EARLY RECOGNITION OF MUNG LEAF DISEASES BASED ON SUPPORT VECTOR MACHINE AND CONVOLUTIONAL NEURAL NETWORK

AKRUTI NAIK[1]
HETAL THAKER[1]

[1] Atmiya University,
Department of Computer Applications ,
Yogidham Gurukul,Kalawad Road, Rajkot,Gujarat , India

Abstract. This paper proposed a model that Classifies a Mung (Vigna mungo L.) leaf to check if it is healthy or infected with a disease with the aid of Machine Learning and Deep Learning algorithms. The dataset is created in a controlled environment, where a controlled environment is a data item (image) that comprises only a single subject (leaf) and a white background collected from the south Gujarat Region in India. SVM and CNNs with different architectures have been trained and compared to each other. It aimed at detecting 3 mung leaf disease categories and a healthy leaf category. The model extracts complex features of various diseases. Comparative experiment results show that in the proposed work SVM overfit the data and CNN achieves 95.05 percentage of identification accuracy on the Mung leaf image dataset. Early detection will help farmers to improve their productivity. The main objective was to automate Mung Leaf disease identification using advanced deep learning approaches and image data.

Keywords: Mung leaf, Classification in Machine Learning, SVM, Deep Neural Networks, Convolutional Neural Networks.

(Received November 23rd, 2021 / Accepted May 10th , 2022)

1 Introduction

Mungbean is also known as green gram which is a highly nutritious legume crop and considered as a quality pulse due to its rich protein content and excellent digestibility. India is the largest producer of mungbean where it is the third most important pulse crop with an area of approximately 34.50 lakh hectares with 15.91 lakh tonnes of total production [32]. To meet global demand, it is commanding to increase the current average global productivity [23]. Having diseases in plants is a natural process. In traditional practice, farmers try to evaluate the diseases by their past experience. Or in other cases, the expert observes the plant organs like leaves and stems for any diseases. It is a very time-consuming and costly method because it requires continuous monitoring by an expert in large fields. Early identification and treatment will help farmers to reduce the overall loss. We require a fast approach to protect the crop from diseases. Using advanced technology like mobile phones, tablets, and similar devices farmers can input data in form of digital images and get an immediate response. That will result in crop productivity. We need automated approaches that can support farmers in the early detection and prevention of Mung leaf diseases. Machine Learning and Deep Learning are part of Artificial Intelligence that emphasizes making predictions using algorithms that increase automatically through experience and by the use of data. Algorithms build an inference model based on training data to generalize the context and make predictions. Different deep learning models such as VGG16, MobileNetV2, and

Custom CNN were implemented. Here we try to classify a mung leaf to check if it is healthy or diseased. Our model performance showed favourable results.

