

Using Transfer Learning for Nutrient Deficiency Prediction and Classification in Tomato Plant

Vrunda Kusanur¹

Research Scholar, Visvesvaraya Technological University
Bengaluru, Karnataka, India

Dr. Veena S Chakravarthi²

Professor, Department of Electronics and Communication
Engineering, BNMIT, Bengaluru, Karnataka, India

Abstract—Plants need nutrients to develop normally. The essential nutrients like carbon, oxygen, and hydrogen are obtained from sunlight, air, and water to prepare food and plant growth. For healthy growth, plants also need macronutrients such as Potassium, Calcium, Nitrogen, Sulphur, Magnesium, and Phosphorus in relatively great quantities. When a plant doesn't find necessary nutrients for its growth inadequate amount, deficiency of plant nutrients occur. Plants exhibit various symptoms to indicate the deficiency. Automatic identification and differentiation of these deficiencies are very important in the greenhouse environment. Deep Neural Networks are extremely efficient in image categorization problems. In this work, we used the part of the pre-trained deep learning model i.e. Transfer Learning model to detect the nutrient stress in the plant. We compared three different architectures including Inception-V3, ResNet50, and VGG16 with two classifiers: RF and SVM to improve, classification accuracy. A total of 880 images of Calcium and Magnesium deficiencies in the Tomato plant from the greenhouse were collected to form a dataset. For training, 704(80%) images are used and for testing, 176(20%) images are used to examine the model performance. Experimental results demonstrated that the largest accuracy of 99.14% has resulted for the VGG16 model with SVM classifier and 98.71% for Inception-V3 with Random Forest Classifier. For a batch size of 8 and epochs equal to 10, the Inception -V3 architecture attained the highest validation accuracy of 99.99% and the least validation loss of 0.0000384 on an average.

Keywords—Nutrient deficiency; plant nutrients; deep neural networks; transfer learning; random forest (RF); support vector machine (SVM)

I. INTRODUCTION

A proper combination of nutrients is required for plants to live, develop and reproduce. So, plant analysis is a necessary tool that assists farmers by providing significant information about the nutritional description of the growing plant to obtain a better yield. Generally, plant analysis indicates the analysis of magnesium (Mg), sulphur (S), phosphorous (P), calcium (Ca), nitrogen (N), potassium (K), etc. Normally, plants show signs of being unhealthy when they suffer from undernutrition. For example, yellow around the edges of its leaves are a sign of magnesium deficiency. Yellow spots between the leaf veins and Blossom End Root denote the absence of calcium. Brown edges along the plant leaves indicate a deficiency of potassium. Yellow or pale green leaves imply the need for nitrogen [1]. These nutrient deficiency symptoms will help growers to identify the nutrient status of plants for a better crop yield. Manually diagnosing these deficiencies is a difficult task. So,

the key objective of this work is to automate the identification of nutrient deficiencies in plants using Convolutional Neural Networks (CNN).

Artificial Intelligence has numerous applications in multiple industries, healthcare, environment, finance, education, agriculture, etc. to solve complex problems and make our daily life more secure and fast.

G. Madhulatha et al. [2] proposed an automatic plant disease detection on the plant leaves to decrease crop loss and increase productivity. Plant diseases are predicted and classified with 96.50% accuracy based on visual symptoms using deep CNN. The authors used a dataset from the "Plant-Village" dataset for plant leaf diseases. The model was pre-trained using AlexNet. Muhammad Hammad Saleem et al. [3] developed three Deep Learning meta-architectures namely; Faster Region-based Convolutional Neural Network (RCNN), Single Shot MultiBoX Detector (SSD), and Region-based Fully Convolutional Networks (RFCN) to recognize plant disease and healthy leaves. All three models include a feature extractor and a base network. This research used Gradient Descent with its Momentum version, Adaptive Moment Estimation (Adam), and Root Mean Square Propagation (RMSProp) optimization algorithms to increase the performance of the Deep Learning meta-architectures. The authors examined that all the Deep Learning meta-architectures needed 126 epochs (200,000 iterations) for training convergence. When the SSD model was trained using Adam optimizer, the maximum means Average Precision (mAP) of 73.07% was obtained. Guan Wang et al. [4] suggested a deep learning model for control plant disease application. The authors used the apple leaf black rot images produced by the fungus Botryosphaeria obtuse from the PlantVillage dataset for disease severity classification. The highest overall accuracy of 90.4% was obtained for the VGG16 model. Sharada P. Mohanty et al. [5] established a smartphone-assisted application to detect the disease using a deep convolutional neural network. In this research, GoogLeNet architecture performs better and provides 99.35% accuracy as compared to AlexNet architecture. The presently available deep learning methods to identify the plant disease were reviewed by M. Nagaraju and Priyanka Chawla [6].

