2020-11

Early identification of Tuta absoluta in tomato plants using deep learning

Mkonyi, Lilian

Elsevier B.V.

https://doi.org/10.1016/j.sciaf.2020.e00590

Scientific African 10 (2020) e00590

Contents lists available at ScienceDirect

Scientific African

journal homepage: www.elsevier.com/locate/sciaf

Early identification of *Tuta absoluta* in tomato plants using deep learning

Lilian Mkonyi[a,d,*], Denis Rubanga[b], Mgaya Richard[c], Never Zekeya[a], Shimada Sawahiko[b], Baraka Maiseli[d], Dina Machuve[a]

[a] The Nelson Mandela African Institution of Science and Technology, P.O.Box 447 Arusha, Tanzania
[b] Tokyo University of Agriculture, Setagaya-ku, Tokyo 156-8502, Japan
[c] Sokoine University of Agriculture, P.O. Box 3000, Morogoro, Tanzania
[d] University of Dar es Salaam, P.O Box 33335, Dar es Salaam, Tanzania

ARTICLE INFO

Article history:
Received 11 November 2019
Revised 16 June 2020
Accepted 6 August 2020

Keywords:
Tuta absoluta
Classification
Convolutional neural network
Transfer learning

ABSTRACT

The agricultural sector is highly challenged by plant pests and diseases. A high–yielding crop, such as tomato with high economic returns, can greatly increase the income of smallholder farmers income when its health is maintained. This work introduces an approach to strengthen phytosanitary capacity and systems to help solve tomato plant pest *Tuta absoluta* devastation at early tomato growth stages. We present a deep learning approach to identify tomato leaf miner pest (*Tuta absoluta*) invasion. The Convolutional Neural Network architectures (VGG16, VGG19, and ResNet50) were used in training classifiers on tomato image dataset captured from the field containing healthy and infested tomato leaves. We evaluated performance of each classifier by considering accuracy of classifying the tomato canopy into correct category. Experimental results show that VGG16 attained the highest accuracy of 91.9% in classifying tomato plant leaves into correct categories. Our model may be used to establish methods for early detection of *Tuta absoluta* pest invasion at early tomato growth stages, hence assisting farmers overcome yield losses.

1. Introduction

Tomato *(Lycopersicon esculentum)* is a nutrition-rich and an edible plant that is widely grown throughout the world [1]. Globally, approximately 160 million tons of tomato are produced each year [2]. In 2016, more than 247,135 tons of tomatoes were harvested in Tanzania within an area of 54,520 hectare. This production is equivalent to 64% of all fruits and vegetables in the country [3]. Tomato is considered to be a source of income to small-scale farmers, and, therefore, the plant contributes largely to poverty reduction. Given the economic importance of tomato, we should consider the factors affecting its production and find more appropriate technological solutions to maximize its productivity.

The production of tomato is threatened by an invasive pest called tomato leaf miner scientifically known as *Tuta absoluta* (Meyrick) (Lepidoptera:Gelechiidae) (hereafter referred to as (*T. absoluta*)), which tends to attack the plant and weaken its

* Corresponding author.
E-mail addresses: mkony.lily@gmail.com, mkonyil@nm-aist.ac.tz (L. Mkonyi).

https://doi.org/10.1016/j.sciaf.2020.e00590

growth and yield capacity. Tomato leaf miner was originated from South America and later spread to the rest of the world. The pest has reproductive rate of around 12 generations per year. The mature female can lay between 250 and 300 eggs at once, and has a life cycle with four development stages: egg, larva, pupa, and adult.st has reproductive rate of around 12 generations per year. The mature female can lay between 250 and 300 eggs at once, and has a life cycle with four development stages: egg, larva, pupa, and adult. The second stage (larva) is the most dangerous one because the pest at this stage can mine, develop, and feed on leaves, stems, and fruits of the tomato plant [4]. Therefore, if the larva is left uncontrolled at the early stages of its growth, it may consume all plants in the farm [5]. The management of tomato leaf miner has continued to be a great constraint in the industry of tomato production, hence calling for scholars to devise approaches of identifying and combating it before causing great losses to farmers. Recent statistics show that farmers of tomato, the main host for the pest, have continued to incur heavy yield loss, ranging from 80% to 100%, due to the invasion [6].