2 Related Work

The yield of mungbean is affected by several diseases from which the three most common mung leaf diseases are Cercospora Leaf Spot, Powdery Mildew, and Yellow Mosaic Virus. Mungbean yellow mosaic disease (MYMD) is one of the major destructive diseases of mungbean in India. It was first reported on Mungbean from India in 1940[18], since then, it has been reported from all over India and other countries of the Indian subcontinent[31]. When it is severe crop losses extent up to 85-90 percentage. Itâs considered to be a potential threat to the cultivation of not only mungbean but also in other species like soybean, urdbean, moth bean, and cowpea[15]. CLS is the most widespread and destructive fungal disease of the mungbean. Cercospora leaf spot disease caused by Cercospora spp. The disease was first time reported in Delhi, India[17]. Cercospora leaf spot is also causing serious losses to mungbean crop. It 58 percentage of yield loss annually[12]. The disease starts appearing about 30-40 days after planting. The leaf spots develop on infected leaves with a somewhat circular/subcircular to broadly irregular shape, the central area turn reddish-brown and grey center surrounded by a dark brown margin. Powdery mildew diseases in mungbean caused by the fungal pathogen Erysiphe polygoni. Yield losses due to the disease were reported to be up to 20-40 percentage at the reproductive stages[6], but the damage can be more serious when the epidemic starts at the reproductive stages it may reach up to 55 percentage[20]. Each disease has unique symptoms that appear on the leaf, which can be used to categorize the disease by deep learning algorithms [27][10][21][4]. Deep learning for plant disease detection in the primary works[7][22][5][9] all used leaf images as a data source. However, these methods require a lot of data to work accurately, and this might be a challenge. Initially, gathering new data for the problem domain, for example, object identification in biomedical or medical images may be difficult[30][8][2]. Besides, once the images have been gathered, they must be manually labelled and this is a laborious task and involves an expertâs view to conducting it properly[33]. Data augmentation is an effective method that deals with a limited amount of data [26][25]. This method generates new training samples from the original dataset by applying transformations to them. Several libraries like Tensorflow[1], Augmentor [3], or Imgaug [11], provide features for augmentation. However, these libraries not meant to be deal with object localization, object detection, semantic segmentation, and instance segmentation. Transformation methods used to perform augmentation on images may alter the notation but do not change the class of the image. For illustration, the horizontal or vertical flip operation to an image does not change the class of image, but only the location of the objects in the new image has changed. So for each problem, a special-purpose technique needs to be implemented, or augmented images must be manually labelled. Both the solutions are not feasible when there are hundreds or thousands of images to deal with.

For training using Convolutional neural networks (CNNs) require a large number of sample images. Collecting required images is time-consuming and costly in many applications [13]. For limited dataset conditions, many researchers combined deep learning with transfer learning for data expansion [19]. A method of deep learning model combined with transfer learning proposed by Srdjan [28] classifies 13 different diseases and healthy plant leaf and reaches 96.3 percentage of average accuracy. [14] increases the size of the training dataset 12 times by applying rotation, mirroring, brightness, and contrast adjustment and adding Gaussian noise, and reducing the overfitting problem.

[16] classify and recognize 54,306 diseased and healthy plant leaf images using GoogleNet and AlexNet and conclude that GoogleNet provides a better average classification effect than AlexNet and achieves accuracy on the test set up to 99.35 percenatge. [33] trained a chain of deep convolutional neural networks that detect the severity of diseases and found that VGG16 is the best model and achieves 90.4 percentage of accuracy.[29] performs comparative test and verified that compared to VGG and ResNet, DenseNets requires fewer parameters and less calculation time to achieve advanced performance and achieve 99.75 percenateg test accuracy. Three different CNN architectures were retrained by [24]using the transfer learning method and deep transfer learning was performed using pre-trained models that generate networks that could make accurate predictions. Three methods of regression, focus loss function, and multilabel classification based on DenseNet-121 CNN was proposed by [34] to identify apple leaf diseases and achieve 93.51, 93.31, and 93.71percentage accuracy on the test set.

3 Proposed Model

The proposed algorithms for this work we have used Support Vector Machine and Convolutional Neural Networks to detect mung leaf disease through machine-learning and deep learning. Support Vector Machines

(SVMs) is a model that can be used for both classification and regression. The algorithm tries to find a decision boundary, or a hyperplane when data is characterized in more than two dimensions that splits the classes. A Convolutional Neural Network is a type of neural network that can successfully recognize the Spatial and Temporal dependencies in the data by passing through multiple filters. It is frequently used with images. The architecture of the Convolutional Neural Network is designed in such a way that it performs better because of the relatively different number of parameters involved and the reusability of weights. The pre-processing steps required for ConvNets are considerably less compared to traditional machine learning algorithms. Each image when training goes through a series of operations, known as convolutions, a dot product of a 2D kernel of a specified size is slid over the image and the small region of the image the kernel is connected to. The resultant is then followed by an activation function like ReLU (Rectified Linear Unit) and then followed by a Pooling layer that generally reduces the image resolution by making it half the number of pixels. After this, stacked layers of the fully connected layer are usually added to learn non-linear combinations of the high-level features presented by the convolutional layers. But before passing the feature maps to the fully connected layers, there is a need to flatten the features maps. Figure 1 shows the structure of the Mung Leaf disease detection system.