Many previous works have considered Image Recognition and Machine Learning models to classify the images into healthy and unhealthy images. However, most of these algorithms require image segmentation and feature extraction. But, from the many extracted features, it is difficult to judge

the important and dominant features for plant disease detection. Moreover, under difficult background circumstances, many techniques fail to successfully segment the leaf and will lead to unreliable deficiency recognition. So, image segmentation and feature extraction are still challenging tasks. Therefore, automatic plant disease detection and nutrient deficiency recognition are still challenging tasks. Recently, Convolutional Neural Network (CNN) is becoming the preferred scheme to overcome few challenges.

The main objective of this research is to diagnose nutrient deficiency in plants and take several measures like adjusting the pH value of water to achieve a quality yield, providing the right amount of fertilizer, etc. using deep learning models. For nutrient deficiency classification, we employed the Transfer Learning method, where pre-trained models are used as the entry point to develop the neural network models. In this research, we have used these models to predict Calcium (Ca) and Magnesium (Mg) deficiency in tomato crops grown under a greenhouse environment.

The key advantage of transfer learning is that instead of beginning the learning process from the scratch, the model commences from the characteristics that have been educated when resolving other problems which are analogous to the one being resolved. We have used three pre-trained models- InceptionV3, VGG16, and ResNet50 as a base model and SVM or Random Forest classifier on top of it to attain better results.

The rest of this paper is structured in the following fashion. Section II introduces the images collected to form the dataset of Ca and Mg deficiencies followed by related concepts. This section also presents Inception V3, ResNet50, and VGG16 architectures, and the proposed model to identify and classify the deficiencies. Section III dedicated to the evaluation, and the comparative analysis of results obtained in this experiment. In Section IV, the paper is summarized and future work is mentioned.

II. MATERIALS AND METHODS

A. Data Acquisition

Tomato plants were grown in a greenhouse of a size 10x4 sq.ft. to study and gather the dataset for lack of nutrients in tomato leaves and fruits. The calcium and magnesium deficiencies were induced for the plants in different stages and their images were captured from the camera for training and testing the performance of the model. The dataset was developed with two classes for classification and prediction: Calcium and Magnesium. Altogether, there are 880 images in the dataset. Out of 880 images, 704 (80%) images are for training the model, and 176 (20%) images are for testing the model. There are 374 calcium deficiency and 330 magnesium deficiency images in the training dataset. Further, out of 176 testing images, 94 images are of calcium, and the remaining 82 images are of magnesium deficiency images. To enhance the dataset, the data augmentation methods including image resizing, flipping, random rotation, shearing, etc., are applied. The details of calcium and magnesium nutrient deficiency symptoms in tomatoes are presented in Table I. 256 x 256

pixels is the size of all the resized images. These sample images are input to the convolutional neural network for training the model. The trained model is applied for the class prediction of unseen images. These phases are explained in detail in the following sections. Machine learning algorithms including SVM, Decision-Tree and, RF are excellent in resolving classification problems [10]. However, they go wrong in extracting the proper features from the image. Alternatively, Convolutional Neural Networks receives the raw pixel of the images directly as inputs instead of extracting certain features manually [12-14]. CNN learns how to take out these features from the actual image.

B. Convolutional Neural Networks

CNN's are a class of Deep Neural Networks that can identify and categorize specific features in images and are generally used for examining visual images. Significantly, CNN can yield good results than the traditional feature extraction algorithms in plant disease diagnosis [15-18]. In CNN, the filters are learnable. A classic CNN consists of two components: The Convolution Block and the Fully Connected block, which are detailed as follows.

TABLE I. CALCIUM AND MAGNESIUM NUTRIENT DEFICIENCY SYMPTOMS IN TOMATO

Nutrients	Description	Deficiency Symptoms on Leaf/ Fruit
Calcium (Ca)	Young leaves curl inwards and cause dry decaying areas at Blossom End of the fruit (BER) in Tomatoes.	
Magnesium (Mg)	Interveinal chlorosis (Veins of Leaf remains green whereas the areas between the leaves go yellow). Elder leaves drip their color excluding in the veins. It does not affect the fruit.	

