map looks for the same feature but at different positions of the input image. Basically, the feature map is the result of applying convolution across an image. Each layer’s features are displayed in a different block, where visualization represents the strongest activation for the provided feature map, starting from the first convolutional layer, where features go from individual pixels to simple lines, to the fifth convolutional layer where learned features like shapes and certain parts of leaves are displayed (Figure 3). Another important layer of CNNs is the pooling layer, which is a form of nonlinear

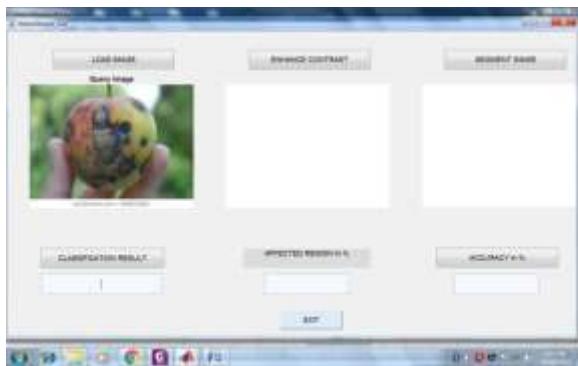
downsampling. Pooling operation gives the form of translation invariance [48]; it operates independently on every depth slice of the input and resizes it spatially. Overlapping pooling is beneficially applied to lessen overfitting. Also in favour of reducing overfitting, a dropout layer [49] is used in the first two fully connected layers. But the shortcoming of dropout is that it increases training time 2-3 times comparing to a standard neural network of the exact architecture [50]. Bayesian optimization experiments also proved that ReLUs and dropout have synergy effects, which means that it is advantageous when they are used together [51]. The advance of CNNs refer to their ability to learn rich mid-level image representations as opposed to handdesigned low-level features used in other image classification methods [52]. Figure 4 illustrates the filtered output images after every convolutional and pooling layer of the deep network. Output images are labelled with the name of corresponding layer at the bottom right corner of every image.

3.5. Performed Tests. The common approach in measuring performance of artificial neural networks is splitting data into the training set and the test set and then training a neural network on the training set and using the test set for prediction. Thus, since the original outcomes for the testing set and our model predicted outcomes are known, the accuracy of our prediction can be calculated. Different tests were performed with 2589 original images, when trained with 30880 images from database. For the accuracy test, 10-fold cross validation technique was used to evaluate a predictive model. The cross validation procedure was repeated after every thousand training iteration. Overall estimated result of the test is graphically represented as top-1, to test if the top class (the one having the highest probability) is the same as the target label. The top-5 error rate is there to test if the target label is one of the top 5 predictions, the ones with 5 of the highest probabilities. The number of images used for the validation test from each labelled class is given in Table 1. Test results are presented in Section 4, for both complete dataset and each class separately.

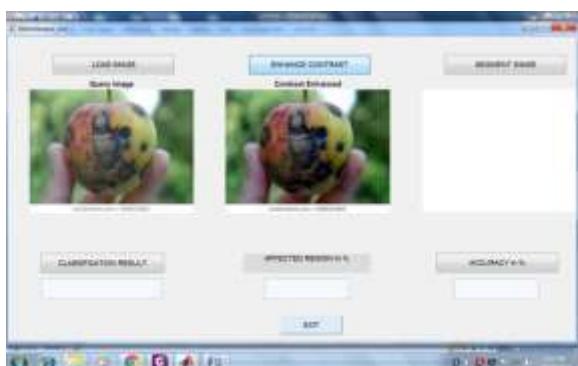
3.6. Fine-Tuning. Fine-tuning seeks to increase the effectiveness or efficiency of a process or function by making small modifications to improve or optimize

the outcome. The classification function in the original CaffeNet model is Softmax classifier that computes probability of 1,000 classes of the ImageNet dataset. Fine-tuned learning experiments require a bit of learning, but they are still much faster than learning from scratch [53]. To start the fine-tuning procedure, this softmax classifier was removed, as mentioned and illustrated in Section 3.4 and the new one was initialized with random values. The new softmax classifier was trained from scratch using the back-propagation algorithm with data from the dataset described in Section 3.1. This dataset has 15 different categories [43]. Due to the smaller size of the dataset used for this research when compared to ImageNet, ILSVRC-2012 [54], overfitting was constrained by using lower initial learning rates for the fine-tuned hidden layers [55]. The learning rate of the top layer was set to 10, while the learning rate of all the other seven learning layers was 0.1. The back-propagation algorithm ran for 100,000 iterations. The process of finetuning was repeated changing parameters of hidden layers and hyperparameters.

