International Information and Engineering Technology Association
Advancing the World of Information and Engineering

Traitement du Signal
Vol. 39, No. 3, June, 2022, pp. 1013-1018
Journal homepage: http://iieta.org/journals/ts

An Enhanced Identification and Classification Algorithm for Plant Leaf Diseases Based on Deep Learning

Umamageswari Arasakumaran[1*], Shiny Duela Johnson[1], Dioline Sara[2], Raja Kothandaraman[1]

[1] Department of CSE, SRM Institute of Science and Technology, Ramapuram, Chennai 600089, India
[2] Geetam School of Technology, Bangalore, Karnataka 561205, India

Corresponding Author Email: umamagea@srmist.edu.in

https://doi.org/10.18280/ts.390328

Received: 11 February 2022
Accepted: 24 April 2022

Keywords:
image processing, progressive neural architecture search (PNAS), gray level co-occurrence matrix (GLCM), Mendeley dataset, fuzzy c-means clustering (FCM), leaf disease identification

ABSTRACT

Identification of plant disease sis a difficult task for farmers. If the diseases are misidentified, there will be a huge crop failure, which threatens the living of farmers. This paper proposes a new tool for farmers to identify plant leaf diseases automatically, and provide solutions to this problem on expert database. Firstly, the infected spots of the leaf are recognized through fuzzy c-means clustering (FCM). Then, the features are extracted by gray-level co-occurrence matrix (COLCM), and classified by progressive neural architecture search (PNAS). The proposed tool was tested on Mendeley Dataset, which covers 2,278 images of healthy leaves, and 2,225 leaves with leaf blights, rust, mealy bugs, and powderily mildew, angular leaf spot, and downy mildew. The experimental results show that our approach outperformed the other methods in accuracy (up to 95%).

1. INTRODUCTION

Modern advances have enabled the human society to provide enough food to feed a large population. Mishro Pranaba K et.al and Xie Xiaoyue et.al proposed a solution to meet the food demand of a large population, based on deep learning algorithms [1, 2]. However, food security is continuously jeopardized by environmental variation [3], pollination decline, plant diseases [4], and other factors. Among them, plant diseases not only threaten universal food safety, but also bring severe consequences to small holder farmers, whose livelihoods are solely based on harvests. In the modern world, smallholder farmers produce close to 80% of all the food (UNEP, 2013). The production losses mainly occur due to insects and plant diseases [5]. A number of rules have been devised to prevent crop loss induced by diseases. For example, integrated pest management (IPM) systems have been improved over time, replacing conventional approaches of widespread chemical treatment [6, 7].

The initial step of plant infection management is to recognize a disease effectively. Generally, the proof for infection recognition is provided by agriculture associations or different foundations, namely, plant clinics [8]. More recently, web-based data are provided to facilitate disease determination, utilizing the expanding Internet access around the world [9, 10]. In addition, according to ITU (International Telecommunication Union) plant disease identification tools have multiplied, thanks to the quick innovation of cell phones.

The primary goal of this paper is to build up a successful technique for identifying the diseases of plant leaves, and their indices, thus developing an appropriate framework for early and savvy recognition of plant leave diseases [11]. In recent years, computer vision and deep learning have gained popularity in the research of fungal diseases, owing to their computing ability and accuracy [12]. Deep architectures learn features via multiple layers. Based on these architectures, it is possible to learn suitable predictive features from instances, except hand-engineering features [13, 14].