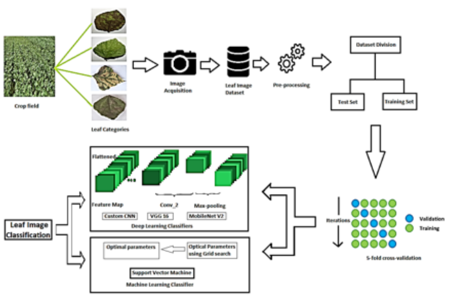

Figure 1: The Structure of Mung Leaf Disease Detection

4 Dataset and PreProcessing

4.1 the Dataset

The dataset primarily accounts for a controlled environment. An image from the controlled environment contains a single mung leaf with a white background i.e. no noise. After image acquisition, the images are manually screened to avoid duplication and classification in the dataset. Finally, a dataset contained a total of 883 Mung leaf images for controlled environments: Cercospora (224), Healthy (211), Powdery Mildew (225), and Yellow Mosaic (223) is obtained. After that size of each picture is fixed at 256 x 256. A leaf can be one of the four distinct categories i.e. Healthy, Cercospora, Yellow Mosaic, and Powdery Mildew. The images of Mung leaves in 4 categories are shown in Figure 2.

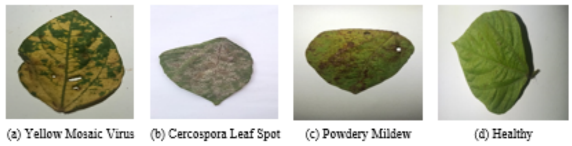

Figure 2: All Categories (Controlled Environment)

4.2 Preprocessing

In this study SVM and ConvNets are built with different model architectures and hence need to implement different preprocessing steps. The image first needs to be read into a three-dimensional NumPy array and rescaled to one-third its size to train an SVM. In Computer Vision, the Histogram of Object Gradients (HOG) is used for object detection. HOG act as feature descriptors by focusing on the shape or structure of the object. In the end, a histogram is created for each local region of the image. The image is converted to grayscale before applying this method. Mapping of the label is done using integers 1 through 4. Figure 3 displays a converted grayscale leaf image and HOG image. Before applying this technique, we change the leaf image to grayscale. As we are using Sklearnâs SVM, so there is no need to hot encode the training labels. Labels are mapped here to the integers 1 through 4. When using ConvNets, low preprocessing steps prove to be sufficient to get decent results. The image is first read into a 3-dimensional NumPy array and then resized to a size of 256 x 256 pixels. Data normalization ensures that each pixel of the image has a similar data distribution and helps to converge faster while training the model.

4.3 Data Augmentation

A ConvNet is said to have invariance when it is robust enough to classify objects even in different orientations. A model can be trained to be invariant to size, translation, and even illumination. We can generate additional synthetically modified data to train our network prediction accurately in various conditions. This is known as Data Augmentation. It involves augmenting the datasets with perturbed versions of themselves. Various Augmentation properties are applied to the dataset to create new images. This paper used a variety of image enhancement techniques for enhanced image data.