Of the available approaches to address the issue, deep learning has demonstrated successful results [7]. Various deep learning techniques have been applied to identify, classify, and quantify diseases, pests, and stress on different crops. Among these techniques, Convolutional Neural Network (CNN) [8] provides sophisticated ways of image analysis, and thus facilitates diagnosis of plant diseases. We need to apply these advanced techniques to develop more effective approaches for identifying early invasion of *T. absoluta* in tomato.

Invasion of *T. absoluta* has, for years, been causing great production and economic loses in the world. And, to date, no suitable solution is available to control its spread. Despite existence of various ways of controlling the pest (using chemical pesticides and pheromone traps, and cultivation of resistant tomato varieties), early identification of the pest remains an open-ended research question [5,9]. In Tanzania, for instance, the agricultural system depends on extension officers as key facilitators in providing farmers with appropriate knowledge on pest and disease management. However, the extension service system is currently conducted locally by limited extension officers' visits to provide training and workshop to meet demands of all farmers in the given area [10]. This challenge calls for a need to integrate sophisticated technologies, including those based on deep learning, into agriculture to identify pest and to maximize productivity [11].

This paper introduces transfer learning, a deep learning approach, for identifying the invasion of *T. absoluta* at early stages. The approach reinforces classification of leaf images collected from a field setup in a controlled environment (controlled environment refers to preventing the spread of *T. absoluta* to other neighboring tomato fields using net house). Convolution Neural Network was selected as it performs automatic feature extraction thereby saving experts from the labor-intensive task of feature extraction that usually generates erroneous results hence making it more accurate and computational efficient. In essence, training a CNN through transfer learning has the ability of improving computational performance by speeding up the training time through reuse of models that were trained on similar tasks. Transfer learning allows the use of fewer data in training a neural network compared to training from scratch that requires large amounts of data.

2. Related works

Various studies have revealed that image-based plant diagnosis methods generate more accurate results compared with human visual diagnosis.

2.1. Computer vision in agriculture crops

Ramcharan et al. used a pretrained InceptionV3 to detect incidence of three cassava diseases and two pests on the image dataset with 11,670 images collected from a field in Tanzania [12]. The model could correctly identify the diseases and pest damages with various accuracies: 98%, brown leaf spot; 96%, red mite damage; 95%, green mite damage; 98%, cassava brown streak disease; and, 96%, cassava mosaic disease. The study recommended transfer learning as a powerful deep learning technique for developing highly performing classifiers.

Maize, the source of starch crop grown worldwide, is also affected by diseases and pests that have devastating effects on its productivity – a consequence that threatens food security. Dechant and his colleagues [13] used convolutional neural network to detect a disease called Northern leaf blight (NLB) in maize. The study involved inoculation of maize leaves with fungal, a causal agent of NLB, for acquiring dataset from the infected plant. The analysis was carried out on 1796 images composed of health and infested images, and the authors' model yielded an accuracy of 96.7% on the dataset.

The authors in [14] used a pretrained deep learning model to identify three corn leaf diseases. They used PlantVillage dataset containing 8506 healthy and unhealthy corn leaf images; the unhealthy ones had the following diseases: common rust, northern blight, and gray spot. The results obtained after training the model were 98.95%, 98.25%, and 98.79% for the ResNet50, InceptionV3, and MobileNet, respectively. The study revealed that the pretrained deep learning models perform well and can be widely adopted in other agricultural crops.

Another work by Liu et al. [15] proposed a deep learning model to classify four diseases from apple leaves dataset containing 1053 images of diseased and healthy leaves. The author used AlexNet architecture to classify apple Mosaic, Rust, Brown spot, and Alternaria leaf spot. The approach attained an overall accuracy of 97.62%.

Furthermore, Lu et al. successfully identified 10 rice diseases from a dataset of 500 images containing health, and infected leaves of rice and stems [16]. The study used CNN to more accurately classify the images into their respective classes. The authors concluded that CNN yields better results compared with the traditional machine learning techniques of identifying diseases on rice.