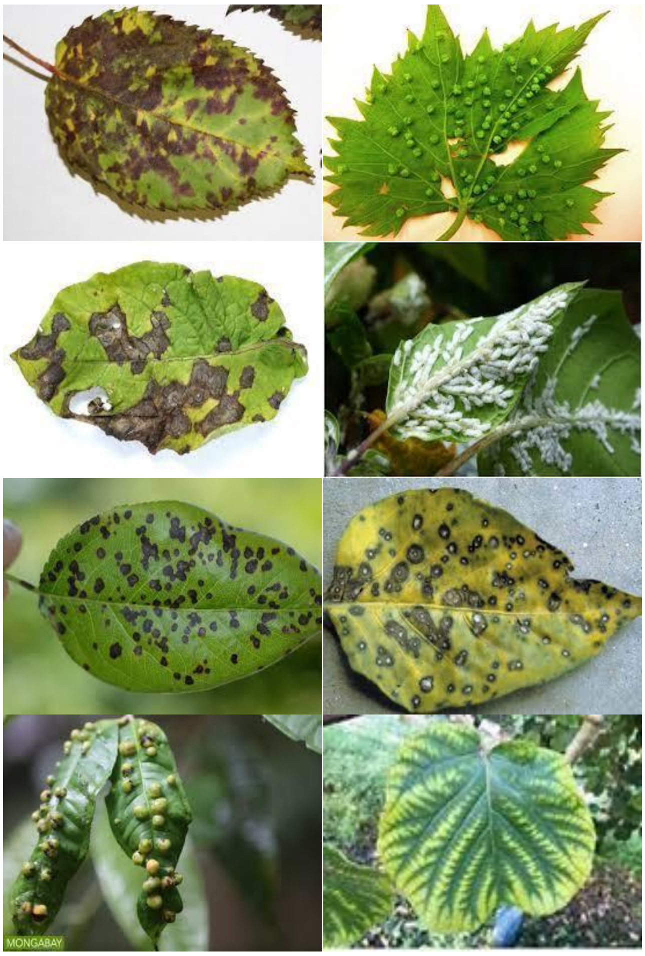


Figure 1. Leaves with various diseases

This paper demonstrates the practicality of a deep learning approach based on Mendeley -6718 MB leaf dataset on the following leaves: Arjun, Bael, Basil, Chinar, Guava, Jamun, Jatropha, Lemon, Mango, Pomegranate, and some grains and grasses. Some images are displayed in Figure 1. The common diseases of these leaves include anthracnose, scab, leaf blotch, shot hole, leaf blights, rust, mealy bugs, powderily mildew, angular leaf spot, and downy mildew [9, 15].

The remainder of this paper is organized as follows: Section 2 reviews the related literature; Section 3 introduces our model for leaf disease prediction; Section 4 presents and discusses the results; Section 5 draws the conclusions.

2. LITERATURE REVIEW

So far, many deep learning tools have been applied to identify various leaf diseases. For example, support vector machine (SVM) classifier, global pooling dilated convolution neural network (GPDCNN), generative adversarial networks (DCGAN), and improved convolutional neural network (ICNN) are introduced to identify rice leaf diseases (e.g., tungro, brown spot, blast, and bacterial blight), cucumber leaf diseases, tomato leaf diseases, and grape leaf diseases [1, 2].

2.1 Single shot multi-box detector (SSD)

The SSD is a one-step object discovery technique that visualizes the sorts of substances, and organizes the associated bounding boxes, without needing region proposals [16]. To process objects of varied sizes, the classic SSD associates multiple feature maps of various dimensions. The SSD boasts a faster discovery speed than faster recurrent CNN (Faster R-CNN) [3], although the two approaches have nearly identical discovery accuracies. Another advantage of the SSD lies in the fusion of multi-angle features. In this study, this technique is employed as the core procedure of object detection.

2.2 Inception module

The most candid way to increase the feature extraction ability of a deep neural network is to broaden or deepen that network. However, there are two drawbacks of this approach [17, 18]. On the one hand, the beginning segment employs similar layers with different kernels, and combines their outputs. The network thus enlarged may face the risk of over fitting. On the other hand, the computing overhead may surge. Here, the number of parameters is reduced by replacing one 5×5 convolutional layer with two 3×3 convolutional layers [3], which preserves the diversity of sensitive fields.

2.3 VGGNet

VGGNet [19] is a highly portable model widely adopted for migration learning. Compared to traditional CNNs, VGGNet boasts a high precision in diagnosing common leaf diseases of apples. As a result, this paper designates VGGNet as the simple pre-network model.