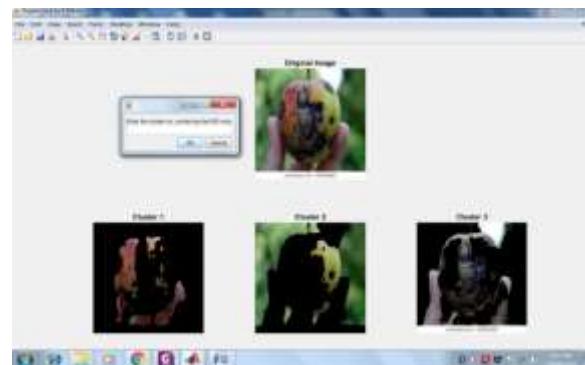
4. Results and Discussion



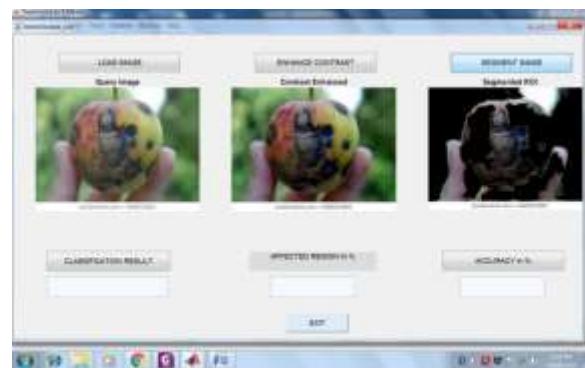
(a) selection of image(fruit/leaves)



(b) image with enhanced contrast



(c) various cluster images for selection

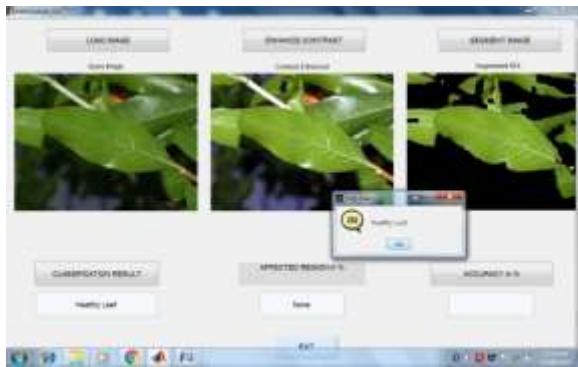


(d) segmented image with Region of Infection



(e) final results of affected region[37% affected]

In this section we have shown different results obtained from simulation, Results includes various infected images identification, recognition and showing the amount of infected area using MATLAB simulation.



(f)images of healthy leaves



(g)final image of infected leave[15%]

5. Conclusion

In this paper, it is proposed to find how much percent the fruit/plants is affected and recognize the defected fruit in the given image. This feature is very useful for the farmers and useful for different purposes. To get better results in the classification and identification of fruit diseases ANN model and image processing are used. We have added this feature to model so that the fusion of image processing and deep neural network not only useful for the fruit disease detection but also it is useful detecting diseases in vegetables and plants and completely helpful for the agricultural industry.

References

- [1] K. A. Garrett, S. P. Dendy, E. E. Frank, M. N. Rouse, and S. E. Travers, "Climate change effects on plant disease: genomes to ecosystems," *Annual Review of Phytopathology*, vol. 44, pp. 489–509, 2006.
- [2] S. M. Coakley, H. Scherm, and S. Chakraborty, "Climate change and plant disease management," *Annual Review of Phytopathology*, vol. 37, no. 1, pp. 399–426, 1999.
- [3] S. Chakraborty, A. V. Tiedemann, and P. S. Teng, "Climate change: potential impact on plant diseases,"

Environmental Pollution, vol. 108, no. 3, pp. 317–326, 2000.

