

Groundnuts Leaf Disease Recognition using Neural Network with Progressive Resizing

Rajnish M. Rakholia¹, Jinal H. Tailor², Jatinderkumar R. Saini^{3*}, Jasleen Kaur⁴, Hardik Pahuja⁵

S. S. Agrawal Institute of Management & Technology, Navsari, India¹

S. S. Agrawal Institute of Management & Technology, Navsari, India²

Symbiosis Institute of Computer Studies and Research³

Symbiosis International (Deemed University), Pune, India³

School of Engineering, P P Savani University, Surat, India⁴

Yardi Systems Inc., Pune, India⁵

Abstract—Groundnut is an important oilseed crop in the world, and India is the second-largest producer of groundnuts. This crop is prone to attack by numerous diseases which is one of the most important factors contributing to the loss of productivity and degradation in the quality; both of these finally result in a low agricultural economy. Therefore, it is necessary to find better and more reliable automation solutions to recognize groundnut leaf diseases. In this paper, a deep learning based model with progressive resizing is proposed for groundnut leaf disease recognition and classification tasks. Five major categories of groundnut leaf diseases namely leaf spot, armyworms effect, wilts, yellow leaf, and healthy leaf are considered. The proposed model was trained with and without progressive resizing while it was validated using cross-entropy loss. The first of its kind dataset used for training and validation purposes was manually created from the Saurashtra region of Gujarat state of India. The created dataset was imbalanced in terms of a different number of samples for each category. To handle the imbalanced dataset problem, the extended focal loss function was used. To evaluate the performance of the proposed model, different performance measures including precision, sensitivity, F1-score, and accuracy were applied. The proposed model achieved state-of-the-art accuracy of 96.12%. The model with progressive resizing performed better than the traditional core neural network-based model built on cross-entropy loss.

Keywords—Groundnut leaf disease recognition; progressive resizing; deep learning; neural network

I. INTRODUCTION

The groundnut crop plays an important role in the agricultural export commodity and edible oilseed economy of India. In the year 2019 alone, total groundnut acreage and production in India were 3.931 million hectares and 6.862 million MT respectively (IOPEPC, 2019)[1], but still, the average yield is low. Disease attack is a major factor contributing to the loss of productivity, quality, and early death of the leaves (Konate et al., 2020)[2]. Therefore, it is necessary to take steps toward developing a fast and accurate groundnut leaf disease recognition methodology to increase productivity sustainably. This will be of great significance to the various stakeholders. Till now almost no commercial tools are available for accurate recognition of groundnut leaf disease and very less quality research articles are published for the same. One of the key reasons behind this might be a lack of

benchmark datasets available for groundnut leaf disease recognition research and experiments.

In this research, the groundnut leaf dataset was created manually from the Saurashtra regions of Gujarat. Initially, all images were captured in fixed background squared format with the size of 3000x3000 pixel (3 color channels), and then final datasets were prepared in different sizes of images including 32x32, 64x64, 128x128, and 256x256. Based on the thorough review of the literature and a comprehensive review carried out by (Ngugi et al., 2020; Chouhan et al., 2020; Kaur et al., 2019) [3-5] on methods used in leaf disease recognition using image processing, machine learning, and deep learning techniques, it was derived for the current research work that very few researchers have worked on groundnut leaf disease recognition. Chen et al. (2019) [6] used spectral index and disease index based on their correlation in leaf spectrum range between 325nm to 1075nm. Their results showed that near-infrared regions' canopy spectral reflectance decreased as the disease index increased. In the regression model, normalized difference spectral indexes were R938, R761 with the value of R2 up to 0.68 for peanut leaf spot disease detection. Based on the index model high fit between estimated and observed values, they concluded that the model could be used for peanut leaf spot disease detection. For groundnut disease classification (Chaudhary et al., 2016)[7] proposed an improved Random Forest Classifier using instance Filter-Resample and attribute evaluator methods for balancing the class distributions of the multi-class dataset. The proposed method was also applied on five different datasets such as Diabetes, Soybean, Audiology, Vote, and Breast Cancer, and obtained the value of the F1-measure in the range between 0.89 and 0.97. They advocated that the result of their method was effective when the dataset is unbalanced in terms of the number of samples varying among different classes. Ramakrishnan and Sahaya (2015) [8] applied a backpropagation algorithm for groundnut leaf disease detection and classification. Initially, RGB was converted into HSV and then plane separation and color features extraction steps were carried out. Dong et al. (2019)[9] applied a capsule network for peanut leaf disease recognition with the use of dynamic routing to overcome the problem of rotational invariance and spatial relationships. Their empirical observations showed that the recognition accuracy of the capsule network is 82.17% which is better than the