C. Convolutional Neural Networks

CNN's are a class of Deep Neural Networks that can identify and categorize specific features in images and are generally used for examining visual images. Significantly, CNN can yield good results than the traditional feature extraction algorithms in plant disease diagnosis [9, 11]. In CNN, the filters are learnable. A classic CNN consists of two components: The Convolution Block and the Fully Connected block, which are detailed as follows.

1) *Convolution block*: The convolution block contains the Convolution Layer and the Pooling Layer. In this block, the task of feature extraction is accomplished. The convolutional layer produces the feature maps or activation maps by applying filters to input images using the ReLU activation function. The ReLU function returns x for all the values of $x > 0$, and returns 0 for all values of $x \leq 0$ and is given in equation 1.

$$F(x) = \max(0, x) \quad (1)$$

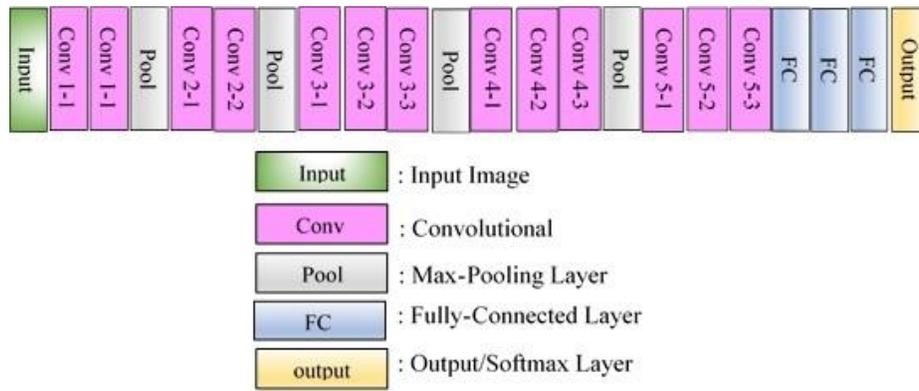


Fig. 1. VGG16 Architecture

Fig. 1. VGG16 Architecture.

The convolutional layer uses filters kernels to recognize various features like edges, horizontal lines, vertical lines, etc., in an image. To extract more composite and thoughtful features, the same size convolution kernel is used again and again multiple times. The pooling layer is enforced next to a convolution layer in which a down sampling operation is performed on a convolved feature to scale down the number of dimensions of the feature map. Commonly, the average and maximum values are selected by the pooling layer for this task.

2) *Fully connected block*: The Fully Connected block comprises of fully connected simple neural network design which does classification depending on convolutional block inputs. Convolutional Neural Network has one or more fully connected layers at the end of it. At the end of the fully connected layer, there is a softmax activation function whose output is a probability (from 0 to 1) for every classification label.

D. VGG16 Model Architecture

VGG is a pre-trained model and has 138 Million parameters. VGG is trained over 14 million images belonging to 1000 classes and learned to detect generic features from images. There are 16 and 19 weight layers in the network for VGG-16 and VGG-19 respectively.

This research work uses VGG-16 as the base model and altered it to create a different network. As VGG16 attains 92.7% test accuracy in ImageNet, and because of its high performance, the pre-trained weights are retained and only the top three Fully Connected Layers or Dense Layers are modified to fine-tune the neural network. In this work, the features extracted from VGG16 are given as input to RF or SVM Classifiers to reduce the training time and increase the classification accuracy. Fig. 1 describes the VGG16 scheme. All the resized images in this model are of fixed size 244x244.

The VGG16 model used NVIDIA Titan Black GPUs and was trained for weeks. The VGG16 model can categorise the images into 1000 classes. The VGG model handles the input image and yields the vector of 2 values. y^\wedge denotes the probability of classification for the corresponding class and is given by equation 2.

$$y^\wedge = \begin{bmatrix} y_0^\wedge \\ y_1^\wedge \end{bmatrix} \quad (2)$$

Where, y_0^\wedge represents the probability with which class 0 (Ca) is predicted and y_1^\wedge represents the probability with which class 1(Mg) is predicted. The RGB image of constant size 224x224 is the input to the conv1 layer. The image is moved through several convolutional layers. Each layer uses a small 3x3 or 1x1 filter. Five max-pooling layers perform spatial pooling. A 2x2 pixel window with a stride of 2 is used to implement max-pooling. There are three Fully-Connected (FC) layers where there are 4096 channels in each of the first two layers and the third layer comprises 1000 channels. The softmax layer is the terminating layer. All networks have a similar configuration of the fully connected layers. A non-linear ReLu activation function is used by all hidden layers.