- [4] A. J. Tatem, D. J. Rogers, and S. I. Hay, "Global transport networks and infectious disease spread," *Advances in Parasitology*, vol. 62, pp. 293–343, 2006.
- [5] J. R. Rohr, T. R. Raffel, J. M. Romansic, H. McCallum, and P. J. Hudson, "Evaluating the links between climate, disease spread, and amphibian declines," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 105, no. 45, pp. 17436–17441, 2008.
- [6] T. Van der Zwet, "Present worldwide distribution of fire blight," in *Proceedings of the 9th International Workshop on Fire Blight*, vol. 590, Napier, New Zealand, October 2001.
- [7] S. A. Miller, F. D. Beed, and C. L. Harmon, "Plant disease diagnostic capabilities and networks," *Annual Review of Phytopathology*, vol. 47, pp. 15–38, 2009.
- [8] M. B. Riley, M. R. Williamson, and O. Maloy, "Plant disease diagnosis. The Plant Health Instructor," 2002.
- [9] J. G. Arnal Barbedo, "Digital image processing techniques for detecting, quantifying and classifying plant diseases," *SpringerPlus*, vol. 2, article 660, pp. 1–12, 2013.
- [10] H. Cartwright, Ed., *Artificial Neural Networks*, Humana Press, 2015.
- [11] I. Steinwart and A. Christmann, *Support Vector Machines*, Springer Science & Business Media, New York, NY, USA, 2008.
- [12] 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.
- [13] P. R. Reddy, S. N. Divya, and R. Vijayalakshmi, "Plant disease detection technique—tool—a theoretical approach," *International Journal of Innovative Technology and Research*, pp. 91–93, 2015.

- [14] A.-K. Mahlein, T. Rumpf, P. Welke et al., "Development of spectral indices for detecting and identifying plant diseases," *Remote Sensing of Environment*, vol. 128, pp. 21–30, 2013.
- [15] W. Xiuqing, W. Haiyan, and Y. Shifeng, "Plant disease detection based on near-field acoustic holography," *Transactions of the Chinese Society for Agricultural Machinery*, vol. 2, article 43, 2014.

- [16] A.-K. Mahlein, E.-C. Oerke, U. Steiner, and H.-W. Dehne, "Recent advances in sensing plant diseases for precision crop protection," *European Journal of Plant Pathology*, vol. 133, no. 1, pp. 197–209, 2012.
- [17] P. Chaudhary, A. K. Chaudhari, A. N. Cheeran, and S. Godara, "Color transform based approach for disease spot detection on plant leaf," *International Journal of Computer Science and Telecommunications*, vol. 3, no. 6, pp. 65–69,

2012.

- [18] S. B. Patil and S. K. Bodhe, "Leaf disease severity measurement using image processing," *International Journal of Engineering and Technology*, vol. 3, no. 5, pp. 297–301, 2011.
- [19] J. K. Patil and R. Kumar, "Feature extraction of diseased leaf images," *Journal of Signal & Image Processing*, vol. 3, no. 1, p.60, 2012.
- [20] T. R. Reed and J. M. H. Dubuf, "A review of recent texture segmentation and feature extraction techniques," *CVGIP: Image Understanding*, vol. 57, no. 3, pp. 359–372, 1993.
- [21] M. S. P. Babu and B. Srinivasa Rao, "Leaves recognition using back propagation neural network-advice for pest and disease control on crops," *IndiaKisan. Net: Expert Advisory System*, 2007.
- [22] P. Revathi and M. Hemalatha, "Identification of cotton diseases based on cross information gain deep forward neural network classifier with PSO feature selection," *International Journal of Engineering and Technology*, vol. 5, no. 6, pp. 4637–4642, 2014.
- [23] C. Zhou, H. B. Gao, L. Gao, and W. G. Zhang, "Particle swarm optimization (PSO) algorithm," *Application Research of Computers*, vol. 12, pp. 7–11, 2003.
- [24] T. Rumpf, A.-K. Mahlein, U. Steiner, E.-C. Oerke, H.-W. Dehne, and L. Plumer, "Early detection and classification of plant diseases with Support Vector Machines based on hyperspectral reflectance," *Computers and Electronics in Agriculture*, vol. 74,no. 1, pp. 91–99, 2010.
- [25] Z. H. Zhou and S. F. Chen, "Neural network ensemble," *Chinese Journal of Computers*, vol. 25, no. 1, pp. 1–8, 2002.