*Corresponding Author.

corresponding value of 81.14% of the convolutional neural network. It is notable to mention here that the capsule network was originally proposed by Sabour et al. (2017) [10]. Vaishnave et al. (2020)[11] used a convolutional network for groundnut disease classification and claimed higher accuracy for training and testing on the PlantVillage dataset, but the fact is, till now no benchmark image dataset for groundnut diseases is released or published by PlantVillage (PlantVillage, 2020) [12].

Since the introduction of Deep learning, many state-of-the-art benchmark architectures such as DenseNet (Huang et al., 2017) [13], Deep residual learning (He et al., 2016) [14], Inception-v4 (Szegedy et al., 2017) [15], GoogLeNet (Szegedy et al., 2015) [16], VGG (Simonyan and Zisserman, 2014) [17] and AlexNet (Krizhevsky et al., 2012) [18] have been found to give an incredible performance for object detection and various computer vision tasks. Many researchers have also used these architectures for transfer learning for plant disease recognition e.g. (Fuentes et al., 2018) [19] have used R-CNN, AlexNet, GoogLeNet for the identification of tomato leaf disease, (Liu et al., 2018)[20] have used AlexNet for apple leaf disease detection, (Kaya et al., 2019; Barbedo 2019; Brahimi et al., 2018; Mohanty et al., 2016)[21-24] have used multiple benchmark architectures for multiple plant disease recognition.

In recent times, convolutional neural network-based methods are great in demand for plant leaf disease recognition due to automatic deep feature extraction. For corn leaf disease recognition and classification (Waheed et al., 2020) [25] proposed an optimized dense convolutional neural network model. Ji et al. (2020) [26] proposed a Convolutional Neural Network-based architecture for multi-label learning for crop leaf diseases recognition and severity estimation. To overcome the problem of the unbalanced dataset (Zhong and Zhao, 2020) [27] have proposed DenseNet-121 as the backbone network and used three methods regression, multi-label classification, and focus on loss function to identify apple life disease and obtained test accuracy of 93.51%, 93.31% and 93.71% respectively which was better than the accuracy of 92.29% obtained by traditional multi-classification method with a cross-entropy loss function. Sethy et al. (2020) [28] used different CNN architectures based on deep features using a support vector machine to identify rice leaf disease.

Self-attention CNN-based architecture was proposed by (Zeng and Li, 2020) [29] for crop leaf disease recognition. Zhang et al. (2019) [30] have used global pooling dilated convolutional neural network for cucumber leaf disease identification. Karlekar and Seal (2020) [31] proposed CNN-based SoyaNet for soybean leaf disease classification. Their proposed network obtained higher precision, recall, and f1-score value compared to the other nine state-of-the-art models.

Almost all the existing CNN-based architectures designed for leaf disease recognition and classification perform well at some level in terms of precision, recall, f1-score, and accuracy. However, the time complexity and model generalization are major problems when CNN-based architecture is trained and

optimized on the high volume of small and large images or in a real-time environment due to the range of features and number of layers in the architecture. In this paper, a CNN-based architecture with progressive resizing for model generalization, optimization, and performance improvement is proposed.

II. MATERIALS AND METHODS

A. Dataset

Based on the best knowledge of related literature, almost no benchmark dataset of groundnut leaf disease is publicly available for research. The dataset for the current research was created manually and comprised of five major groundnut leaf classes, viz. leaf spot, armyworm effects, wilts, yellow leaf, and healthy leaf. All major types of symptomatic leaves were plucked manually from the plants and put onto the fixed background to capture the images. Initially, all the images are captured in squared format with the size of 3000x3000 (3 color channels), and then later all the captured images are resized in different sizes of 32x32, 64x64, 128x128, and 256x256 for model development using progressive resizing. The created dataset was divided into a ratio of 80:20 for training and testing. All the classes and corresponding labels considered for this research are shown in Table I.

