

# Disorder Severity Classification in Tomato Based on Weight Cluster Loss and Convolutional Neural Network

Babatunde Sunday

Department of Computer Science  
Faculty of Physical Science  
Ahmadu Bello University,  
Zaria, Nigeria  
sundaybabatunde53@gmail.com

Yazeed Masha Abdullahi

Department of Computer Science  
Faculty of Physical Science  
Ahmadu Bello University,  
Zaria, Nigeria  
yazv4real@gmail.com

Alice Sikemi Matemilola

Department of Computer Science  
Faculty of Physical Science  
Ahmadu Bello University,  
Zaria, Nigeria  
matemilolao@gmail.com

Yusuf Ali Sahabi

Department of Computer Science  
Faculty of Physical Science  
Ahmadu Bello University,  
Zaria, Nigeria  
sahabiali@yahoo.com

Mohammed Abdullahi

Department of Computer Science  
Faculty of Physical Science  
Ahmadu Bello University,  
Zaria, Nigeria

**Abstract** -Agricultural productivity is something on which the Economy highly depends. Plant Diseases and Pests are a major provocation in the agricultural sector. This is one of the reasons that disorder severity detection in tomato plants plays an important role. Accurate and faster detection of disorder severity in tomato plants could help to develop an early treatment technique while substantially reducing economic losses. Several researchers have come up with solutions using varieties of CNN models to accurately identify and classify disease severity in tomato, yet, there is less amount of training data which leads to overfitting problem and inter-class similarity which results into comparatively low performance measures for tomato disease severity classification. This research proposes a deep transfer-based framework to address overfitting problem caused by insufficient amount of data, in which three CNN models namely, AlexNet, SqueezeNet and InceptionV3 were utilized for classifying disease severity in tomato late blight, tomato early blight, tomato leaf mold, tomato bacteria spot and healthy tomato leaves from PlantVillage dataset. The research also adopts a loss function named weighted-cluster loss to address inter-class similarities. The proposed methodology uses transfer learning approach by employing three CNN architectures in two-stages, supervised pre-training and fine-tuning. The performance metrics was computational accuracy. After running the experiments, InceptionV3 attained the highest classification accuracy of 93.66% as compared with AlexNet which attained 83.03%, and SqueezeNet with 80.09%. As a result, the proposed system will serve as a decision support tool to farmers in identifying disorder severity that occur in tomato plant leaf.

**Keywords:** CNN, Inter-Class Similarities, Overfitting, Weight Cluster Loss,

## I. INTRODUCTION

Plants plays very important role in the life of human. Crop production, fishing, livestock and forestry are the four main agricultural sectors in Nigeria. Crop production continues to be the most significant segment, accounting for about 87.6%

of the sector's total output. Livestock, fishing, and forestry come in second, with 8.1%, 3.2% and 1.1%, respectively. Smart farming plays an important role in addressing the challenges of agricultural production in terms of food security, environmental effects and sustainability. It is important to understand agricultural conditions by observing, measuring, and evaluating different physical factors and phenomena to overcome these challenges.

According to Food and Agricultural Organization of United Nations, Statistics (FAOSTAT), in 2017, global tomato production was reported to be about 170.8 million tons. Tomatoes are vulnerable to diseases and pests that can be found in the farm field.

There have been numerous efforts made using Artificial Intelligence (AI) to help small farmers to accurately identify and classify the pests and diseases that affect their production and the severity of the symptoms of each disease. According to [2] Computer-Aided Diagnosis (CAD) systems allow any farmer with accessibility to a computer or smartphone to enjoy expert knowledge in a practical and low-cost way.

According to [8] stated that based on many research, it has been revealed that in the area of Artificial Intelligence (AI) there have been notable growth in the few years concerning the applications of Machine Learning, which led to the emergence of a new category of models called Deep Learning. [1] stated that Deep Learning methods have shown state-of-the-art performance when compared to traditional Machine Learning approaches in different domains such as agriculture, education, medicine and many more.