2.2. Computer vision in tomato disease identification

Several studies have proposed deep learning as an effective approach of diagnosing various tomato stress. Consequently, we have witnessed great revolution in agriculture, including substantial increase in crop production. The study by Zhang et al., for instance, used CNN architectures, pretrained on 5550 images (from an open access repository), to identify eight tomato diseases: early blight, yellow leaf curl, corynespora leaf spot, leaf mold, virus, late blight, septoria leaf spot, and two-spotted spider mite [17]. All the authors' models could clearly and correctly classify the diseases at the following performances: 95.83%, AlexNet; 95.66%, GoogleNet; and, 96.51%, ResNet50.

Brahimi et al. compared the performances of shallow models (Simple Vector Machine and Random Forest) against pretrained deep models (AlexNet and GoogleNet) in the identification of nine tomato diseases [18]. The pretrained deep models outperformed the shallow models by identifying the diseases with high accuracies of 98.66% and 98.53% for AlexNet and GoogleNet, respectively; Simple Vector Machine and Random Forest generated accuracies of 94.53% and 95.46%, respectively, much lower than those depicted deep models.

Rangarajan et al. [19] used two pretrained deep learning models, VGG16 and AlexNet, to classify six tomato diseases. They used images from PlantVillage dataset containing healthy leaves and unhealthy ones with six tomato diseases: late blight, leaf mold, two-spotted spider might, target spot, mosaic virus, and yellow leaf curl virus. The models attained classification accuracies of 99.24% and 96.51% for VGG16 and AlexNet, respectively.

Ferentinos used deep learning, specifically the VGG model, to recognize eight tomato plant diseases and two tomato pests from a dataset of 87,848 tomato leaves images [20]. The model exhibited a great performance of 99.53% in plant disease detection. This high-level performance suggests that Convolutional Neural Networks are suitable for the automatic detection of plant pest and diseases through the analysis of leaf images.

Generally, various deep learning techniques have been applied for plant disease detection. These techniques have exhibited good performance; however, no technique has been developed to detect tomato leaf miner invasion. In addition, there has been no publicly available dataset with images of tomatoes infected by *T. absoluta*. This lack of dataset hinders progress of research on early detection of *T. absoluta* in tomatoes. Therefore, using images we captured from the field, the current study presents a deep learning approach for *T. absoluta* identification at early stages of the tomato plant growth. Our dataset will be deposited in a public repository to facilitate further research in *T. absoluta* identification from diseased tomato plants.

3. The *TutaAbsoluta* DeepNet

Deep learning consists of multiple processing layers that allow representation learning of multiple level data abstraction. The strength of deep learning emanates from its capacity to create and extrapolate new features from raw representations of input data without being instructed explicitly on which features to use and on how such features can be extracted [7,21]. This technique has been applied in various fields, including computer vision, natural language processing, speech recognition, and bioinformatics. More specifically, in computer vision, deep learning have demonstrated high accuracy in image classification and object This category of deep learning uses CNN that takes in an input image, processes it, and classifies it under certain categories. CNN models can be built from scratch or from transfer learning. Compared with pretrained models through transfer learning, building a model from scratch, however, requires well-labelled data and many computational resources.

In this work, we have proposed a transfer learning approach based on CNN models trained on ImageNet dataset. This approach was preferred because of insufficient number of images available as inputs to our work. Transfer learning can be the best approach for building powerful classifiers, especially under conditions of limited data, through fine-tuning the parameters of network trained on a larger dataset [22]. Consequently, we have explored three CNN architectures, namely VGGNet (VGG16 and VGG19) [23] and ResNet50 [24], and have evaluated their performances on our dataset to classify images into correct categories.

VGGNet is the widely used architecture for ImageNet, and composes of VGG16 and VGG19 with 16 and 19 weight layers, respectively. The architecture takes 224 $\times$224 input images and generates multiple outputs with probabilities corresponding to each class. VGG16 contains thirteen convolution layers, three fully connected layers, and five pooling layers. Furthermore, VGG19 contains sixteen convolution layers, three fully connected layers, and five pooling layers. Convolutional layers are used for extracting features from an image; each layer contains a 3x3 filter with a one-pixel stride and a ReLU activation function. The output layer contains a sigmoid activation function, which is used for classification.

ResNet50 is a convolutional neural network trained on more than a million images from the ImageNet database. This network takes a 224 $\times$224 image and produces an output with a probability of a specific class. ResNet50 contains 50 layers deep, and can classify images into 1000 object categories, including keyboard, mouse, pencil, and animals. In 2015, ResNet emerged as the first winner of the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) classification task.