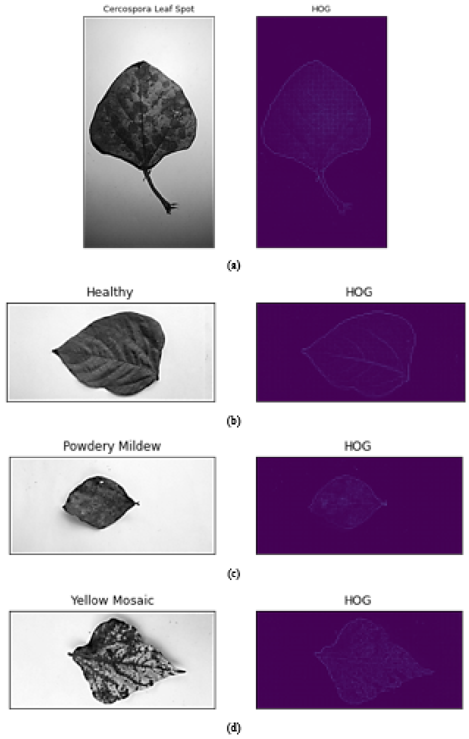

Figure 3: A leaf image after applying HOG: (a) Cercospora Leaf Spot, (b) Healthy, (c) Powdery Mildew, (d) Yellow Mosaic Virus

Table 1 shows various augmentation properties applied to the dataset.

Table 1: Data Augmentation Property

$Property$	Min	Max
$WidthShiftRange$	0	0.2
$HeightShiftRange$	0	0.2
$BrightnessRange$	0.8	1.8
$ZoomRange$	0	0.3
$Flip$	Horizontal	vertical

5 Training Process

5.1 Support Vector Machine

An SVM is capable of providing high accuracy compared to other machine learning models like logistic regression etc. To handle nonlinear input spaces, the SVM uses a kernel trick to map the data to a higher dimension so that it is possible to find a hyperplane that divides the different classes. Sklearn.svm.SVC provides a Support Vector Classifier. An SVC with a polynomial kernel and the regularization parameter C is set on a âOne vs. One Strategyâ. This strategy is a heuristic method to use binary classification on all the classes one by one for multiclass classification. Each binary classification predicts one class label and the model with the most predictions is predicted by the one-by-one strategy. SVM after training yielded a test accuracy of 86.9 percentage but it overfits the dataset and thus was not a reliable model. Figure 4 shows the confusion matrix for 4 Mung leaf categories that include 3 diseased and a healthy category and Table 2 shows the classification report for the controlled environment with SVM.

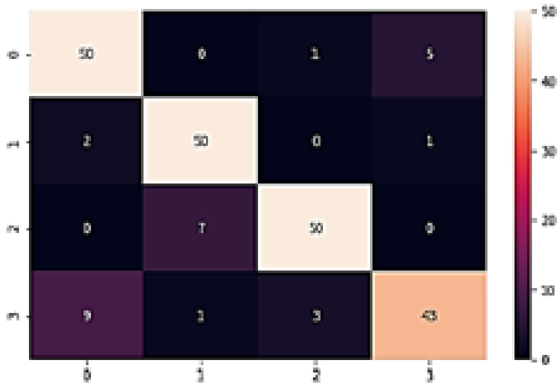

Figure 4: Confusion Matrix (Controlled); 0-Cercospora, 1-Healthy, 2-Powdery Mildew, 3-Yellow Mosaic

Table 2: Classification Report

	$Preci-sion$	$Recall$	$f1-Score$	$Support$
0	0.82	0.89	0.85	56
1	0.86	0.94	0.90	53
2	0.93	0.88	0.90	57
3	0.88	0.77	0.82	56
$Accuracy$			0.87	222
$MacroAvg$	0.87	0.87	0.87	222
$WeightedAvg$	0.87	0.87	0.87	222

0	$Cercospora$
1	$Healthy$
2	$PowderyMildew$
3	$YellowMosaicVirus$

5.2 Data Augmentation

To regularize the effect of overfitting, different values for the regularization parameter and other hyperparameters are tried. Grid Search is used to find the hyperparameters that yield better accuracy and do not overfit. Various hyperparameters like C, Gamma, Kernel, Degree, and Strategy with a set of values are applied on Grid Search with 5-fold cross-validation.

Training accuracy of the model reached 100 percentage and test accuracy fell to 86.4 percentage when the results of the grid search were applied.