The distribution of training and testing datasets is depicted in Fig. 1 which also indicates that the dataset is imbalanced as the distribution is not in equal proportion for each class. This was solved using the Focal loss function and is discussed in section 2.4 in detail. According to the data presented in Fig. 1, the most commonly occurring leaf disease in the groundnut crop is Leaf Spot which contributes the largest proportion, except healthy leaf, in the dataset. Similarly, the least proportion in the dataset is comprised of groundnut wilts. Notably, it is also a more harmful disease.

B. Proposed Network Architecture

The well-known terminologies used in CNN such as a convolutional layer, pooling layer, filters, fully connected layer, etc., are not addressed here to rule out redundancy.

Model built on the standard convolutional neural network works well when input images are fixed in size. The model gives good accuracy when feeding larger images but it takes a long time and uses more computation power during the training phase. Scale-up and scale-down are required during training when input images are very small and large, respectively.

TABLE I. GROUNDNUT LEAF DISEASE CLASSES AND CORRESPONDING LABEL

Class Name	Class Label
Groundnut Yellow Leaf	0
Groundnut Wilts	1
Groundnut Leaf Spot	2
Groundnut Healthy Leaf	3
Groundnuts ArmywormsEffect	4

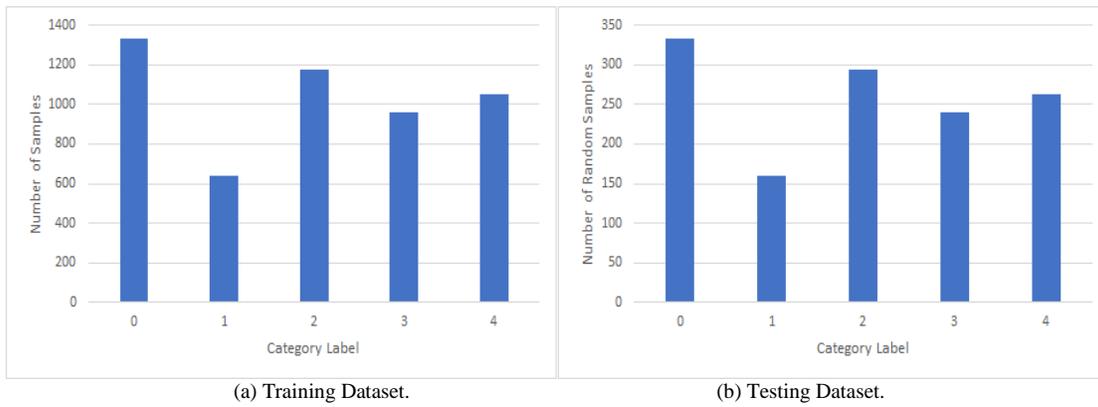


Fig. 1. Data Distribution for Training and Testing.

In this research, we have considered progressive-resizing methodology (originally proposed by Jeremy Howard, 2018) [32] for training a model to improve the recognition rate and for the model generalization. The workflow of progressive resizing for the proposed work is as follows:

Phase-1: The first model was trained on small images with the size of 32x32 with 3 color channels.

Phase-2: The next model was trained on upscaled images with the size of 64x64. Here, the layers and weights used in the previous smaller scale model were incorporated during training.

Phase-3: The third model was trained on 128x128 images; the output of phase-2 was fed as the input of phase-3. Each model is responsible to find some new features and patterns which were hidden in a previous smaller-scale model.

Phase-4: The final architecture was built and trained using the size of 256x256 images. Each larger scale model incorporates the previous smaller scale model in its architecture.

The proposed model was started to be built on 32x32 sizes and then scaled up to 4X, where X was the initial size of input images. Each phase of the proposed network was trained on the specific size of images and extracted some features. The trained model was saved with their weights and the weight was not changed in further training. Each subsequent phase was

responsible to extract additional findable features which were hidden and not found in the previous phase of the network. Models built on small-size images generalize well to larger input sizes and they take less time in processing (Howard, 2018) [33]. The proposed combined architecture is depicted in Fig. 2.

In order to introduce nonlinearity into the model, Rectified Linear Unit (ReLU) was used in each convolution operation. The ReLU function, $F(x) = \max(0, x)$, returns x for all values of $x > 0$, and returns 0 for all values of $x \leq 0$.

C. SoftmaxLoss

Here, the Softmax loss is categorical cross-entropy loss which is computed based on class probability generated by Softmax activation and using cross-entropy loss function. It has been referred to as $f(s_i)$ and defined as follows:

$$f(s)_i = \frac{e^{s_i}}{\sum_j^C e^{s_j}}$$

Where s_i , is a network score of each class i in C .