We expect that CNN can learn feature contained in our training data automatically and use them to classify unseen data. Hence we will no longer rely on experts in identifying the features associated with *T. absoluta* infection. The approach will

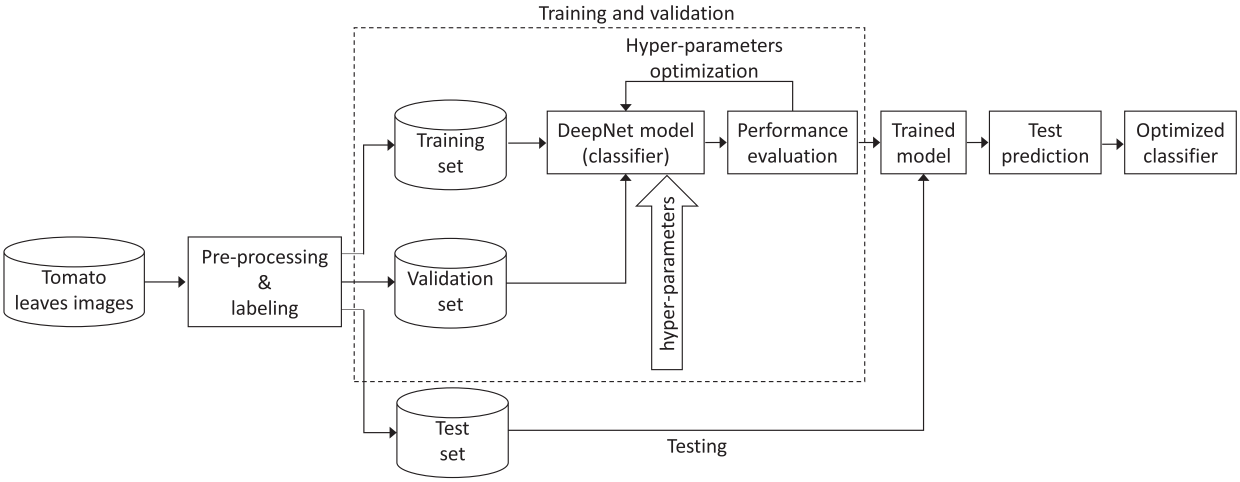

Fig. 1. Proposed conceptual framework.

Fig. 2. Inoculation process performed by agricultural expert in our in-house field located in Arusha, Tanzania.

improve the method of *T. absoluta* identification as farmers will be able to detect the invasion earlier and take appropriate measures to rescue the farms and hence improve production.

4. Experimental setup

4.1. The dataset

Two in-house experiments were conducted between two seasons of tomato growth (August–November 2018 and January–May 2019) in Arusha region, located at the Northern part of Tanzania. Guided by the agricultural expert, the tomatoes were inoculated with *T. absoluta* Fig. 2 (controlled from other pests) under commonly practiced agronomic practice at the early growth stage. Images of tomato plants were collected using the Canon EOS Kiss X7 camera with a resolution of 5184 $\times$3456 pixels. The dataset contained colored images of healthy and unhealthy tomato plants (where the unhealthy ones were inoculated with *T. absoluta* larvae), making 2145 images (330 infested with *T. absoluta*) as examined and labelled by an agricultural expert. The images were collected within 14 days from the day of inoculation in each experiment. We focused on capturing the upper part of the plant at nadir, approximately 40 cm away from the plant, specifically the plant crown because this part is always affected at early growth stages of the plant. Fig. 3 shows sample leaf images collected from the field.

The dataset collected from the field contains more images with healthy tomato leaves than those infected with *T. absoluta*. This situation introduces data imbalance, implying a huge difference between number of samples per class. To reduce the bias that our neural network may encounter towards health samples, the number of samples per class should be balanced. Therefore, to address the data imbalance, we used the following approach: 10% of the infected images were held as test set, and the remaining 90% was sub-divided into training and validation sets at the ratios of 75:25, 80:20, and 85:25 as in Table 1. For the healthy leaves, the images equivalent to 10% of the infected images were held as the test set, and the

(a) (b) (c) (d) (e)

Fig. 3. Development of the mines associated with *T. absoluta* infection. The red bounding boxes show the infected leaf after inoculation with the pest. (a) is the health leaf before inoculation and the red boxes shows the infected leaf, in (b), (c), (d) and (e) is the infected leaf on the 2nd, 4th, 6th, and 8th days respectively, after inoculation with the *T. absoluta*. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Table 1
The numbers of images in training,validation and testing.