In a CNN, the initial layers usually extract color and corner. It is of minima value to extract these features, using origin. Hence, the Conv1 1 to Pool3 layers of the VGGNet were preserved, and the following Conv4 1 and Conv4 2 layers were superseded with at least two origin components, to enhance the ability to extract multi-scale targets. In addition, Conv4 to Pool5 layers were placed after the origin component, without any modification [20]. To overcome the constraint on input size, the fully-connecting layers were replaced with 1×1 convolutional layers for Conv6 to Conv8. The final layer was designed as a 5-way soft max layer.

2.4 Stochastic gradient descent (SGD)

During the learning process, the VGGNet relies on the SGD rule to solve the biases and weights, which minimizes the loss function [21, 22]. The SGD rule randomly selects a few small cluster heads. In this paper, the number of clusters and the learning rate are set to 32 and 0.001, respectively. The SGD algorithm can quickly converge to the optimal solution.

3. METHODOLOGY

This paper presents a novel approach for identification and classification of plant diseases. Figure 2 shows the architecture of our approach.

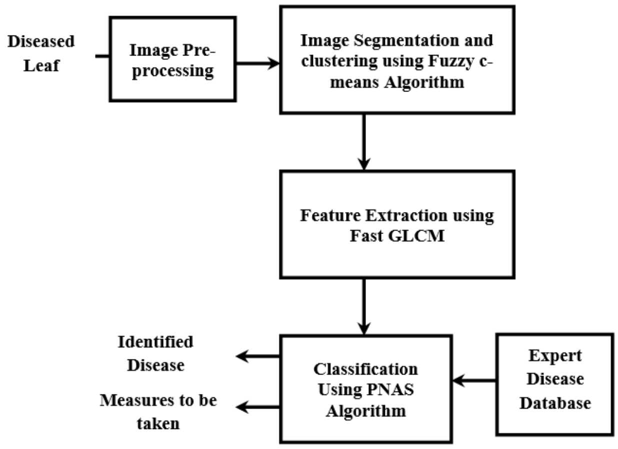

Diseased Leaf
Image Pre-processing
Image Segmentation and clustering using Fuzzy c-means Algorithm
Feature Extraction using Fast GLCM
Classification Using PNAS Algorithm
Expert Disease Database
Identified Disease
Measures to be taken

Figure 2. Architecture of our approach

```
Let X = {x1, x2, x3 ..., xn} be the set of data points and
V = {v1, v2, v3 ..., vc} be the set of centers.
1) Randomly select ‘c’ cluster centers.
2) Calculate the fuzzy membership 'µij'
3) Compute the fuzzy centers 'vj' using
4) Repeat step 2) and 3) until the minimum 'J' value is achieved or ||U(k+1) - U(k)|| <β.
where,
‘k’ is the iteration step.
‘β’ is the termination criterion between [0, 1].
‘U = (µij)n*c’ is the fuzzy membership matrix.
‘J’ is the objective function.
```

As shown in Figure 2, our approach contains preprocessing, spot or infection feature extraction, classification, and segmentation. The image features like color, color histogram, and texture are extracted and merged. The original image is segmented through fuzzy c-means clustering (FCE). To facilitate the extraction of salient visual features, this paper extracts and optimizes these features by fast gray level co-occurrence matrix (GLCM), and classifies them by progressive neural architecture search (PNAS), a deep learning classifier. The effectiveness of our approach was verified on Mendeley -6718 MB Leaf Dataset.

3.1 Image segmentation by FCM

The unsupervised FCM can solve various tasks, such as clustering, feature analysis, and classifier planning. During the FCM, each evident focus of a component is allocated into a cluster, and the original image is thus divided into dissimilar components. The clustering is done by repeatedly reducing the distance of each pixel to its cluster in the element space.

In an image, the pixels have strong correlations. For example, the pixels in the salient area share almost the same element information. Thus, the spatial relationship between adjacent pixels can guide image segmentation effectively.