According to [7] A deep learning-based disease severity system usually uses Convolutional Neural Networks (CNN) model to extract and learn high-level features directly from input images. Then, an output layer (which usually uses softmax as an activation function) is attached on top of the CNN model to differentiate the diseases severity to be

identified. [14] stated that the use of CNN model allows a faster system with higher accuracy in challenging, real-world environments. In this research, we will make use of the powerful feature-learning capacity of the deep CNN to build our models.

## II. LITERATURE REVIEW

Few studies have been conducted on the use of Convolutional Neural Network models for the classification of plant diseases severity such as [4], [3], [2], [11], [15], [9], [6], [10], [12], and [5]. [4] proposes a deep learning approach to automatically discover the discriminative features for fine-grained classification, which enables the end-to-end pipeline for diagnosing plant disease severity with a test accuracy of 79.3%. The findings of their study reveals that fine tuning on pre-trained deep models can significantly improve the performance accuracy of the model on few data.

[3] presented different types of meta architectures namely single shot multibox detector (SSD), faster region-based CNN (Faster R-CNN), and region-based fully CNN (R-FCN) to identify the disease-affected parts table in leaf images. These architectures are utilized by CNN models such as VGG16 and ResNet. The experimental result shows that ResNet with R-FCN outperforms with an accuracy of 85.98%.

[2] employed SqueezeNet and AlexNet to detect the tomato disease by means of leaf images acquired from the plantvillage database. The highest accuracy of 95.65% is obtained by AlexNet whereas SqueezeNet produces 94.30%.

[11] attempted the pretrained deep learning model adopting transfer learning (TL) concepts namely AlexNet and VGG16 to extract the features from tomato images for classification of healthy and disease-affected classes. This work produces a classification accuracy of 97.49% and 97.23% for AlexNet and VGG16, respectively.

[15] employed TL-based CNN model to predict the tomato plant disease through leaves images. The AlexNet, ResNet, and GoogLeNet are the pre-trained models used which utilize stochastic gradient descent (SGD) and adaptive moment estimation (Adam) optimizer for classifying the tomato diseases automatically. The ResNet model with SGD optimizer produces greater accuracy of 97.28% compared with other pre-trained models such as GoogLeNet and AlexNet.

[9] implemented a new deep learning architecture namely PD2SENet for evaluating the plant disease severity and classification, as well as plant species identification, with ResNet50 as the base model, achieving the accuracies of 91%, 98% and 99% respectively. The findings of their study revealed that data augmentation and visualization of CNNs were beneficial to improve the performance of the model.

[6] proposed models that has the capability to identify and estimate the stress severity that is caused by the biotic agents on coffee leaves. The findings of their research shows that the emergence of multiple stresses on a single leaf can significantly alter the characteristics of symptoms, especially when they overlap. These symptom overlaps are quite rare to capture and would likely cause the system to make misclassifications.

[10] proposed a method which focused on recognizing generic features that are associated with diseases regardless of disease type which enables detection of new instances that might not have been seen before. The findings of their study revealed that spider mite disease and target spot disease are closest in appearance to those of healthy leaves. In these two cases, the models showed no ability to discriminate between the diseased and healthy leaves in which the model nearly always predicts the healthy class.

[12] proposed an application of Convolutional Neural Networks frameworks which are AlexNet, SqueezeNet and Inception V3 for evaluation of disease severity in tomato plant considering only late blight disease with an accuracy of 89.69%. The findings of their study revealed that working with smaller datasets, fine-tuning the models yields better results when compared to transfer learning, the feature extraction strategy yielded better outcomes with less effort. AlexNet outperformed the other two networks in terms of fine-grained picture classification. The overall performance of the networks did not improve or alter much when the train-test divides were changed.

[5] implemented Convolutional Neural Networks frameworks for Severity Levels of Potato Blight Disease. The CNN implementation was performed on 6 classes, out of these 5 classes are disease severity levels and one class belongs to healthy potatoes. The total number of images is 9600 and the accuracy achieved is 96.29% and the macro average is 86.625%.

## III. METHODOLOGY

The specifications that was used for the design and implementation of this work are described in Table 1, below.