In this research specifically disease classification, labels are considered as one-hot, which means only positive class C_p were considered in cross-entropy loss which can be defined as follows:

$$\text{Cross-Entropy} = - \sum_i^C t_i \log(f(s)_i)$$

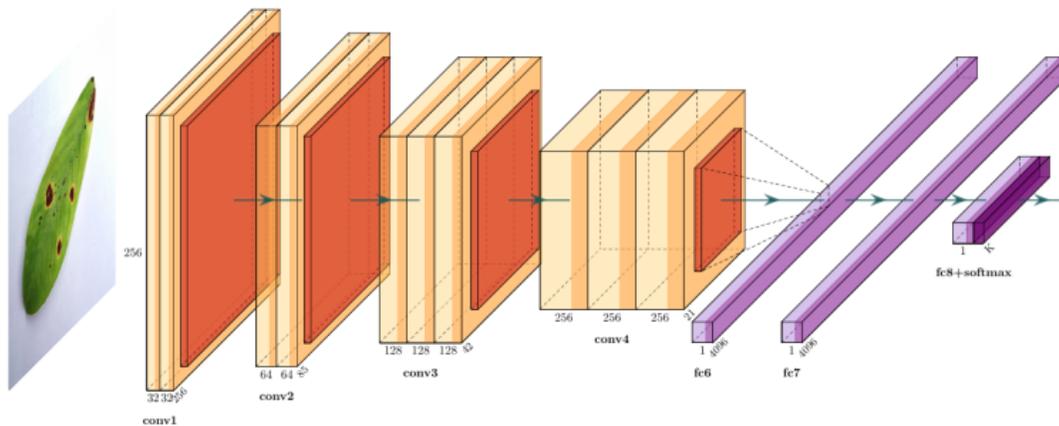


Fig. 2. Proposed CNN based Architecture [Image by Author].

Target vector t only contained non-zero elements. We can write the following equation for cross-entropy loss after discarding the elements of summation which are zero because of target labels, so $t_i=t_p$.

$$\text{Cross-entropy} = -\log\left(\frac{e^{s_p}}{\sum_j^c e^{s_j}}\right)$$

Where s_p is a score of positive class in CNN.

D. Focal Loss

The proposed classification also involved the problem of the imbalanced dataset. Some classes have a very small number of samples whereas some other classes contained almost double the number of samples. The network trained using an imbalanced dataset makes the network biased towards learning in favor of the classes having a higher number of samples while the remaining classes go under-looked. To tackle the class imbalance problem, a focal loss function with the multiplication of cross-entropy loss function with modulating factors was used. The Focal Local loss function is an improved version of cross-entropy loss which is made by adding focusing and balancing parameters in the cross-entropy loss function. This enhances the efficiency of the network and improves the results of misclassified observations. The focal loss function was originally proposed by Facebook AI Research (Lin et. al., 2017) [33] for binary classification in the object detection task. We have extended the concept of focal loss for multi-class classification. The general form of the focal loss function is:

$$\text{Focal Loss} = -(1 - p_t)^\gamma \log(p_t)$$

The focal loss for multi-class classification can be derived as follows:

$$\text{Focal Loss} = -(1 - f(s)_i)^\gamma \log(f(s)_i)$$

Where gamma (γ) is focusing parameter and $f(s)_i$ is softmax used for multi-class classification in cross-entropy. If $\gamma = 0$, then this equation is equivalent the equation of cross-entropy loss. Tunable focusing parameter γ should be ≥ 0 . Gamma (γ) controls the shape of the curve. The higher value of gamma (γ) reduces the loss of well-classified observations at some level. In this equation $(1 - f(s)_i)^\gamma$ is considered as modulating factor.

Finally, focal loss function can be defined with balancing parameter for imbalanced data as follows:

$$\text{Focal Loss} = -\alpha(1 - f(s)_i)^\gamma \log(f(s)_i)$$

Where, alpha (α) is a balancing parameter. In this case, alpha refers to the weights used in the network, and small weights were assigned to dominating class while higher weights were assigned to the rare class.

III. RESULT AND DISCUSSION

The training parameters used for the proposed model are shown in Table II. The model was trained on different sizes of images starting from 32x32x3 and the final model was trained on 256x256x3 images. After doing many experiments, the final batch size and learning rate were set to 32 and 1e-3 respectively. Optimizer Adam was used with the decay of 1e-5.