5.3 Convolutional Neural Networks

CNN models with different architectures are trained to complete our objective. The comparison process is divided into 2 different rounds. A batch size of 64, input shape (256, 256, 3), and a learning rate of 0.0003 are maintained throughout the comparison. In the first round, a custom CNN architecture and two pre-built models: VGG16 and MobileNetV2 are compared with each other. In the convolutional layer, the padding âsameâ is applied in all three models. Except for the Custom CNN, the pre-trained weights for the other models are loaded and thus its training process is Transfer Learning. All three models are trained for 20 epochs and categorical cross-entropy as their loss function. MobileNet V2 model performs very poorly and also overfits the dataset with a huge difference. On the other hand, the VGG16 architecture performs the best with a test accuracy of 95.5 percentage. Our Custom CNN model also performs decently with a test accuracy of 89.9 percentage but can still be improved by hyperparameter tuning. Data Augmentation did not positively affect the training process and thus is not applied in the training process. Table 3 displays Custom CNN model architecture. Here, Figures 5 displays accuracy and loss graphs. The accuracy and loss comparison is displayed in below table 4.

Table 3: Model Architecture (Custom CNN)

Layey	Filters	Kernal	Act	MP
Conv2D	32	3 X 3	ReLu	Y
Conv2D	64	3 X 3	ReLu	Y
Conv2D	128	3 X 3	ReLu	Y
Conv2D	128	3 X 3	ReLu	Y
Conv2D	256	3 X 3	ReLu	Y
layer	Neurons		Act	
Flattern	-	-	-	-
Dense	256	-	ReLu	-
Dense	128	-	ReLu	-
Dense	128	-	ReLu	-
Dense	64	-	ReLu	-
Dense	4	-	Softmax	-

In the second round, we try to tune the hyperparameters of the custom CNN model to yield the best results we could. The following parameter grid combination is used to search for the best hyperparameters: After performing the hyperparameter tuning using Keras-tuner, we found that the best hyperparameter for our Custom

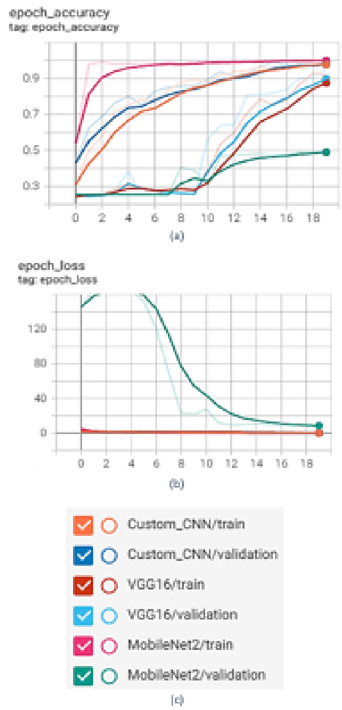

Figure 5: : (a) Accuracy Graph for Custom CNN, VGG 16, and Mobilenet V2 for train and validation, (b) Loss Graph for Custom CNN, VGG 16, and Mobilenet V2 for train and validation, (c) Legends

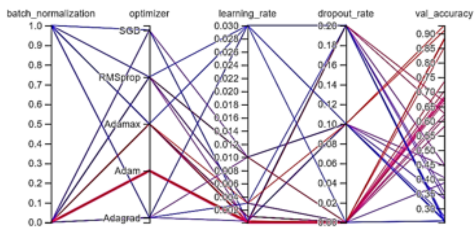

Figure 6: Keras Tuner Results (Controlled - CNN)

Table 4: : Accuracy and Loss Metrics in the first round

Model	Accuracy (Train/Test)	Loss (Train/Test)
$CustomCNN$	96.67 / 89.19	0.1006 / 0.2553
$VGG16$	93.65 / 95.5	0.2182 / 0.1487
$CMobileNetV2$	99.8 / 35.14	0.0001 / 29.7

Table 5: : Parameter grid (controlled â CNN)

Batch Norm.	True, False
Optimizer	SGD, RMSprop, Adagrad, Adadelta, Adam, Adamax
Learning Rate	0.01, 0.03, 0.001, 0.003, 0.0001, 0.0003
Dropout rate	0.0, 0.1, 0.2

CNN model was:

Batch Norm. - False Optimizer - Adamax Dropout rate - 0.1 Learning rate - 0.0003.