Through the FCM, the pixels are allocated to different classes through fuzzy calculation. Let X=(x_1, x_2,…,x_N) be an image of N pixels; c be the number of clusters; x_i be the multispectral salient information. Then, the clustering is performed repeatedly at the lowest possible cost in equation [1]:

$$l = \sum_{i=1}^{n} \sum_{j=1}^{m} u_{ij} \, [\mathrm{x_j} - \mathrm{v_i}]^2 \qquad (1)$$

where, u_{ij} is the membership of pixel i to cluster j; v is the cluster size; m is the standard metric; m=2 is a constant of fuzziness.

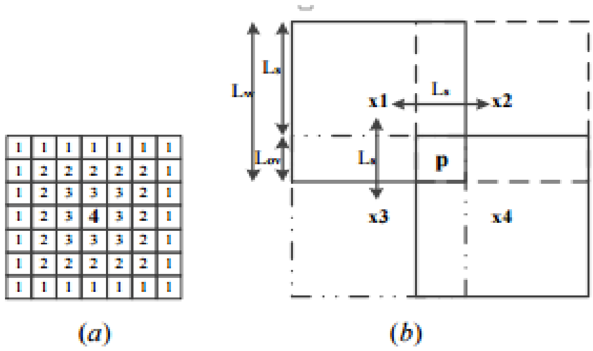
Every pixel close to the centroid of its cluster is assigned a high membership, while every pixel far from that centroid is assigned a low membership. The clustering is completed based on the membership.

3.2 Fast GLCM

The fast GLCM is suitable for eliminating unevenly correlated features like the traditional GLCM, but with a much shorter time (about 200 times faster). It also outshines the traditional GLCM in the precise classification of the pixels close to the class boundaries. To assess the effectiveness of fast GLCM, the general practice is to weaken the weight of GLCM media in each cycle, lower the sparsity of GLCM matrices, and re-quantize the gray levels of the input image to decrease G. Here, the G value is reduced from 256 levels to 32 levels. Thus, the created GLCM matrices are of the size 32×32. The defining features of GLCM matrices are correlation, energy, contrast, homogeneity, variance, and entropy. To prevent these features from being excessively large or small, a principal component transform (PCT) needs to be applied on these features.

To disclose the impact of the primary boundary of the fast GLCM, e.g., the effect of the step length (Ls) on the quality of removed features, the value range of Ls is set to [1, 20]. Note that Ls=1 represents the classic GLCM. Here, machine learning grouping accuracy is selected as the performance metric.

To prepare the machine learning classifier, 5% of pixels are chosen randomly for network training. The other 95% of pixels are utilized as test set. Figure 3 shows the general characterization accuracy of the machine learning classifier, and the overall extraction times of GLCM for salient regions. The results were measured at different step lengths Ls=1, …, 16. The given processing time is normalized to obtain the preparation time for the instance of Ls=1. It can be observed that fast GLCM could greatly shorten the preparation time, roughly by a factor of Ls=2, while ensuring the excellence of the structure.



1	1	1	1	1	1	1
1	2	2	2	2	2	1
1	2	3	3	3	2	1
1	2	3	4	3	2	1
1	2	3	3	3	2	1
1	2	2	2	2	2	1
1	1	1	1	1	1	1

(a) Linear weighting window (b) Weighting window with an overlap

Figure 3. General characterization accuracy of the machine learning classifier, and the overall extraction times of GLCM for salient regions

3.3 PNAS

```
Inputs: B (No. of maximum blocks), E (No.of
Maximum Epochs),
F (Filters in the first layer count),
K (Beam Size)N (no. of times to enroll cells),
trainingSet, valueSet
$SS_1$=B1 || candidate structure set with one block
$MM_1$ = cell_to_ConNeuNet($S_1$, N, F)
$CC_1$ = train_ConNeuNet($C_1$, valueSet)
$AA_1$ = eval_ConNeuNet($C_1$, valueSet)
$\pi$= fit ($S_1$, $A_1$)
for b=2: B do
        $SS_b^1$ = expanded-cell ($S_b$-1)
        $AA_b^1$ = prediction-cell ($S_b^1$- $\Pi$)
        $SS_b$ = topper-K($S_b^1$, $A_b^1$, K)
        $MM_b$ = cell_to_ConNeuNet($S_b$, N, F)
        $CC_b$ = train_ConNeuNet($M_b$, E, trainingSet)
        $AA_b$ = eval_ConNeuNet($C_b$, valueSet)
        $\pi$= update_predictor($S_b$, $A_b$, $\pi$)
end for
Return topper-K($S_B$, $A_B$, 1)
```