TABLE II. TRAINING PARAMETERS

Parameter	Settings
Final Input size	(256,256,3)
Batch size	32
Learning Rate	1e-3
Optimizer	Adam with decay of 1e-5

The distribution of testing data for each category is shown in Table III.

TABLE III. DISTRIBUTION OF TEST DATASET

Class	Number of Samples
Groundnuts Healthy Leaf	333
Groundnuts Armyworms Effect	263
Groundnuts Wilts	160
Groundnuts Leaf Spot	293
Groundnut Yellow Leaf	240

The core CNN-based model without progressive resizing was evaluated using cross-entropy loss with different statistical measures, as shown in Table IV. The maximum and minimum F1 scores obtained were 0.934773 and 0.853771 for leaf spot and Wilts categories respectively. The average accuracy was reported to be 0.918823. The evaluation of the proposed model with progressive resizing and cross-entropy loss is shown in Table V. The result shows that the accuracy of 0.949381 obtained in progressive resizing is better than the accuracy of 0.918823 obtained with the core CNN model.

TABLE IV. CROSS ENTROPY LOSS WITHOUT PROGRESSIVE RESIZING

	Precision (%)	Sensitivity (%)	F1 score (%)
Groundnuts Healthy Leaf	0.929509	0.919712	0.924585
Groundnuts Armyworms Effect	0.929019	0.937031	0.933008
Groundnuts Wilts	0.857722	0.849856	0.853771
Groundnuts Leaf Spot	0.938732	0.930847	0.934773
Groundnut Yellow Leaf	0.919686	0.928934	0.924287
Weighted avg	0.920766	0.918823	0.919774

TABLE V. CROSS ENTROPY LOSS WITH PROGRESSIVE RESIZING

	Precision (%)	Sensitivity (%)	F1 score (%)
Groundnuts Healthy Leaf	0.963758	0.958712	0.961228
Groundnuts Armyworms Effect	0.961034	0.960001	0.960517
Groundnuts Wilts	0.913827	0.901834	0.907791
Groundnuts Leaf Spot	0.962093	0.959823	0.960957
Groundnut Yellow Leaf	0.942003	0.943748	0.942875
Weighted avg	0.952575	0.949381	0.950971

The results of the proposed model without progressive resizing using focal loss ($\gamma=2$) is shown in Table VI. The average accuracy of 0.930404 was reported which is better

than the accuracy of 0.918823 achieved in cross-entropy loss. The training-validation accuracy and training validation loss using the focal loss function are depicted in Fig. 3(a) and Fig. 3(b) respectively. Here, focal loss down-weighted the easy observations and focused training on hard observations of imbalanced classes.

The evaluation results of the proposed CNN-based model with progressive resizing using focal loss function is shown in Table VII. The obtained average accuracy was 0.961229 which is better than the accuracy obtained in all other cases. Basically, setting the value of $\gamma > 0$ reduces the relative loss for well-classified observations, for the proposed model we obtained a higher accuracy when $\gamma=2$ set. The minimum F1 score was reported for the Groundnuts Wilts class and the maximum F1 score was reported for the Groundnuts Armyworms Effect class. The training and validation accuracy and loss for CNN-based model with progressive resizing using focal loss function is depicted in Fig. 4(a) and 4(b), respectively.

TABLE VI. FOCAL LOSS ($\Gamma=2$) WITHOUT PROGRESSIVE RESIZING

Categories	Precision (%)	Sensitivity (%)	F1 score (%)
Groundnuts Healthy Leaf	0.937483	0.929594	0.933522
Groundnuts Armyworms Effect	0.932743	0.942032	0.937364
Groundnuts Wilts	0.896029	0.88	0.887942
Groundnuts Leaf Spot	0.9488	0.948921	0.948860
Groundnut Yellow Leaf	0.926023	0.929782	0.927899
Weighted avg	0.931809	0.930404	0.931088

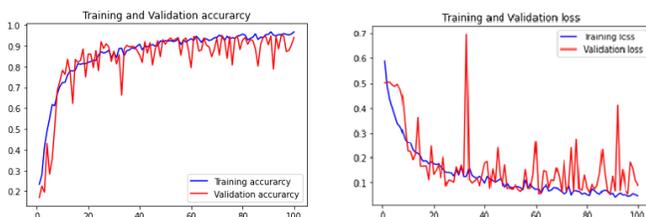


Fig. 3. Training and Validation of Core CNN Model with Focal Loss.