E. Inception-V3 Model Architecture

Inception-V3 is the most generally used CNN architecture and achieved more than 78.1% accuracy for image prediction on the ImageNet dataset. The model comprises Convolution Layers, Max pooling Layers, Average pooling Layers, Concat Layers, Dropout Layers, and Fully Connected Layers. In Inception V3, the resized images are of size 299x299x3 pixels. The structure of Inception-V3 is analogous to Inception-V2 with few modifications including Label Smoothing Regularization, Batch normalization, Auxiliary Classifier. Use of Factorized 7x7 convolutions. Inception-V3 is a CNN with 48 layers in depth. The inception model is a concatenation of parallel convolution layers with 1x1, 3x3, 5x5, etc. sized filters and a max pooling layers of 3x3 matrix. The error rate improved to 0.2 % by adding label smoothing in Inception-V3 architecture. Fig. 2 describes the Inception-V3 scheme.

Inception-V3 model used RMSProp optimizer that offered significant results in connection with accuracy and time to achieve it. RMSProp is fixed as a default optimizer. The update dynamics in the Inception-V3 model are given by equation 2 and equation 3.

$$g_{k+1}^{-2} = \alpha g_k^{-2} + (1 - \alpha) g_k^2 \quad (3)$$

$$w_{k+1} = \beta w_k + \frac{\eta}{\sqrt{g_{k+1}^{-2} + \epsilon}} \nabla f(w_k) \quad (4)$$

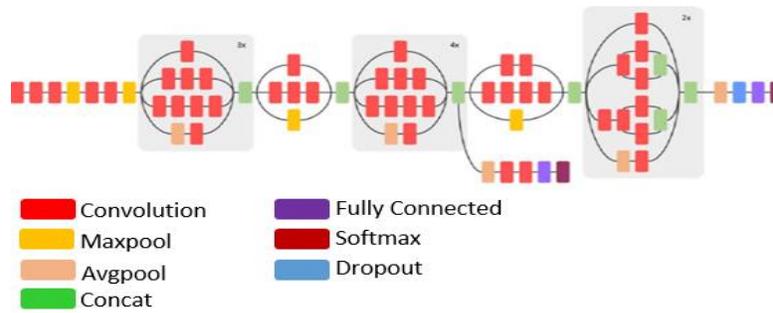


Fig. 2. Inception V3 Architecture.

Where, g_k is gradient descent at time k , g_{k+1} is gradient descent at time $k+1$, w_k is the weight at time k , w_{k+1} is the updated weight at the time $k+1$, ' α ' is the step size, ' β ' is known as momentum and ' ϵ ' is small positive constant to avoid division by zero in implementation, ' ∇ ' is the gradient, which is taken of f , ' η ' is learning rate.

In RMSProp, the parameters ' α ', ' β ', and ' ϵ ' are set as decay $\alpha = 0.9$, momentum $\beta = 0.9$, and $\epsilon = 1.0$

F. ResNet50 Model Architecture

ResNet (Residual Network) is presented by Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun in 2015 in their paper "Deep Residual Learning for Image Recognition". The development of ResNet improved the problem of training deep neural networks. The simple element in ResNet is as depicted in Fig. 3. In the Residual network, there is a straight connection called 'skip connection' which skips some in between layers.

The 'skip connection' is used to resolve the vanishing gradient problem and to learn the identity functions. The output $H(X)$ with the introduction of skip connection is given by the equation $H(X) = F(X) + X$. Table III shows the elements of the ResNet50 model. The ResNet model was tested on the ImageNet set and attained a 20.47% top-1 error rate also 5.25% top-5 error rate.

The proposed model used these transfer learning techniques for feature extraction and altered their basic structures by adding Random Forest or SVM classifiers to improve the classification ability of the models as illustrated in Fig. 4.

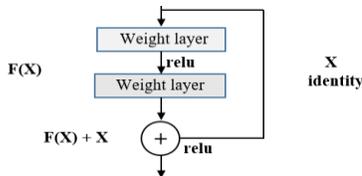


Fig. 3. A Basic Residual Network.