TABLE 1: SYSTEM SPECIFICATIONS

System Specifications	
<b>Language</b>	Python
<b>Platform</b>	Google Colab
<b>Processor</b>	Intel(R) Core(TM) i5-3360M CPU @ 2.80GHz 2.80 GHz
<b>RAM</b>	8.0 GB

## Methods

In this section, we first describe our deep transfer-learning framework and then propose weighted-cluster loss. The framework of the proposed method is shown in figure 1.

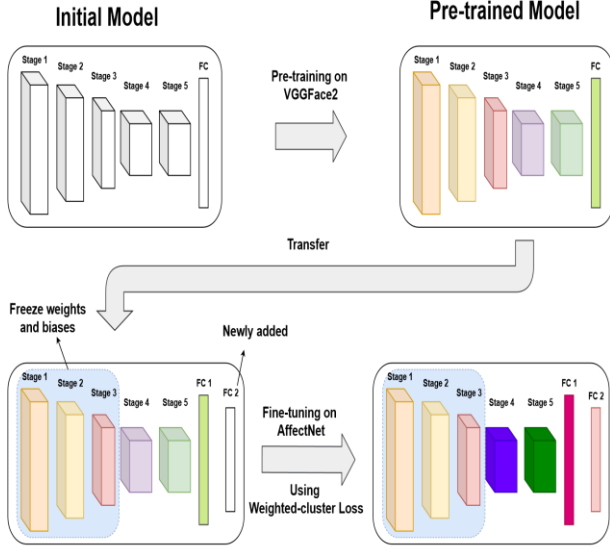


Figure 1: Framework of the proposed method.

AlexNet, SqueezeNet and InceptionV3 model, which was pre-trained on PlantVillage data for disease identification, is fine-tuned for disease recognition using weighted-cluster loss. Before the fine-tuning phase, we add one more fully connected layer to the model while froze the three first stages of the pre-trained model to save computing power. The weighted-cluster loss is used at the output layer to update model parameters.

### A. Base Model and Pre-Training Base Models

Convolutional neural networks have achieved great success in the fields of pattern recognition and computer vision. This motivated us to base our AlexNet, SqueezeNet and InceptionV3 models on a recent representative CNN architecture in the computer vision field. The research employed the AlexNet, SqueezeNet and InceptionV3 model as our base model. A wide range of experiments showed the effectiveness of AlexNet, SqueezeNet and InceptionV3 that achieve state-of-the-art performance across multiple datasets and tasks. This was demonstrated for object and scene classification, with a squeeze-and-excitation network winning the ILSVRC 2017 competition.

### B. Pre-Training

This work uses publicly available pre-trained model of the base model (i.e., transfer learning-based model) instead of

directly pre-training it. Specifically, the AlexNet, SqueezeNet and InceptionV3 pre-trained model, which is published on the official website of the PlantVillage database, was trained with the PlantVillage dataset for the task of recognizing six thousand two hundred and sixty-four (6264) diseases. As a results, the network learned rich feature represent for disease factors.

### C. Preprocessing Mechanisms and Preparation Methods

#### 1. Fine-Tuning

In this stage, the research add one more fully connected layer to the pre-trained AlexNet, SqueezeNet and InceptionV3 model (before the output layer) and we then changed the output layer to eight-class classification. In addition, we froze the weights and biases of Stage 1 to Stage 3 on the pre-trained model. This process is usually found in many transfer learning-based systems when computing power is limited.

#### 2. Data Collection

For this research, data was extracted from PlantVillage for experiments as illustrated in table 3. The choice of PlantVillage is a result of the simplicity it offers.

TABLE 2: SUMMARY OF THE CLASSES OF TOMATO PLANT DISORDER AND IMAGES FOR EXPERIMENT.