Several previous methods delve straight into the filled cells, or worse, the whole CNN. For example, NAS uses 50-step RNN6 as the checker for cell design. The CNN is designed through classic mutations of fixed-length binary strings. However, this straightforward technique is hard to traverse straight in a big search engine, particularly when there is little understanding of the composition of the actual model. In addition, this paper examines the space in reverse order, opening with the simplest copies. We began by building all feasible cell configurations from B1 (which has only one block), and arrange them in a queue. Then, all the copies in the queue are evaluated and trained in parallel. After that, each block is enlarged by one, among all the virtual blocks in B2. Since it is impossible to train and compute all the child networks, we developed an interpreting module based on the visited cells.

Then, all the cells are processed on Analyst, and the top-K elite individuals are retained in the queue. The above steps are repeated until all cells with sufficient blocks (B) are identified.

The optimal cell structure is determined using PNAS. A predetermined number of basic cell copies are stacked on top of each other. The stride 1 and stride 2 are changed according to the number of epochs. The classifier is based on global average pooling. The stacking model is trained on the Mendeley dataset. In terms of CIFAR-10, the images are of the size 32×32.In terms of ImageNet, there are two image sizes: the smaller size of 224×224, and the larger size of 331×331. Only one image is of the larger size. To lower the computing cost, opening part of the network has a convolutional kernel of 3×3, using stride 2. Only one kind of cells is adopted to narrow the cell search space.

4. EXPERIMENTAL RESULTS

Our approach was simulated on Matlab, using Python for coding. The neural network was programmed with Keras, and tested on 4,503 images in Mendeley database. The training set and test set were developed at a ratio of 3: 2. The accuracy of our approach was calculated to see if the performance on unknown images is consistent with that on the training images. Table 1 compares the accuracy of our approach and several deep learning CNNs: AlexNet [15], GoogLeNet [3], InceptionV3 [3], and VGGNet-16 [19]. Figures 4 to 6 display the outputs of our approach.

During training, the SGD algorithm was implemented to configure the weights and biases of each network, aiming to reduce the loss function. A small training set and sample size were selected randomly for the SGD. The sample size was set to 32, and the learning rate to 0.001. The small sample size ensures the precision of search. The momentum was set to 0.9, which determines the speed of the SGD to converge to the optimal solution. Figure 7 shows the relationship between accuracy and the number of training epochs. Figures 8 and 9 show the specificity and sensitivity of the various pre-networks in Table 1. It can be observed that PNAS achieved the highest accuracy, specificity, sensitivities, and convergence speed.