The CNN model has trained again but with the given hyperparameters and yields a training accuracy of 97.67 percentage and testing accuracy of 91.03 percentage Figures 9 represent the accuracy and loss graphs for training and validation.

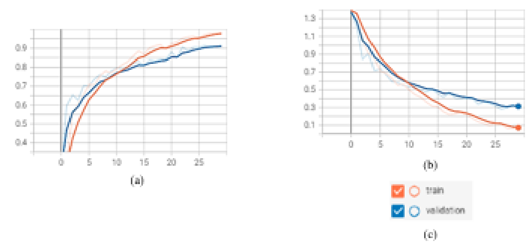

Figure 7: : (a) Accuracy Graph for Custom CNN, (b) Loss Graph for Custom CNN, (c) Legends

6 Conclusion

6. CONCLUSION In the controlled environment, even after applying high regularization, the SVM overfit the data. Due to the lack of sufficient data to train the model, it overfits. In machine learning, a huge sample of data is needed to decently predict any activity. CNN proved to be robust in a Controlled Environment despite the lack of huge amounts of data. They successfully captured features from the image and classified them with a test accuracy of 95.05 percentage.

7 Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

8 Bibliography References

References

[1] Abadi, M. Tensorflow: Large-scale machine learning on heterogeneous systems. Accessed 8.

[2] Asperti, A. and Mastronardo, C. The effectiveness of data augmentation for detection of gastrointestinal diseases from endoscopical images. *CoRR*, 03689:1â7.

[3] Bloice, M., Stocker, C., and Holzinger, A. Augmentor: An image augmentation library for machine learning. *CoRR*, 04680:1â5.

[4] Ciresan, D., Meier, U., and Schmidhuber, J. *Multi-column deep neural networks for image classification*. CVPR, In.

[5] Ferentinos, K. Deep learning models for plant disease detection and diagnosis. *Comput Electron Agric*, 145:311â8.

[6] Fernandez, G. and Shanmugasundaram, S. The avrdc mungbean improvement program: The past, present and future. In Mungbean. Eds. Shanmugasundaram, S. and McLean, B., editors, *Proceedings of the Second International Symposium held at Bangkok*, page 58â70, Thailand.

[7] Fuentes, A., Yoon, S., Kim, S., and Park, D. A robust deep-learning-based detector for real-time tomato plant diseases and pestâs recognition. *Sensors*, 2017;17.

[8] Galdran, A. Data-driven color augmentation techniques for deep skin image analysis. *CoRR*, 03702:1â4.

[9] Godliver, O., Friedrich, M., Mwebaze, E., Quinn John, A., and Biehl, M. Machine learning for diagnosis of disease in plants using spectral data. *Intâl Conf. Artificial Intelligence*, ICAIâ18.

[10] Huang, K. Application of artificial neural network for detecting phalaenopsis seedling diseases using color and texture features. *Comput Electron Agric*, 57:3â11.

[11] Jung, A. imgaug: a library for image augmentation in machine learning experiments. Accessed 8.

[12] Lal, G., Kim, D., Shanmugasundaram, S., and T, K. Mungbean production. *AVRDC*, page 6.

[13] Lee, S., Chan, C., and Wilkin, P. Deep-plant: Plant identification with convolutional neural networks. In *IEEE International Conference on Image Processing*, volume 2015, page 452â456.

[14] Liu, B., Zhang, Y., and He, D. Identification of apple leaf diseases based on deep convolutional neural networks. *Symmetry*, 10(1).

[15] M., P., G., P. Y., S., D., M., T., K., G., G., R. R., and B, V. Hyperspectral remote sensing of yellow mosaic severity and associated pigment losses in vigna mungo using multinomial logistic regression modelsâ. *Elsevier Crop Protection*, 45:132â140.