Class	Dataset	Number of images for Training	Number of images for Validation	Number of images for Testing
Health	75:25	223 ×6	74 ×6	33
	80:20	237 ×6	60 ×6	33
	85:15	252 ×6	45 ×6	33
Not	75:25	223	74	33
Health	80:20	237	60	33
	85:15	252	45	33

remaining percentage of images was divided into six clusters, each with 297 images. The overall accuracy was calculated by averaging over the six runs on the clusters.

4.2. Image preprocessing

Image preprocessing refers to the manipulation of raw image data before being processed by the deep learning algorithm, the purpose being to enhance data quality. Building a well-performing model requires careful consideration of the network architecture as well as the input data format. We, therefore, pre–processed our dataset to allow the proposed model undertake intelligent diagnosis of extracting appropriate features from the images [25].

The pre-processing involved three stages: image labeling, resizing, and augmentation. In the second stage (resizing), the goal was to generate 224 ×224 images required by VGG16, VGG19, and ResNet50. The standard *resize* function in Keras can, by default, resample an input image to a target size. We resized all images to uniform sizes of 256 ×256 for the *resize* function to resample such resized images according to the proposed architecture requirements. The last pre-processing stage (augmentation) ensures the availability of a large amount of training data to clearly learn features contained in the training data and to attain high classification accuracy on the unseen data. Because of insufficient data in our research, a challenge that could promote overfitting and generalization (on test data) issues, we performed several random augmentations, including rescaling, shearing, flipping, zooming, rotation, and channel shifting. This approach increased the size of our dataset and enabled our classifier to learn more features while achieving the outstanding performance [26].

4.3. Training our classifier

We used three pretrained architectures (VGG16, VGG19, and ResNet50) as classifiers that were fine-tuned through transfer learning. The fully connected layer for each pretrained architecture was replaced by the new layer, and then we fine–tuned the convolutional blocks for the VGGnet and fine–tunedthe top residual block for ResNet50. Firstly, we freezed all layers, except the new added fully connected layer, such that this layer could be trained on the output of the final convolutional layer, generating the learned weights that could be used as the initial values in fine tuning. Thereafter, the top convolutional layers for VGG16 and VGG19, and the top residual block for ResNet50, were unfreezed and trained with the new fully connected layer.

We conducted hyper-parameters search when training our model, and achieved the hyper-parameter values with an optimal performance (Table 2). We trained our classifiers using 1000 epochs with a batch size of eight and a Stochastic Gradient Descent (SGD) Optimizer of a learning rate of 1×10^{-5}.

We stopped the training process at the point when performance on a validation dataset started to degrade. The training process was evaluated after each epoch by using the validation set. In other words, if the validation loss started to increase then the training process was stopped. We stopped the training process at the point when performance on a validation

Table 2
The hyper-parameters used during training.

Parameter	Value
Epochs	1000
Batch size	8
Optimizer	SGD
Learning rate	1e-5
Dropout	0.5
Momentum	0.9
Early stopping	50 epochs

Table 3
Classifier performance for every Dataset, F1–score $_{\{\text{mean precision, mean recall, overall accuracy}\}}$.

Dataset	VGG16	ResNet50	VGG19
75:25	$0.901_{\{0.909,\ 0.901,\ 0.901\}}$	$0.852_{\{0.856,\ 0.853,\ 0.853\}}$	$0.839_{\{0.852,\ 0.841,\ 0.841\}}$
80:20	$0.906_{\{0.915,\ 0.915,\ 0.905\}}$	$0.854_{\{0.867, 0.856,\ 0.856\}}$	$0.831_{\{0.853,\ 0.841,\ 0.836\}}$
85:15	$0.919_{\{0.922,\ 0.919,\ 0.919\}}$	$0.868_{\{0.871,\ 0.868, 0.868\}}$	$0.831_{\{0.851,\ 0.833,\ 0.833\}}$

dataset started to degrade. The training process was evaluated after each epoch by using the validation set. We set up a stopping patience of 50 epochs that allowed the training to continue for an additional 50 epochs after the point where performance started to degrade. The initial learning rate was reduced by a factor of 0.2 if no improvement was observed after the stopping patience. And, if the stopping patience was reached again with no improvement, the training process was stopped.