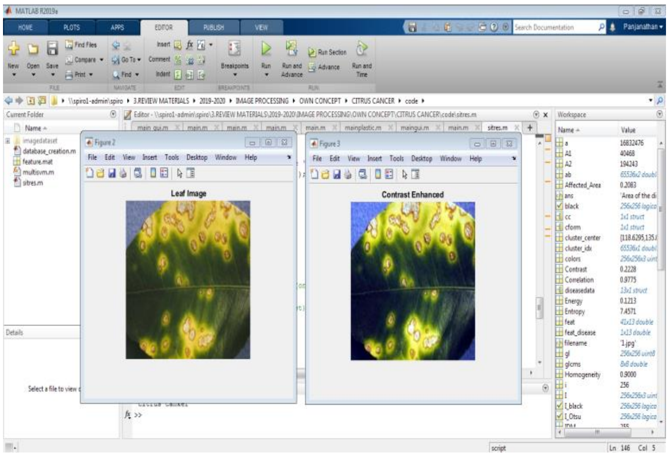


Figure 4. Input and contrast enhanced image

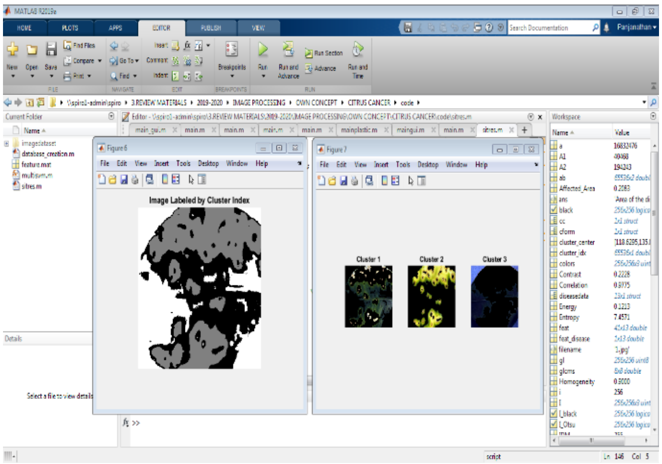


Figure 5. Clustered image

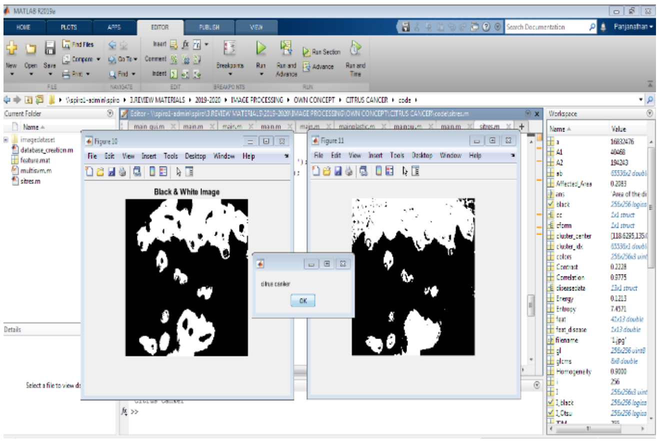


Figure 6. Output image

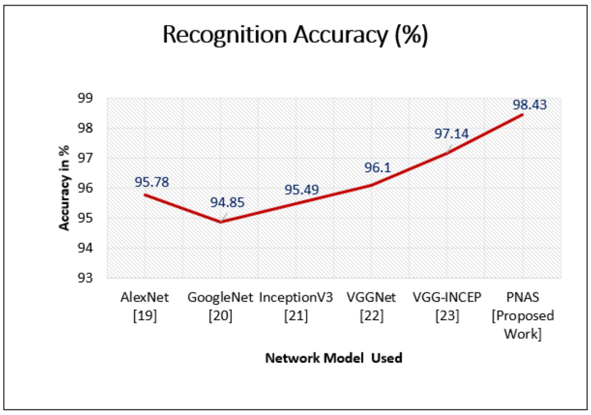


Figure 7. Graphical representation of accuracy

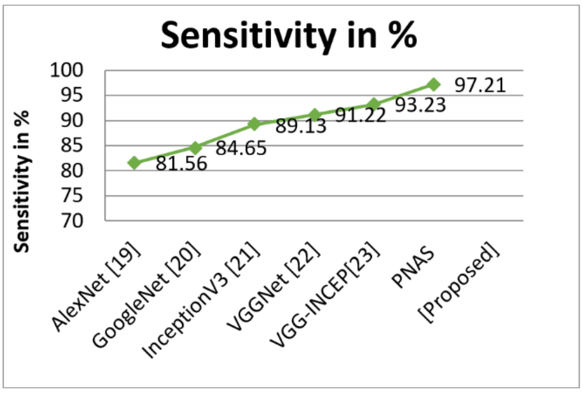


Figure 8. Graphical representation of sensitivity

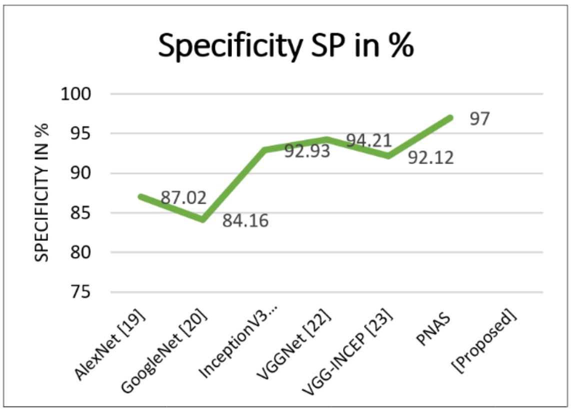


Figure 9. Graphical representation of specificity

Table 1. Accuracy of different approaches

S.No	Network Model	I/P Size	Acc in %	SN in %	SP in %
1	AlexNet [21]	227	95.78	81.56	87.02
2	GoogleNet [22]	224	94.85	84.65	84.16
3	InceptionV3 [23]	223	95.49	89.13	92.93
4	VGGNet [24]	224	96.10	91.22	94.21
5	VGG-INCEP [23]	224	97.14	93.23	92.12
6	PNAS [Proposed]	224	98.43	97.21	97.00

5. CONCLUSIONS

This paper presents a real-time detector of leaf diseases based on enhanced PNAS. This deep learning algorithm extracts the discriminative components of diseased leaf images automatically, and detects the general categories of leaf diseases with a high accuracy in real time. As it is understood throughout this paper, a methodology that instinctively grades the diseases on plant leaves is essential in the present scenario. Experimental results show that our PNAS model can detect general categories of leaf diseases in the Medeley dataset with accuracy up to 98.43% in real time. This offers potential remedies for the problems in plant leaf disease detection.

REFERENCES

[1] Mishro, P.K., Agrawal, S., Panda, R., Abraham, A. (2020). A novel type-2 fuzzy C-means clustering for brain MR image segmentation. IEEE Transactions on Cybernetics, 51(8): 3901-3912. https://doi.org/10.1109/TCYB.2020.2994235

[2] Xie, X., Ma, Y., Liu, B., He, J., Li, S., Wang, H. (2020). A deep-learning-based real-time detector for grape leaf diseases using improved convolutional neural networks. Frontiers in Plant Science, 11: 751. https://doi.org/10.3389/fpls.2020.00751

