A transfer learning-based deep neural network for tomato plant disease classification

Fadwa Lachhab[1,2], El Mahdi Aboulmanadel[2]
[1]Lab. SIV, Department of Computer Science, Faculty of Sciences, Ibn Zohr University, Agadir, Morocco
[2]LIDRA Lab, Polytechnic School of Agadir, International University of Agadir, Agadir, Morocco

Article Info

Article history:

Received Feb 21, 2024
Revised Oct 23, 2024
Accepted Nov 14, 2024

Keywords:

Convolutional neural network
Disease detection
Image acquisition
Sustainable farming
Tomato plant diseases
Transfer learning model

ABSTRACT

The agriculture sector plays a significant role in Morocco's economy, and tomato farming is an essential component of this industry. However, tomato plants are prone to various diseases that can adversely affect productivity and quality. A novel approach to detect tomato plant diseases is proposed int this study, by modeling and developing a transfer learning-based convolution neural network (CNN) model that processes real-time images. The model is trained and validated with a deep CNN using a private dataset of 18,159 annotated tomato leaf images collected from experimental farms over five months. The performance of our residual neural network (ResNet-50) model is evaluated using stochastic gradient descent (SGD) and adaptive moment estimation (Adam) optimizers to demonstrate superior efficiency. Farmers can simply send images of their tomato leaves through our platform, and the trained model will identify accurately the disease. The developed model demonstrates exceptional performance, achieving a 0.96 F1 score and an 97% accuracy rate when tested on a dataset generated from real-world fields. This approach not only improves disease detection but also contributes to sustainable farming practices and enhanced productivity.

Corresponding Author:

Fadwa Lachhab
Lab. SIV, Department of Computer Science, Faculty of Sciences, Ibn Zohr University
Agadir, Morocco
Email: f.lachhab@uiz.ac.ma

1. INTRODUCTION

Agriculture plays a vital role in Morocco's economy, providing employment, food security, and contributing to the nation's overall growth. Tomato farming is a significant component of the Moroccan agricultural sector, with the country being among the top exporters of tomatoes worldwide [1]. Morocco has particularly established fresh tomatoes as a prominent export product, with an impressive export volume of 517 thousand tons [2]. This success positions tomatoes as one of the leading agricultural commodities driving the country's export market. The significant quantity of exported fresh tomatoes highlights their importance and desirability in international trade. This achievement showcases Morocco's ability to produce high-quality tomatoes that meet the demands of global consumers, contributing to the country's economic growth through agricultural exports. Ensuring the health and productivity of tomato crops is crucial for both the national economy and the livelihoods of Moroccan farmers.

Plant diseases are a significant challenge for agriculture worldwide, causing substantial losses in crop yield and quality. Among various crops, tomato plants are particularly prone to various diseases, which can have adverse effects on productivity and the overall quality of the produce [3]. Some common tomato plant diseases include blight, bacterial spot, mosaic virus, and other tomato plant diseases, the latter includes highly destructive pest that can cause considerable damage to tomato crops [4]. To identify plant diseases, farmers often turn to pathologists or consult their personal resources [5]. In fact, the considered time and technical knowledge can be very challenging for farmers. Therefore, the need for real-time and accurate recognition solution is crucial.

Over the past few years, artificial intelligence (AI) has gained recognition as a valuable technology for tackling a variety of issues in agriculture, including disease detection and management [6]–[8]. Deep learning algorithms, in particular, have proven highly effective in understanding intricate relationships and capturing essential features in real-world applications [9]–[11]. Among these algorithms, as a powerful method for direct decision-making, convolutional neural networks (CNNs) have gained significant traction and object detection based on image analysis [12]–[14]. These AI-driven models can analyze extensive data set, including images of plant leaves, to identify and classify diseases with high accuracy and efficiency.

Several studies using methods of deep learning have been employed to identify diseases in plants. Li *et al.* [15] provides an extensive explanation of diverse deep learning techniques employed for visualizing various plant diseases and classifying them. The methods utilized in the study are evaluated using a range of performance metrics to assess their effectiveness. Sladojevic *et al.* [16] developed a deep convolutional model to identify 13 distinct types of plant diseases within healthy leaves, while successfully differentiating the leaves from their environment, achieving a mean accuracy of 96%. Cheng *et al*. [17] used deep residual learning in complex background to identify agricultural pests. They compared support vector machine and traditional back propagation neural networks by using pre-trained models ResNet and AlexNet. The best accuracy they achieved is 98.6% for 10 classes of crop pest images. Ozguven and Adem [18] proposed an updated architecture of faster region-convolutional neural network (R-CNN) for the automatic recognition of sugar beet leaf disease. The imaging-based model used changed CNN parameters to increase the success of faster R-CNN architecture. However, the classification was trained with 155 images and found to be 95.4%. SoyNet [19] proposed a deep learning-based CNN for soybean plan diseases detection, by subtracting complex background and extracts the leaf part. Their proposed model was compared with six pre-trained models namely ResNet-50, Dense121, XceptionNet, LeNet, GoogleLeNet and VGG19. They achieved identification accuracy of 98.14%.