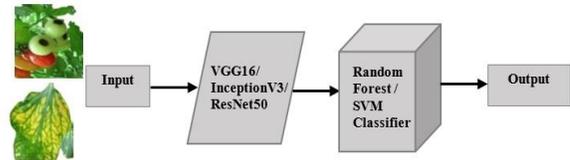


Fig. 4. Proposed Model.

III. EXPERIMENTAL RESULTS AND ANALYSIS

In this work, image pre-processing techniques, data augmentation, and implementation of Convolutional Neural Network algorithms were conducted using Jupyter notebook(Python 3.9), Keras API, OpenCV library, Matplotlib visualization library, OS module, glob module, and so on. The hardware specifications in this experiment to train and test our model includes Intel(R) Core (TM) i7-4210U CPU, 4.00 GB RAM. In this experiment, the CNN is developed using InceptionV3, ResNet50, and VGG16 Transfer Learning Models.

All three models used pre-trained weights from the ImageNet dataset by eliminating the upper layer and redefining a fresh fully connected Softmax layer with 2 classes for classification [7, 8]. In this experiment, the batch size was fixed to 8 and the number of epochs was set to 10 with Adam optimizer. The features extracted from the Transfer Learning technique were used by SVM and Random-Forest classifiers. 80% of the total images were used to form a training dataset to train the model and 20% were used to form a testing dataset. For the Inception-V3 model, all the images were resized to 299x299x3, the input image size for ResNet-50 and VGG16 was 224x224. Inception V3 attained the validation accuracy of 99.99 % and the validation loss of 0.0000384 as depicted in Table II out of the three models.

The accuracy and loss obtained from three different Transfer Learning models are presented in Fig. 5 to 7.

TABLE II. ACCURACY AND LOSS OF DIFFERENT TRANSFER LEARNING MODELS AFTER 5 AND 10 EPOCHS

Transfer Learning Model	Training Accuracy (%)		Validation Accuracy (%)		Training Loss		Validation Loss	
	5 Epochs	10 Epochs	5 Epochs	10 Epochs	5 Epochs	10 Epochs	5 Epochs	10 Epochs
Inception V3	99.23	100	98	99.99	0.1364	0.000264	0.3685	0.0000384
ResNet-50	89.40	82.28	87.00	82.10	0.2632	0.5541	0.2568	0.4783
VGG16	100	100	98.86	98.86	0.000001	0	0.3572	0.2220

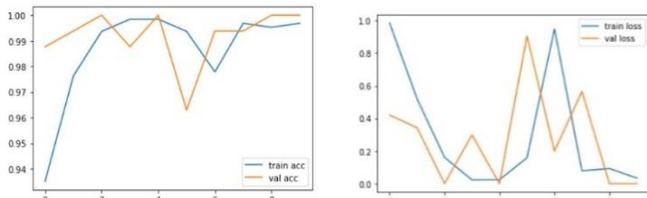


Fig. 5. Inception V3: Accuracy and Loss Model.

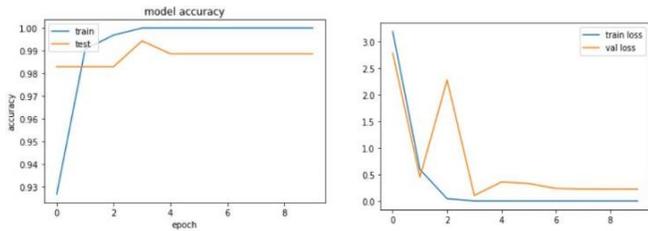


Fig. 6. ResNet50: Accuracy and Loss Model.

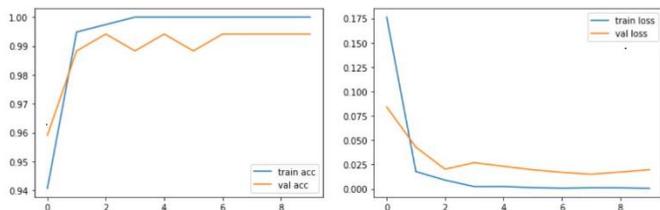


Fig. 7. VGG16: Accuracy and Loss Model.