[16] Mohanty, S., Hughes, D., and SalathÃ©, M. Using deep learning for image-based plant disease detection. *Front Plant Sci*, 7(1419).

[17] Munjal, R., Lall, G., and Chona, B. Some cercospora species from india-iv. *Indian*, (ytopatholology,13):144â149.

[18] Nariani, T. Yellow mosaic of mungbean (phaseolus aureus. *Indian Phytopathol*, 13:24 29.

[19] Pan, S. Yang qa survey on transfer learning. *IEEE Transactions on Knowledge Data Engineering*, 2010,22(10):1345-1359.

[20] Poehlman, J. *The mungbean*. Westview Press, Boulder, Colo.

[21] Price, T., Gross, R., Wey, J., and Osborne, C. A comparison of visual and digital image-processing methods in quantifying the severity of coffee leaf rust (hemileia vastatrix. *Aust J Exp Agric*, 33:97â101.

[22] Ramcharan, A., Baranowski, K., McCloskey, P., Ahmed, B., Legg, J., and Hughes, D. Deep learning for image-based cassava disease detection. *Front Plant Sci*, 8(1852).

[23] R.M., N., Schafleitner, R., Kenyon, L., Srinivasan, R., Easdown, W., Ebert, A., and Hanson, P. Genetic improvement of mungbean productivity. In *Proc. Of the 12th SABRAO Congress on Plant Breeding towards 2025: Challenges in a Rapidly changing World*, page 27â28, Chiang Mai, Thailand.

[24] Selvaraj, M., Vergara, A., and Ruiz, H. Ai-powered banana diseases and pest detection. *Plant Methods*, 15(92):7â019â0475â.

[25] Simard, P., Steinkraus, D., and Platt, J. Best practices for convolutional neural networks applied to visual document analysis. In editor., editor, *Society IC*, volume ICDARâ03), vol. 2, page 958â64, Edinburgh. IEEE Computer Society.

[26] Simard, P., Victorri, B., LeCun, Y., and Denker, J. Tangent prop â a formalism for specifying selected invariances in an adaptive network. In *Proceedings of the 4th International Conference on Neural Information Processing Systems (NIPSâ91). Advances in Neural Information Processing Systems*, volume 4, page 895â903, Denver. MIT Press.

[27] Sladojevic, S., Arsenovic, M., Anderla, A., Culibrk, D., and Stefanovic, D. Deep neural networks based recognition of plant diseases by leaf image classification. *Comput Intellig Neurosci*, 2016(11).

[28] Srdjan, S., Marko, A., and Andras, A. Deep neural networks based recognition of plant diseases by leaf image classification. *Comput Intell Neurosci*, 6:1â11.

[29] Too, E., Li, Y., and Njuki, S. A comparative study of fine-tuning deep learning models for plant disease identification. *Comput Electron Agric*, 161:272â9.

[30] Valle, E. Data, depth, and design: Learning reliable models for melanoma screening. *CoRR*, 00441:1â10.

[31] Verma, A., Dhar, A., and Malathi, V. Cloning and restriction analysis of mungbean yellow mosaic virus. In *International Conf. Virology in the Tropics Lucknow, India*, page â14.

[32] Vinod, K. and Pandey, S. Current status of mungbean in madhya pradesh - a review. *International Journal of Current Microbiology and Applied Sciences, ISSN*, 7706:1062â1072.

[33] Wang, X. Chestx-ray8: Hospital-scale chest x-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases. In *Proceedings of the 2017 IEEE Computer Society Conference on Computer Vision*

and Pattern Recognition (CVPRâ17). CVPR â17, Hawai. IEEE Computer Society.

[34] Yong, Z. and Ming, Z. Research on deep learning in apple leaf disease recognition. *Comput Electron Agric*, 168(105146):2019 10514 6.