Database	Class	Count of Images
PlantVillage	Healthy	1591
	Early blight	1000
	Late bright mold	1910
	Leaf mold	952
	Bacterial spot	811

#### 3. Weighted-Cluster Loss

To address the limitations of existing loss functions (e.g., weighted-softmax loss and center loss), this research adopts a new loss function called weighted-cluster loss. Furthermore, weighted-cluster loss adds a new term to center loss, which simultaneously pulls the centers of each class apart. This may allow the model to simultaneously handle the high intra-variation and the inter-class similarity in the PlantVillage dataset. The weighted-cluster loss for the  $i$ th training sample is given by Equation 3.1:

$$L_{\text{Weighted-cluster}} = 1/2 \sum_{i=1} W_{y_i} \frac{\|x_i - c_{y_i}\|_2^2}{\left(\sum_{j=1, j \neq i}^k \|c_j - c_{y_i}\|_2^2\right) + \alpha} \quad (3.1)$$

where  $W_{y_i}$  denotes the weight assigned for the class where the label is  $y_i$ ,  $k$  denotes the number of disease classes, and the constant,  $\alpha$ , prevents the denominator from equaling 0. In this paper, we set  $\alpha = 1$  by default.

In Equation (3.2), the numerator penalizes the distance between the deep feature of the training sample (i.e., taken

from the last fully connected layer before the output layer) and its corresponding center, and the denominator penalizes the distance between the corresponding center and all other class centers. By minimizing weighted-cluster loss, the deep features of training samples from the same class (i.e., the cluster) was compacted in the feature space, whereas the distance between different classes of clusters was enlarged. In addition, weight  $W_{yi}$  is used to tackle the imbalanced dataset problem by penalizing the models less for misclassifying samples from majority classes while heavily penalizing the model for misclassifying samples from minority classes.

The overall loss function of PlantVillage training is given by:

$$L = L_{\text{Weighted-softmax}} + \lambda L_{\text{Weighted-cluster}} \quad (3.2)$$

$$= - \sum W_{yi} (\log(\hat{p}_i) - \lambda \frac{\| \mathbf{x}_i - \mathbf{c}_i \|^2_2}{2 \sum_{j=1, j \neq i}^k \| \mathbf{c}_j - \mathbf{c}_i \|^2_2 + \alpha})$$

where  $\lambda$  is used to balance the two losses. In this work,  $W_{yi}$  is define as:

$$W_{yi} = \frac{N_{\min}}{N_{yi}} \quad (3.3)$$

where  $N_{\min}$  denotes the number of samples from the smallest class (i.e., disgust), and  $N_{yi}$  denotes the number of samples from the class where the label is  $y_i$ .

The joint weighted-cluster loss and weighted-softmax loss can be directly used for training deep neural networks. The class centers,  $\mathbf{c}_i$ , are updated each iteration through the training process. Specifically, the class centers are updated based on mini-batches, and use a scalar  $\gamma$  to control the learning rate of class centers (and in our experiments, we set  $\gamma = 1$ ). The partial derivative of the weighted cluster loss,  $L_{\text{Weighted-cluster}}$ , with respect to the sample's feature,  $\mathbf{x}_i$ , can be calculated as follows:

$$\frac{\partial L_{\text{Weighted-cluster}}}{\partial \mathbf{x}_i} = W_{yi} \frac{\mathbf{x}_i - \mathbf{c}_i}{\left( \sum_{l=1, l \neq i}^k \| \mathbf{c}_l - \mathbf{c}_i \|^2_2 \right) + \alpha} \quad (3.4)$$

The update of the  $j$ th class center can be calculated with Equation (8):

$$\Delta \mathbf{c}_j = \sum_{i=1} W_{yi} \frac{\delta(y_i = j)(\mathbf{x}_i - \mathbf{c}_j)}{\left( \sum_{l=1, l \neq i}^k \| \mathbf{c}_l - \mathbf{c}_j \|^2_2 \right) + \alpha} \quad (3.5)$$

where  $\delta(y_i = j) = 1$  if  $y_i = j$ , and  $\delta(y_i = j) = 0$  if  $y_i \neq j$ . Then, the class centers can be updated in each mini-batch with a learning rate  $\gamma$ :

$$\mathbf{c}_{t+1} = \mathbf{c}_t - \gamma \Delta \mathbf{c}_t \quad (3.6)$$

In Algorithm 3.1, we summarize the learning process of the PlantVillage model with the weight cluster loss functions.