[3] Umamageswari, A., Bharathiraja, N., Irene, D.S. (2021). A novel fuzzy C-means based chameleon swarm algorithm for segmentation and progressive neural architecture search for plant disease classification. ICT Express. https://doi.org/10.1016/j.icte.2021.08.019

[4] Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., Wojna, Z. (2016). Rethinking the inception architecture for computer vision. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2818-2826. https://doi.org/10.1109/CVPR.2016.308

[5] Ehler, L.E. (2006). Integrated pest management (IPM): definition, historical development and implementation, and the other IPM. Pest Management Science, 62(9): 787-789. https://doi.org/10.1002/ps.1247

[6] Ariyapadath, S. (2021). Plant leaf classification and comparative analysis of combined feature set using machine learning techniques. Traitement du Signal, 38(6): 1587-1598. https://doi.org/10.18280/ts.380603

[7] Simonyan, K., Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556. https://doi.org/10.48550/arXiv.1409.1556

[8] Garcia-Ruiz, F., Sankaran, S., Maja, J.M., Lee, W.S., Rasmussen, J., Ehsani, R. (2013). Comparison of two aerial imaging platforms for identification of Huanglongbing-infected citrus trees. Computers and Electronics in Agriculture, 91: 106-115. https://doi.org/10.1016/j.compag.2012.12.002

[9] Huang, K.Y. (2007). Application of artificial neural network for detecting Phalaenopsis seedling diseases using color and texture features. Computers and Electronics in Agriculture, 57(1): 3-11. https://doi.org/10.1016/j.compag.2007.01.015

[10] Hughes, D., Salathé, M. (2015). An open access repository of images on plant health to enable the development of mobile disease diagnostics. arXiv preprint arXiv:1511.08060. https://doi.org/10.48550/arXiv.1511.08060

[11] Strange, R.N., Scott, P.R. (2005). Plant disease: A threat to global food security. Annual Review of Phytopathology, 43(1): 83-116.