Fig. 8 to 10 represents the confusion matrix of plant nutrient deficiencies identification using three Transfer Learning models with RF and SVM classifiers. The classification accuracy chart of plant nutrient deficiency identification in tomato plants using Transfer Learning models with Random Forest and SVM classifiers is presented in Fig. 11. It is noticed from the chart that the largest accuracy of 99.14% has resulted using the VGG16 model with SVM classifier and 98.71% for Inception-V3 with Random Forest Classifier.

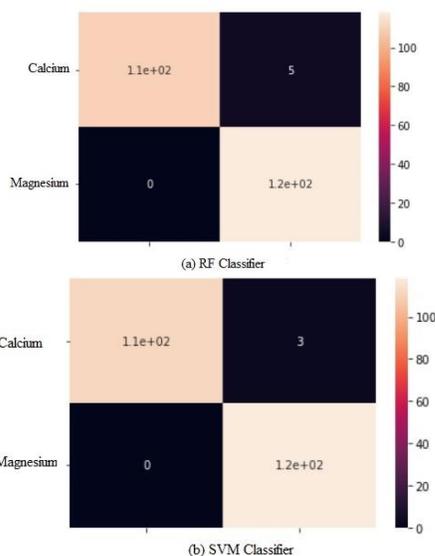


Fig. 8. Confusion Matrix using Inception V3.

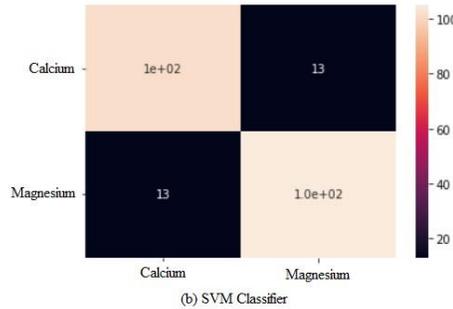
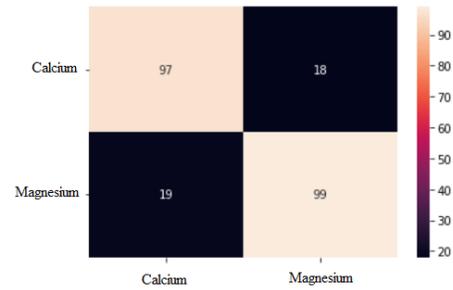


Fig. 9. Confusion Matrix using ResNet50.

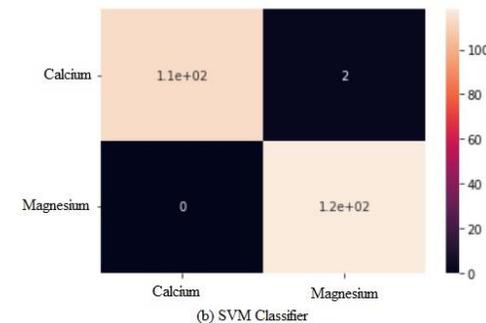
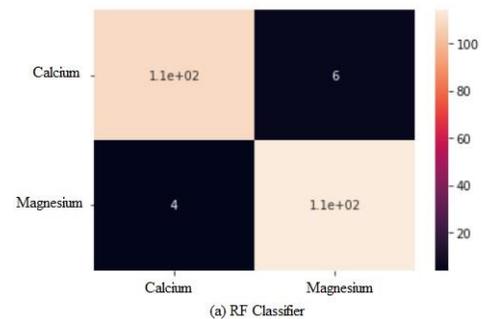


Fig. 10. Confusion Matrix using VGG16.

The results of lack of nutrients predicted from three different models with RF and SVM classifiers on few samples are displayed in Fig. 12. From Table II, it can be observed that almost calcium and magnesium deficiencies were detected properly by all three Transfer Learning models with RF and SVM classifiers. The average classification accuracy is high for InceptionV3 and VGG16 models in various experiments. These models could be extended for the identification of other nutrient deficiencies.