---

**Algorithm 3.1** Learning algorithm of the PlantVillage model with the weight cluster loss functions

---

**Input:** Training data  $\{\mathbf{x}_i\}$ , mini-batch size  $m$ , number of iterations  $T$ , learning rates  $\mu$  and  $\gamma$ , and hyper-parameters  $\lambda$

- 1: **Initialize:** Network layer parameters  $W$ , weighted-softmax loss parameters  $\theta$ , and weighted-cluster loss parameters (i.e., centers)  $\{\mathbf{c}_j | j = 1, 2, \dots, n\}$ .
  - 2: **for**  $t = 1$  to  $T$  **do**
  - 3: Calculate the joint loss
  - 4:  $L_t = L_{\text{Weighted-softmax}} + \lambda L_{\text{Weighted-cluster}}$
  - 5: Compute the backpropagation error for each  $i$
  - 6:  $\frac{\partial \partial L_{\mathbf{x}_i}}{\partial \theta} = \frac{\partial L_{\text{Weighted-softmax}}}{\partial \theta} + \lambda \frac{\partial L_{\text{Weighted-cluster}}}{\partial \theta}$
  - 7: Update the weighted-softmax loss parameters  $\theta$   
 $\theta_{t+1} = \theta_t - \mu \frac{\partial L_{\text{Weighted-softmax}}}{\partial \theta}$
  - 8:  $\theta = \theta_{t+1}$
  - 9: Update the weighted-cluster loss parameters  $\mathbf{c}_j$  for each  $j$  as Equation (9)
  - 10:  $\mathbf{c}_{t+1} = \mathbf{c}_t - \gamma \Delta \mathbf{c}_t$
  - 11: Update the network parameters  $W$   $\frac{\partial \partial L_{\mathbf{x}_i}}{\partial W}$
  - 12:  $W_{t+1} = W_t - \mu \frac{\partial \partial L_{\mathbf{x}_i}}{\partial W}$
  - 13: **end for**
- Output:** Network layer parameters  $W$

Evaluation Metrics

Evaluation metrics quantify the performance of a predictive model [13]. A widely used case is the model classification accuracy, given by equation 3.7,

$$\text{Accuracy} = \frac{\text{Correct Predictions}}{\text{Total Predictions}} \quad (3.7)$$

Mathematically, the accuracy of the model is given by equation 3.2,

$$\text{Accuracy of a model} = \frac{(TP + TN)}{(TP + FN + FP + TN)} \quad (3.8)$$

## IV. RESULT AND DISCUSSIONS

### A. Classification Analysis

Table 2 presents the classification accuracy of all three CNN models via transfer learning respectively. The early (Stage 1), middle (Stage 2) and end (Stage 3) stages were mostly correctly classified. As can be observed, InceptionV3 gives better results on the test set as compared to other two architectures, proving to be more suitable for fine-grained classification, as the input images inter-image similarities was handled by the weighted cluster loss. Using transfer

learning approach, AlexNet achieves 83.03% accuracy, SqueezeNet achieves 80.19% accuracy while InceptionV3 achieves 93.66% accuracy.

TABLE 3: COMPARISON BASED ON ACCURACY USING TRANSFER LEARNING

Dataset: <b>PlantVillage</b>				
Base Model	Split Ratio	Inception V3	AlexNet	SqueezeNet
		<b>80:20</b>	82.87%	<b>89.69%</b>
	<b>70:30</b>	79.61%	<b>88.45%</b>	87.15%
Proposed Model	80:20	<b>93.66%</b>	83.03%	80.19%
	<b>70:30</b>	<b>91.87%</b>	82.11%	80.11%

From the Table 3, the base model shows that AlexNet gives better results on the test set as compared to the proposed model which shows that InceptionV3 is better.