The test dataset used for evaluating our classifier contained 33 images of healthy leaves and 33 images of infected leaves. We used these images to evaluate the performance of our classifier on the new images. The results were visualized in a confusion matrix (Fig. 5), which suggests that the proposed classifier can achieve a promising performance when applied on a new dataset.

The experiments were conducted on a desktop computer, pre-installed with Ubuntu 18.04 and equipped with one Intel Core i9-9900 3.6 GHz CPU (16 GB RAM) accelerated by one GeForce RTX 2080Ti GPU (12 GB memory). The Keras deep learning library with Tensorflow backend was used.

5. Results and discussions

All the CNN architectures presented in Section 4.3 were trained on our dataset by using the hyper-parameters in Table 2. Such architectures were compared based on their performances on the test dataset, and based on various metrics, such as accuracy, precision, recall, and F1-score. Performance evaluation was done by averaging the metrics over six runs for each dataset division.

Table 3 reports F1-score, mean precision, mean recall, and overall accuracy of the classifiers trained on each dataset, as calculated by using equations (1) through (4) below:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$

$$\text{Precision} = \frac{TP}{TP + FP} \tag{2}$$

$$\text{Recall} = \frac{TP}{TP + FN} \tag{3}$$

$$\text{F1-score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{4}$$

where TP = ”True Positive”, number of images with *T. absoluta* and classified as having *T. absoluta*; TN =”True negative”, number of images with no *T. absoluta* and classified as not having *T. absoluta*; FP = ”False Positive”, number of images with no *T. absoluta* and classified as having *T. absoluta*; and, FN = ”False Negative”, number of images with *T. absoluta* and classified as not having *T. absoluta*.

The overall accuracy was considered as the evaluation metric for our experiments. The best performance accuracy, attained by VGG16 on 85:15 dataset, was 91.9%.

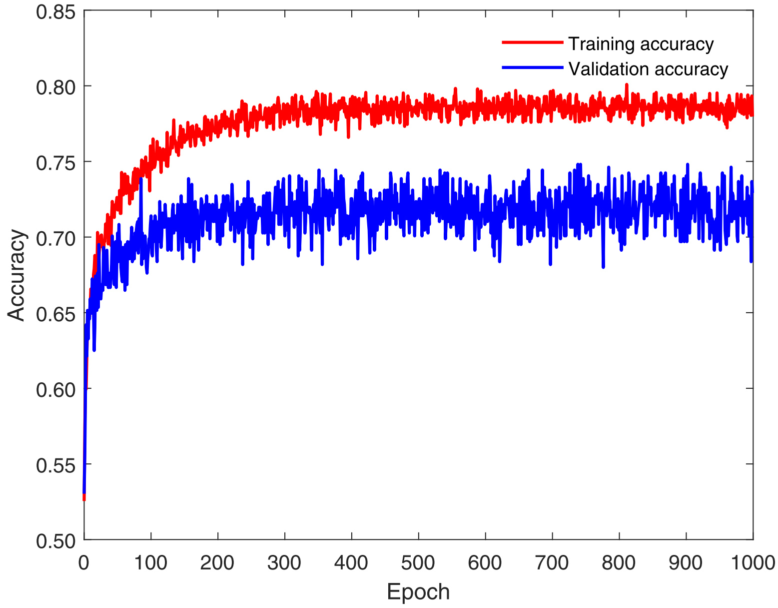

Fig. 4. Training and validation accuracy for the best classifier.