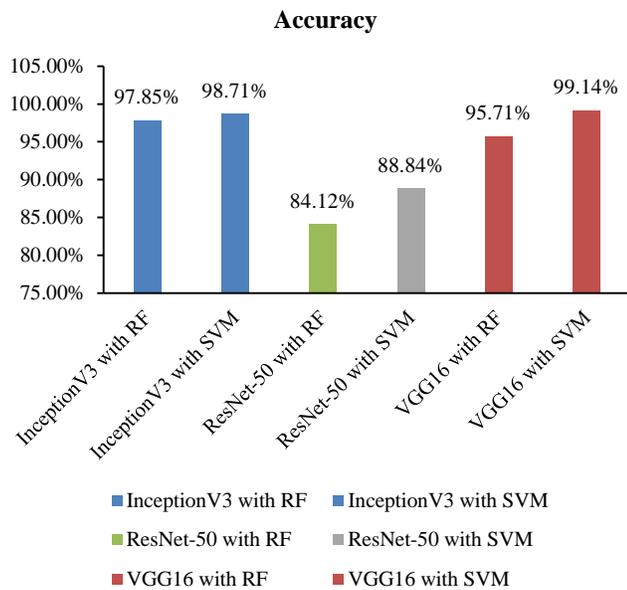


Fig. 11. Comparison Efficiency for Transfer Learning Models with RF and SVM Classifiers.

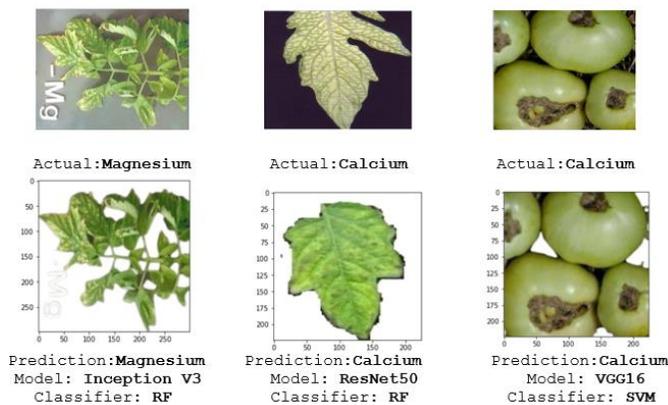


Fig. 12. The Predicted Samples of Plant Nutrient Deficiency Images.

IV. CONCLUSION

Quick identification of plant nutrient deficiency is necessary for a greenhouse environment. Manual inspection of these deficiency symptoms in a large greenhouse requires more effort. Consequently, automated plant nutrient deficiency diagnosis is required in greenhouse technology. With technology growth, a CNN using Transfer Learning models such as Inception V3, ResNet50, and VGG16 were proposed along with Random Forest (RF) and SVM classifiers to improve the efficiency. These models are pre-trained on ImageNet dataset and are modified for our tomato dataset with images of calcium and magnesium deficiencies in this research. On average, out of all the three Transfer Learning techniques, Inception V3 attained the highest validation accuracy of 99.99 % and the least validation loss of 0.0000384 for 10 epochs. Further, when the experiment was conducted for Random Forest (RF) and SVM classifiers, results show that the largest accuracy of 99.14% has resulted using the VGG16 model with

SVM classifier and 98.71% for InceptionV3 with Random Forest Classifier.

To control these plant nutrient deficiencies, the tomato greenhouse environmental factors such as humidity, temperature, pH, and soil moisture need to be monitored to find out the right quantity of fertilizer to be applied. Hence, in the future, this work can be improved by monitoring the greenhouse parameters by Wireless Sensor Network (WSN) to apply fertilizer precisely in a greenhouse.

ACKNOWLEDGMENT

We extend our heartfelt gratitude to the management of the BNM Institute of Technology, Bengaluru for providing us with all the necessary sources to accomplish this work and all the support to do the subsequent publications. We are also grateful to Visvesvaraya Technological University for giving us an appropriate platform to complete this research.