1) *Comparison of the average accuracy of the three models*

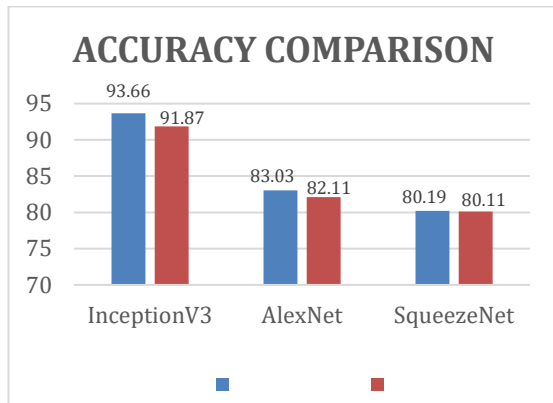


Figure 2: Comparison of accuracy of the three proposed models

2) *Performance Comparison with Benchmark Datasets of [12]*

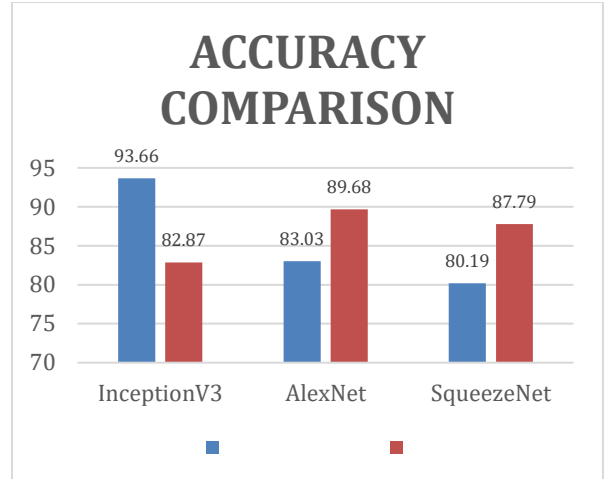


Figure 3: Comparison of the proposed framework with existing framework of [12]

3) *Performance Evaluation on Benchmark datasets.*

Finally, we test our proposed approach on the benchmark datasets from PlantVillage. Table 4 describes the performance metrics of the proposed disorder severity classification models.

TABLE 4: PERFORMANCE EVALUATION METRICS OF THE PROPOSED MODELS ON BENCHMARK DATASET

Dataset	Models	Accuracy
PlantVillage	InceptionV3	93.66%
	AlexNet	83.03%
	SqueezeNet	80.09%

TABLE 5: CONFUSION MATRIX FOR HEALTHY AND UNHEALTHY TOMATO LEAF IMAGES BASED ON ALEXNET AND WEIGHT CLUSTER LOSS

	Tomato_Bacterial_spot	Tomato_Early_blight	Tomato_Late_blight	Tomato_Leaf_Mold	Tomato_healthy
Tomato_Bacterial_spot	<b>0.88</b>	0.00	0.10	0.00	0.02
Tomato_Early_blight	0.02	<b>0.91</b>	0.00	0.07	0.00
Tomato_Late_blight	0.06	0.00	<b>0.89</b>	0.00	0.05
Tomato_Leaf_Mold	0.03	0.07	0.07	<b>0.83</b>	0.00
Tomato_healthy	0.06	0.05	0.00	0.03	<b>0.86</b>

The confusion matrix for healthy and unhealthy tomato leaf images based on AlexNet and Weight Cluster Loss is shown in Table 5. It can be noticed that the five-class problem, which consisted of one healthy class and four different unhealthy classes, 0.86 is correctly predicted by the model as healthy while only 0.14 healthy tomato leaf images were misclassified as unhealthy images.