[12] LeCun, Y., Bengio, Y., Hinton, G. (2015). Deep learning. Nature, 521(7553): 436-444. https://doi.org/10.1038/nature14539

[13] Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A.C., Li, F.F. (2015). ImageNet large scale visual recognition challenge. International Journal of Computer Vision, 115(3): 211-252. https://doi.org/10.1007/s11263-015-0816-y

[14] Krizhevsky, A., Sutskever, I., Hinton, G.E. (2012). ImageNet classification with deep convolutional neural networks. Proceedings of the 25th International Conference on Neural Information Processing Systems, pp. 1097-1105.

[15] Sanchez, P.A., Swaminathan, M.S. (2005). Cutting world hunger in half. Science, 307(5708): 357-359. https://doi.org/10.1126/science.110905

[16] Mokhtar, U., Ali, M.A., Hassanien, A.E., Hefny, H. (2015). Identifying two of tomatoes leaf viruses using support vector machine. In: Mandal, J., Satapathy, S., Kumar Sanyal, M., Sarkar, P., Mukhopadhyay, A. (eds) Information Systems Design and Intelligent Applications. Advances in Intelligent Systems and Computing, vol 339. Springer, New Delhi. https://doi.org/10.1007/978-81-322-2250-7_77

[17] Jiang, P., Chen, Y., Liu, B., He, D., Liang, C. (2019). Real-time detection of apple leaf diseases using deep learning approach based on improved convolutional neural networks. IEEE Access, 7: 59069-59080. https://doi.org/10.1109/ACCESS.2019.2914929

[18] Singh, A., Ganapathysubramanian, B., Singh, A.K., Sarkar, S. (2016). Machine learning for high-throughput stress phenotyping in plants. Trends in Plant Science,

21(2): 110-124. https://doi.org/10.1016/j.tplants.2015.10.015

[19] Poplin, R., Varadarajan, A.V., Blumer, K., Liu, Y., Mcconnell, M.V., Corrado, G.S., Webster, D.R. (2017). Predicting cardiovascular risk factors from retinal fundus photographs using deep learning. arXiv 2017. arXiv preprint arXiv:1708.09843.

[20] Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Rabinovich, A. (2015). Going deeper with convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1-9. https://doi.org/10.1109/CVPR.2015.7298594

[21] Tai, A.P., Martin, M.V., Heald, C.L. (2014). Threat to future global food security from climate change and ozone air pollution. Nature Climate Change, 4(9): 817-821. https://doi.org/10.1038/nclimate2317

[22] Wetterich, C.B., Kumar, R., Sankaran, S., Junior, J.B., Ehsani, R., Marcassa, L.G. (2013). A comparative study on application of computer vision and fluorescence imaging spectroscopy for detection of citrus Huanglongbing disease in USA and Brazil. In Laser Science, JW3A-26. https://doi.org/10.1364/FIO.2013.JW3A.26

[23] Pereira, F., Burges, C.J.C., Bottou, L., Weinberger, K.Q. (Hrsg) (2012). Advances in Neural Information Processing Systems. Curran Associates, Inc; (25): 1097-1105.

[24] Harvey, C.A., Rakotobe, Z.L., Rao, N.S., Dave, R., Razafimahatratra, H., Rabarijohn, R.H., MacKinnon, J.L. (2014). Extreme vulnerability of smallholder farmers to agricultural risks and climate change in Madagascar. Philosophical Transactions of the Royal Society B: Biological Sciences, 369(1639): 20130089. https://doi.org/10.1098/rstb.2013.0089