REFERENCES

- [1] Gaganjot Kaur, "Automated Nutrient Deficiency Detection In Plants: A Review", *Palarch's Journal of Archaeology Of Egypt/Egyptology*, vol. 17, no. 6, pp. 5894-5901.
- [2] G. Madhulatha, O. Ramadevi, "Recognition of Plant Diseases using Convolutional Neural Network", *International Conference on IoT in Social, Mobile, Analytics and Cloud (I-SMAC)*, 2020, ISBN: 978-1-7281-5464-0, DOI: 10.1109/I-SMAC49090.2020.9243422.
- [3] Muhammad Hammad Saleem, Sapna Khanchi, Johan Potgieter and Khalid Mahmood Arif, "Image-Based Plant Disease Identification by Deep Learning Meta-Architectures", *Plants* 2020,9,1451, MDPI Publication, DOI: 10.3390/plants9111451.
- [4] Guan Wang, Yu Sun, and Jianxin Wang, "Automatic Image-Based Plant Disease Severity Estimation Using Deep Learning", *Computational Intelligence and Neuroscience*, Volume 2017, Article ID 2917536, <https://doi.org/10.1155/2017/2917536>.
- [5] Sharada P. Mohanty, David P. Hughes, and Marcel Salathe, "Using Deep Learning for Image Based Plant Disease Detection", *Frontiers in Plant Science*, 7, 1419, 2016.
- [6] M. Nagaraju, Priyanka Chawla, "Systematic review of deep learning techniques in plant disease detection", *Int J Syst Assur Engg Manag.* 11(3), 547-560, 2020.
- [7] Nafees Akhter Farooqui and Ritika, "An Identification and Detection Process for Leaves Disease of Wheat Using Advance Machine Learning Techniques", *Bioscience Biotech Research Communication*, vol. 12, no.4, pp. 1081-1091, 2019, DOI: 10.21786/bbrc/12.4/31.
- [8] Zhe Xu, Xi Guo, Anfan Zhu, Xiaolin He, Xiaomin Zhao, Yi Han, and Roshan Subedi, "Using Deep Convolutional Neural Networks for Image- Based Diagnosis of Nutrient Deficiencies in Rice", *Computational Intelligence and Neuroscience*, Hindawi Publications, vol. 2020, 12 pages, DOI: 10.1155/2020/7307252.
- [9] Shima Ramesh, Mr. Ramachandra Hebbar, Niveditha M, Pooja R, Prasad N, Shashank N, Mr. P V Vinod, "Plant Disease Detection Using Machine Learning", *International Conference on Design Innovations for 3Cs Compute*, 2018, ISBN: 978-1-5386-7523-6, DOI: 10.1109/ICD13C 2018.00017.
- [10] Jayme Garcia Anal Barbedo, "Detection of nutrition deficiencies in plants using proximal images and machine learning: A review", *Computers and Electronics in Agriculture*, Elsevier Publications, vol. 162, pp. 482-492, 2019, DOI: <https://doi.org/10.1016/j.compag.2019.04.035>.
- [11] Aravind Krishnaswamy Rangarajan, Raja Purushothaman, Anirudh Ramesh, "Tomato crop disease classification using pre-trained deep learning algorithms", *International Conference on Robotics and Smart Manufacturing (RoSMa2018)*, 2018, pp. 1040-1047, DOI: 10.1016/j.procs.2018.07.070.
- [12] Yan Guo, Jin Zhang, Chengxin Yin, Xiaonan Hu, Yo Zou, Zhipeng Xue, Wei Wang, "Plant Disease Identification Based on Deep Learning

- Algorithm in Smart Farming”, *Discrete Dynamics in Nature and Society*, vol.2020, no.7, pp.1-11, 2020. DOI: 10.1155/2020/2479172.
- [13] Lili Ayu Wulandhari, Alexander Agung Santoso Gunawan, Aie Qurania, Prihastuti Harsani, Triastinurmiatingsih, Ferdy Tarawan, and Riska Fauzia Hermawan, “Plant Nutrient Deficiency Detection Using Deep Convolutional Neural Network”, *ICIC International Conference, ICIC Express Letters*, pp. 971-977, vol. 13, no. 10, 2019.
- [14] G.Chu, “How to use transfer learning and fine-tuning in keras and tensorflow to build an image recognition system and classify (almost) any object”, *Deep Learning Sandbox*, 2017.
- [15] A. M. G. J. Hanson, A. Joy, and J. Francis, “Plant leaf disease detection using deep learning and convolutional neural network”, *International Journal of Engineering Science*, vol. 7, no.3, pp.5324-5328, 2017.
- [16] G.L. Grinblat, L.C. Uzal, M.G. Larese and P. M. Granitto “Deep learning for plant identification using vein morphological patterns”, *Computers and Electronics in Agriculture*, vol.127, pp. 418-424, 2016.
- [17] K. P. Ferentinos, “Deep learning models for plant disease detection and diagnosis”, *Computers and Electronics in Agriculture*, vol. 145, pp.311-318, 2018.
- [18] D. Story, M. Kacira, C. Kubota, A. Akoglu and L. An, Lettuce calcium deficiency detection with machine vision computed plant features in controlled environments, *Computers and Electronics in Agriculture*, vol. 74, no.2, pp. 238-243, 2010.