TABLE 6: CONFUSION MATRIX FOR HEALTHY AND UNHEALTHY TOMATO LEAF IMAGES BASED ON SQUEEZENET AND WEIGHT CLUSTER LOSS

		Predicted class				
		Tomato_Bacterial_spot	Tomato_Early_blight	Tomato_Late_blight	Tomato_Leaf_Mold	Tomato_healthy
True class	Tomato_Bacterial_spot	<b>0.89</b>	0.00	0.06	0.00	0.05
	Tomato_Early_blight	0.00	<b>0.92</b>	0.04	0.04	0.00
	Tomato_Late_blight	0.02	0.06	<b>0.90</b>	0.00	0.02
	Tomato_Leaf_Mold	0.03	0.00	0.07	<b>0.90</b>	0.00
	Tomato_healthy	0.00	0.06	0.01	0.02	<b>0.91</b>

The confusion matrix for healthy and unhealthy tomato leaf images based on SqueezeNet and Weight Cluster Loss is shown in Table 6. It can be noticed that the five-class problem, which consisted of one healthy class and four different unhealthy classes, 0.91 was predicted correctly by the model as healthy, only 0.09 healthy tomato leaf images were misclassified as unhealthy images.

TABLE 7: CONFUSION MATRIX FOR HEALTHY AND UNHEALTHY TOMATO LEAF IMAGES BASED ON INCEPTIONV3 AND WEIGHT CLUSTER LOSS.

		Predicted class				
		Tomato_Bacterial_spot	Tomato_Early_blight	Tomato_Late_blight	Tomato_Leaf_Mold	Tomato_healthy
True class	Tomato_Bacterial_spot	<b>0.99</b>	0.00	0.01	0.00	0.00
	Tomato_Early_blight	0.01	<b>0.97</b>	0.00	0.02	0.00
	Tomato_Late_blight	0.00	0.00	<b>0.10</b>	0.00	0.00
	Tomato_Leaf_Mold	0.00	0.00	0.00	<b>0.10</b>	0.00
	Tomato_healthy	0.00	0.02	0.00	0.00	<b>0.98</b>

The confusion matrix for healthy and unhealthy tomato leaf images based on InceptionV3 and Weight Cluster Loss is shown in Table 7. It can be noticed that even with the best performing network InceptionV3 for the five-class problem, which consisted of one healthy class and four different unhealthy classes, 0.98 was correctly predicted by the model as healthy while only 0.2 healthy tomato leaf images were misclassified as unhealthy images.

## V. DISCUSSIONS

From the series of experiments in the previous sections, we can observe the effect of the weight cluster loss and CNN architectures on the disorder severity classification of Tomato. In detail, we fine-tuned the transfer learning-based model (e.g., AlexNet, SqueezeNet and InceptionV3 pre-trained with PlantVillage dataset) with the proposed weighted-cluster loss. In addition, to evaluate the effectiveness of transfer learning, we also trained the base model using only the PlantVillage training dataset with the same loss settings as those used for transfer learning-based model. In all experiments, we set the  $\lambda$  value (i.e., the scalar used to balance the loss terms) to 0.5 when using center loss, and to 1.0 when using weighted-cluster loss.

The proposed model however handled the overfitting problem and interclass similarities by classifying the different classes of disorders ranging from tomato late blight, tomato early blight, tomato leaf mold, tomato bacteria spot and healthy tomato leaves estimating the percentage of infected part and the severity stage of the tomato plant accurately.

## VI. CONCLUSION

In this work, we carried out several experiments using benchmark classification datasets to examine the impact of our proposed model. In this work, we proposed an approach of weight cluster loss to improve tomato disorder severity classification through incorporating it into pre-trained CNN models namely; Alexnet, InceptionV3 and Squeezenet via transfer learning. The weight cluster loss has significantly contributed to reducing the interclass similarities.

With smaller datasets, fine-tuning the models leads to improved results, however, in comparison to the base models using transfer learning, inclusion of weighted cluster loss resulted in better performance, with minimum efforts. Amongst the three networks, InceptionV3 proved to be more suitable for the fine grained image classification. Varying the train-test splits did not result in any significant change or improvement in the overall performance of the networks.

## VII. RECOMMENDATIONS

For future work and directions, the proposed model effectiveness can further be tested on images from several resources with different loss functions to check for better results which could give a clear understanding of the actual applicability of the developed model. Experimenting with smaller datasets with different classes of plants disorder will

possibly result in better performance. The utilization of real life images, captured via digital cameras is absolutely essential. Estimating the disorder severity correctly could lead to an appropriate use of pesticides. For early prediction, hyperspectral imaging along with deep learning could prove to be a promising approach, which could severely diminish the use of pesticides on crops. Hence, a large agricultural dataset, collected from various geographic areas in real field conditions via different remote sensing devices is required for practical applications.