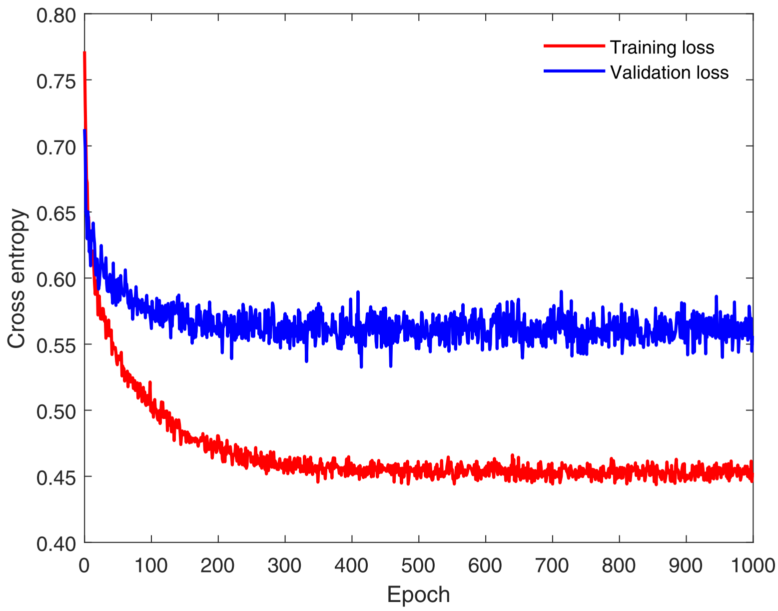

Fig. 5. Training and validation loss for the best classifier.

We have included the learning curves showing the average accuracy and the average loss across the six clusters versus the number of epochs during training process, respectively. Fig. 4 shows that the validation accuracy rises fast at the early training stages and rises slowly at the later stages. In Fig. 5, the losses fell rapidly at the early stages and slowly afterwards. This observation implies that our model learns well the features contained in our dataset at initial and later stages.

To evaluate the ability of our classifier to generalize the unseen images, we performed the prediction on 66 images that were unused during the training process. The confusion matrix in Fig. 6 shows how well the classifier could classify the images into correct categories.

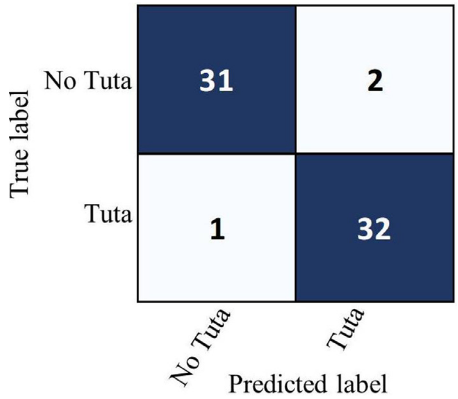

Fig. 6. The confusion matrix for the classifier.

6. Conclusion and future research

In this paper, we have proposed a deep learning model for identifying *T. absoluta* pest in tomato plants. We have used transfer learning through VGG16, VGG19, and ResNet50 models, pretrained on the ImageNet, to train classifiers on our dataset. The training of the models was performed using a dataset with 2145 images with healthy and infected leaf images collected from an in-house experiment. The highly performing model was VGG16, which achieved an overall accuracy of 91.9% in the classification of the previously unseen 66 images from the test set. The results suggest that transfer learning is a powerful method that can achieve high accuracy in the identification of *T. absoluta* pest from tomato plant leaf images. Our method performs automatic feature extraction, in the interest of saving the researchers' time from the labor-intensive task of feature extraction that usually generates erroneous results.

In future, scholars may collect more dataset to increase the performance of our model to classify the unseen images. Consequently, the model will be enhanced with the capability of quantifying the severity of *T. absoluta* invasion in the farm. Therefore, we may have a decision support system to enable farmers take appropriate measures of rescuing the farm after detecting invasion at an early stage of tomato plant growth.

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.

Acknowledgments

This work was support by the African Development Bank (AfDB),United Republic of Tanzania, through project No. P-Z1-IA0-016, and grant No. 2100155032816 together with Tokyo University of Agriculture; the authors would like to express gratitude appreciation for this financial support.