TABLE VII. FOCAL LOSS ($\Gamma=2$) WITH PROGRESSIVE RESIZING

Categories	Precision (%)	Sensitivity (%)	F1 score (%)
Groundnuts Healthy Leaf	0.978302	0.9702	0.974234
Groundnuts Armyworms Effect	0.972003	0.98	0.975985
Groundnuts Wilts	0.92904	0.92893	0.928985
Groundnuts Leaf Spot	0.97	0.9610212	0.965490
Groundnut Yellow Leaf	0.958003	0.95	0.953985
Weighted avg	0.965235	0.961229	0.963217

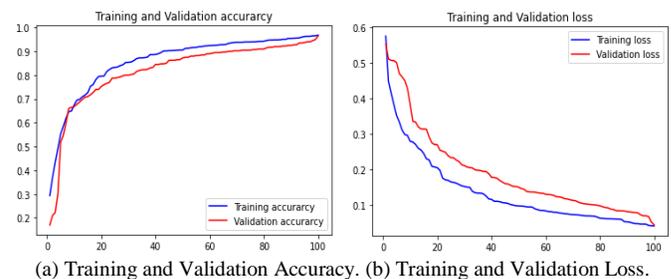


Fig. 4. Training and Validation with Progressive Resizing and Focal Loss.

IV. CONCLUSION

CNN-based architecture with progressive resizing was proposed to classify groundnut leaves into classes, namely, healthy leaf, armyworm effect, groundnut wilts, yellow leaf, and leaf spot which happen to be the most occurring disease in groundnut leaves. The proposed architecture with and without progressive resizing was evaluated using cross-entropy loss and focal loss functions. The obtained results of the proposed model without progressive resizing were 91.88% and 93.04% using cross-entropy loss and focal loss respectively while the average accuracy of the proposed model with progressive resizing using focal loss was 96.12%. Based on the empirical results, it is concluded that the progressive resizing-based model is a more generalized model and it was trained on different scales starting from the small-scale images of 32x32 size while the final model was built using 256x256 size images. The results obtained on different scenarios show that the CNN-based model with progressive resizing outperforms the core CNN-based architecture while the focal loss function helped out to deal with the imbalanced dataset problem. In the future, we plan to implement the proposed concept to recognition of diseases in real-time farm fields by combining computer vision technologies.

REFERENCES

- [1] Indian Oilseed and Produce Export Promotion Council (IOPEPC, 2019), Kharif-2019 Groundnut Crop Survey, retrieved on Feb 4, 2021, from http://www.iopec.org/misc/2019_20/Kharif%202019%20Groundnut%20Crop%20Survey.pdf.
- [2] Konate, M., Sanou, J., Miningou, A., Okello, D. K., Desmae, H., Janila, P., & Mumm, R. H. (2020). Past, present and future perspectives on groundnut breeding in Burkina Faso. *Agronomy*, 10(5), 704.
- [3] Ngugi, L. C., Abelwahab, M., & Abo-Zahhad, M. (2020). Recent advances in image processing techniques for automated leaf pest and disease recognition-a review. *Information Processing in Agriculture*.
- [4] Chouhan, S. S., Singh, U. P., & Jain, S. (2020). Applications of computer vision in plant pathology: a survey. *Archives of computational methods in engineering*, 27(2), 611-632.
- [5] Kaur, S., Pandey, S., & Goel, S. (2019). Plants disease identification and classification through leaf images: A survey. *Archives of Computational Methods in Engineering*, 26(2), 507-530.
- [6] Chen, T., Zhang, J., Chen, Y., Wan, S., & Zhang, L. (2019). Detection of peanut leaf spots disease using canopy hyperspectral reflectance. *Computers and electronics in agriculture*, 156, 677-683.
- [7] Chaudhary, A., Kolhe, S., & Kamal, R. (2016). An improved random forest classifier for multi-class classification. *Information Processing in Agriculture*, 3(4), 215-222.