## REFERENCES

- [1] Alom H., M. I. Lali, M. Z. Nawaz, M. Sharif, and B. A. Saleem, (2019) "Symptom-based automated detection of citrus diseases using color histogram and textural descriptors," *Computers and Electronics in Agriculture*, vol. 138, pp. 92–104, 2017. View at: Publisher Site | Google Scholar
- [2] Durmuş H, Güneş EO, Kırıcı M (2017) Disease detection on the leaves of the tomato plants by using deep learning. In 2017 6th International Conference on Agro-Geoinformatics (pp. 1–5). IEEE.
- [3] Fuentes V., P. Henderson, R. Islam, M.G. Bellemare, J. Pineau, (2017) An Introduction to Deep Reinforcement Learning. 11(3–4). <https://doi.org/10.1561/22000000071>. Vincent.
- [4] Guan Wang, Yu Sun, and Jianxin Wang (2017) Automatic Image-Based Plant Disease Severity Estimation Using Deep Learning, *Computational Intelligence and Neuroscience*, Article ID 2917536, <https://doi.org/10.1155/2017/2917536>
- [5] Jindal, V., Nagpal Y. and Kukreja, V. (2022) "CNN Implementation for Severity Levels of Potato Blight Disease," 2022 International Conference on Data Analytics for Business and Industry (ICDABI), Sakhir, Bahrain, 2022, pp. 438–443, doi: 10.1109/ICDABI56818.2022.10041501.
- [6] Jose M., X. Bu, X. Sun, Z. Yu, and Y. Zheng, (2020) "Rape plant disease recognition method of multi-feature fusion based on D-S evidence theory," *Mathematical and Computational Applications*, vol. 22, no. 1, p. 18, 2017.
- [7] Kim, Y.-H. Lai, and J.-W. Perng, (2017) "Determination of Lycopersicon maturity using convolutional autoencoders," *Scientia Horticulturae*, vol. 256, p. 108538.
- [8] Lecun Y., Y. Bengio, (2015) Convolutional networks for images, speech, and time-series yann lecun, in *The Handbook of Brain Theory and Neural Networks*, 3361(10), 1997.
- [9] Liang D., L. Deng, M. Lee, and H. Wang, (2019) "IoT data feature extraction and intrusion detection system for smart cities based on deep migration learning," *International Journal of Information Management*, vol. 49, pp. 533–545, 2019. View at: Publisher Site | Google Scholar
- [10] Patrick Wspanialy, Medhat Moussa (2020): A detection and severity estimation system for generic diseases of tomato greenhouse plants. *Comput. Electron. Agric.* 178: 105701
- [11] Rangarajan AK, Purushothaman R, Ramesh A (2018) Tomato crop disease classification using pre-trained deep learning algorithm. *Procedia Computer Science* 133:1040–1047
- [12] Shradha Verma, Anuradha Chug & Amit Prakash Singh (2020) Application of Convolutional Neural Networks for evaluation of disease severity in tomato plant, *Journal of Discrete Mathematical Sciences and Cryptography*, 23:1, 273-282, DOI: 10.1080/09720529.2020.1721890.
- [13] Sun, Z. Yu, and Y. Zheng, "Rape plant disease recognition method of multi-feature fusion based on D-S evidence theory," *Mathematical and Computational Applications*, vol. 22, no. 1, p. 18, 2017
- [14] Walecki R., Ognjen R, Vladimir P., Björn S., Maja P. (2017) Deep Structured Learning for Facial Action Unit Intensity Estimation, *Computer Vision and Pattern Recognition (cs.CV)*
- [15] Zhang K, Wu Q, Liu A, Meng X (2018) Can deep learning identify tomato leaf disease?. *Advances in Multimedia*.