References

[1] P. Schreinemachers, E.B. Simmons, M.C. Wopereis, Tapping the economic and nutritional power of vegetables, Glob. Food Secur. 16 (2018) 36–45.
[2] Tomatonews, The global tomato processing in 2018, 2019(Accessed 9 July 2019).
[3] N.B. o. S. Ministry of Agriculture, 2016/17 annual agriculture sample survey initial report, 2016/17.
[4] R.Y. Guimapi, S.A. Mohamed, G.O. Okeyo, F.T. Ndjomatchoua, S. Ekesi, H.E. Tonnang, Modeling the risk of invasion and spread of Tuta absoluta in africa, Ecol. Complex. 28 (2016) 77–93.
[5] N. Zekeya, M. Chacha, P.A. Ndakidemi, C. Materu, M. Chidege, E.R. Mbega, Tomato leafminer (Tuta absoluta meyrick 1917): athreat to tomato production in Africa, J. Agric. Ecol. Res.Int. (2017) 1–10.
[6] N. Zekeya, P.A. Ndakidemi, M. Chacha, E. Mbega, Tomato leafminer, Tuta absoluta (meyrick 1917), an emerging agricultural pest in sub-saharan Africa: current and prospective management strategies, Afr. J. Agric. Res. 12 (6) (2017) 389–396.
[7] Y. LeCun, Y. Bengio, G. Hinton, Deep learning, Nature 521 (7553) (2015) 436.
[8] Y. LeCun, Y. Bengio, et al., Convolutional networks for images, speech, and time series, Handb. Brain Theory Neural Netw. 3361 (10) (1995) 1995.
[9] H.E. Tonnang, S.F. Mohamed, F. Khamis, S. Ekesi, Identification and risk assessment for worldwide invasion and spread of tuta absoluta with a focus on sub-saharan africa: implications for phytosanitary measures and management, PLoS One 10 (8) (2015) e0135283.
[10] T.J. Maginga, T. Nordey, M. Ally, Extension system for improving the management of vegetable cropping systems, J. Inf. Syst. Eng.Manag. 3 (4) (2018).
[11] S.R. Zahedi, S.M. Zahedi, et al., Role of information and communication technologies in modern agriculture, Int. J. Agric. Crop Sci. 4 (23) (2012) 1725–1728.
[12] A. Ramcharan, K. Baranowski, P. McCloskey, B. Ahmed, J. Legg, D.P. Hughes, Deep learning for image-based cassava disease detection, Front. Plant Sci. 8 (2017) 1852.
[13] C. DeChant, T. Wiesner-Hanks, S. Chen, E.L. Stewart, J. Yosinski, M.A. Gore, R.J. Nelson, H. Lipson, Automated identification of northern leaf blight-infected maize plants from field imagery using deep learning, Phytopathology 107 (11) (2017) 1426–1432.

[14] P. Ouppaphan, Corn disease identification from leaf images using convolutional neural networks, in: 2017 21st International Computer Science and Engineering Conference (ICSEC), IEEE, 2017, pp. 1–5.
[15] B. Liu, Y. Zhang, D. He, Y. Li, Identification of apple leaf diseases based on deep convolutional neural networks, Symmetry 10 (1) (2017) 11.
[16] Y. Lu, S. Yi, N. Zeng, Y. Liu, Y. Zhang, Identification of rice diseases using deep convolutional neural networks, Neurocomputing 267 (2017) 378–384.
[17] K. Zhang, Q. Wu, A. Liu, X. Meng, Can deep learning identify tomato leaf disease? Adv. Multimed. 2018 (2018).
[18] M. Brahimi, K. Boukhalfa, A. Moussaoui, Deep learning for tomato diseases: classification and symptoms visualization, Appl. Artif. Intell. 31 (4) (2017) 299–315.
[19] A.K. Rangarajan, R. Purushothaman, A. Ramesh, Tomato crop disease classification using pre-trained deep learning algorithm, Procedia Comput. Sci. 133 (2018) 1040–1047.
[20] K.P. Ferentinos, Deep learning models for plant disease detection and diagnosis, Comput. Electron. Agric. 145 (2018) 311–318.
[21] S.H. Lee, C.S. Chan, S.J. Mayo, P. Remagnino, How deep learning extracts and learns leaf features for plant classification, Pattern Recognit. 71 (2017) 1–13.
[22] M.M. Ghazi, B. Yanikoglu, E. Aptoula, Plant identification using deep neural networks via optimization of transfer learning parameters, Neurocomputing 235 (2017) 228–235.
[23] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, arXiv:1409.1556 (2014).
[24] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 770–778.
[25] H.-J. Jeong, K.-S. Park, Y.-G. Ha, Image preprocessing for efficient training of yolo deep learning networks, in: 2018 IEEE International Conference on Big Data and Smart Computing (BigComp), IEEE, 2018, pp. 635–637.
[26] L. Perez, J. Wang, The effectiveness of data augmentation in image classification using deep learning, arXiv:1712.04621 (2017).