- [8] Ramakrishnan, M. (2015, April). Groundnut leaf disease detection and classification by using back propagation algorithm. In 2015 International Conference on Communications and Signal Processing (ICCSP) (pp. 0964-0968). IEEE.
- [9] Dong, M., Mu, S., Su, T., & Sun, W. (2019, August). Image Recognition of Peanut Leaf Diseases Based on Capsule Networks. In International CCF Conference on Artificial Intelligence (pp. 43-52). Springer, Singapore.
- [10] Sabour, S., Frosst, N., & Hinton, G. E. (2017). Dynamic routing between capsules. In Advances in neural information processing systems (pp. 3856-3866).
- [11] Vaishnav, M. P., Devi, K. S., & Ganeshkumar, P. (2020). Automatic method for classification of groundnut diseases using deep convolutional neural network. SOFT COMPUTING.
- [12] Plantvillage (2020), Topics, retrieved on August 08, 2020, from <https://plantvillage.psu.edu/topics>.
- [13] Huang, G., Liu, Z., Van Der Maaten, L., & Weinberger, K. Q. (2017). Densely connected convolutional networks. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 4700-4708).
- [14] He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 770-778).
- [15] Szegedy, C., Ioffe, S., Vanhoucke, V., & Alemi, A. (2017, February). Inception-v4, inception-resnet and the impact of residual connections on learning. In Proceedings of the AAAI Conference on Artificial Intelligence (Vol. 31, No. 1).
- [16] 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).
- [17] Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556.
- [18] Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. Advances in neural information processing systems, 25, 1097-1105.
- [19] Fuentes, A. F., Yoon, S., Lee, J., & Park, D. S. (2018). High-performance deep neural network-based tomato plant diseases and pests diagnosis system with refinement filter bank. Frontiers in plant science, 9, 1162.
- [20] Liu, B., Zhang, Y., He, D., & Li, Y. (2018). Identification of apple leaf diseases based on deep convolutional neural networks. Symmetry, 10(1), 11.
- [21] Kaya, A., Keceli, A. S., Catal, C., Yalic, H. Y., Temucin, H., & Tekinerdogan, B. (2019). Analysis of transfer learning for deep neural network based plant classification models. Computers and electronics in agriculture, 158, 20-29.
- [22] Barbedo, J. G. A. (2016). A review on the main challenges in automatic plant disease identification based on visible range images. Biosystems engineering, 144, 52-60.
- [23] Brahimi, M., Arsenovic, M., Laraba, S., Sladojevic, S., Boukhalfa, K., & Moussaoui, A. (2018). Deep learning for plant diseases: detection and saliency map visualisation. In Human and machine learning (pp. 93-117). Springer, Cham.
- [24] Mohanty, S. P., Hughes, D. P., & Salathé, M. (2016). Using deep learning for image-based plant disease detection. Frontiers in plant science, 7, 1419.
- [25] Waheed, A., Goyal, M., Gupta, D., Khanna, A., Hassani, A. E., & Pandey, H. M. (2020). An optimized dense convolutional neural network model for disease recognition and classification in corn leaf. Computers and Electronics in Agriculture, 175, 105456.
- [26] Ji, M., Zhang, K., Wu, Q., & Deng, Z. (2020). Multi-label learning for crop leaf diseases recognition and severity estimation based on convolutional neural networks. Soft Computing, 24(20), 15327-15340.
- [27] Zhong, Y., & Zhao, M. (2020). Research on deep learning in apple leaf disease recognition. Computers and Electronics in Agriculture, 168, 105146.
- [28] Sethy, P. K., Barpanda, N. K., Rath, A. K., & Behera, S. K. (2020). Deep feature based rice leaf disease identification using support vector machine. Computers and Electronics in Agriculture, 175, 105527.
- [29] Zeng, W., & Li, M. (2020). Crop leaf disease recognition based on Self-Attention convolutional neural network. Computers and Electronics in Agriculture, 172, 105341.
- [30] Zhang, S., Zhang, S., Zhang, C., Wang, X., & Shi, Y. (2019). Cucumber leaf disease identification with global pooling dilated convolutional neural network. Computers and Electronics in Agriculture, 162, 422-430.
- [31] Karlekar, A., & Seal, A. (2020). SoyNet: Soybean leaf diseases classification. Computers and Electronics in Agriculture, 172, 105342.
- [32] Jeremy Howard (2018), Now anyone can train Imagenet in 18 minutes, retrieved on Jan 1, 2021, from <https://www.fast.ai/2018/08/10/fastai-diu-imagenet/>.
- [33] Lin, T. Y., Goyal, P., Girshick, R., He, K., & Dollár, P. (2017). Focal loss for dense object detection. In Proceedings of the IEEE international conference on computer vision (pp. 2980-2988).