Powerful tools have emerged in the form of deep learning models for detecting and classifying tomato leaf diseases [20]–[22], and among these models, residual neural network (ResNet) stands out as a prominent architecture. In this research [23], ResNet was compared against other CNN architectures for the recognition of nine different diseases and pests. The proposed model achieves an identification accuracy of 82.53%. For classification performance, Zhang *et al.* [24] demonstrates that the ResNet architecture outperforms the others with performance metrics of 95.83% for AlexNet, 95.66% for GoogleNet, and 96.51% for ResNet-50. This result is achieved using the stochastic gradient descent (SGD) optimizer. On other hand, transfer learning-based CNN improves deep neural networks by eliminating the need for extensive data mining and annotation. Thangaraj *et al.* [25] proposed a fine-tuning model using real-time images. The proposed model was fed to ResNet-50 and evaluated using Adam and SGD optimizers.

Based on the literature review, it is observed that there is limited research in the real-time detection and identification of diseases in plant leaves. Most deep learning-based models are optimized for offline usage, making them unsuitable for real-time crop disease detection. Moreover, there is a lack of real-world datasets for training and testing models, the images datasets used by various authors in existing literature are designated for simulation and not for implementation in a real-world scenario.

In this study, we propose a novel AI-driven method to detect and recognize tomato plant diseases. We aim to achieve higher accuracy using ResNet architecture under different optimizers. Our solution integrates a deep learning model into our platform, enabling both small and large-scale farmers to make informed decisions regarding fertilizer use to confront tomato diseases in their crops. The key contributions of this study are summarized below:

- Develop and implement a transfer learning model for classifying and detecting tomato diseases using our private DeepLeafSet dataset which comprises a total of 18,159 images. A total of ten classes were distinguished, including a healthy tomato leaf class and nine tomato diseases: bacterial spot, curl virus, early blight, late blight, leaf mold, mosaic virus, septoria leaf, spider mites, and target spot.
- Evaluate the performance of the proposed approach by utilizing both SGD and Adam optimizers for ResNet-50 model. We compare their performance and evaluate their respective impacts on convergence speed, stability and overall accuracy.

The rest of this paper is organized as follows: the “Method” section presents the deep learning model for tomato plant disease detection, including the dataset utilized for both training and validation, the developed model architecture using ResNet-50, the training process, and the tuning parameters. The “Results and Discussion” section illustrates the experimental comparison between SGD and Adam optimizers, and reports the training and validation parameters. Finally, “Conclusions and Perspectives” section provides a summary of the study and offers suggestions for future work.

2. METHOD

2.1. The dataset

The DeepLeafSet is the dataset used for training and validation of our deep learning model, consisting of 18,159 annotated tomato leaf images. These images were collected over a period of five months from various locations, including the experimental farms of DeepLeaf, Mohammed VI Polytechnic University (UM6P), and several private farms dedicated to tomato cultivation with greenhouses. In total,13 greenhouses were involved in the image collection process. The tomato leaf images were captured from tomato plants at various stages of growth, ranging from young seedlings to mature plants. This intentional variation in the age of the tomatoes, as depicted in Table 1, aimed to diversify the dataset and account for the visual differences and disease progression that may occur at different growth stages.

The disease categories in DeepLeafSet were carefully selected to represent common diseases found in tomato plants in Morocco and Africa. A total of ten classes were distinguished in our dataset as shown in Table 2, including a healthy tomato leaf class and nine tomato diseases. The primary objective of curating this comprehensive dataset was to facilitate the creation of a reliable and precise classification model for tomato plant diseases. Tomato plant diseases pose a significant pest problem, leading to substantial crop losses estimated at around 30% of production for farmers in Morocco and Africa [26]. By leveraging this dataset, our deep learning model seeks to offer an effective solution for early disease detection and intervention, thereby aiding farmers in mitigating losses and improving crop yields.

Table 1. Age interval and its distribution in the DeepLeafSet dataset

Age interval	Percentage (%)	Number of images
55-60 days	20	3,632
61-66 days	20	3,632
67-72 days	20	3,632
73-78 days	20	3,632
79-84 days	20	3,631
Total	100	18,159

Table 2. Table of class names and diseases with total images in the DeepLeafSet dataset

Disease class	Class name	Total images
Bacterial tomato	Tom__Bac	1,703
Curl virus tomato	Tom__CurlV	4,289
Early tomato blight	Tom__EarlBlig	1,918
Healthy tomato	Tom__Healt	2,273
Late tomato blight	Tom__LatBlig	1,528
Leaf tomato mold	Tom__LeaMol	1,263
Mosaic tomato virus	Tom__MosV	1,300
Septoria tomato leaf	Tom__SepLeaf	1,418
Spider tomato mite	Tom__SpidMite	1,342
Target tomato spot	Tom__TargSp	1,125
Total		18,159

2.1.1. Collection and annotation process

The image collection process for DeepLeafSet was conducted by a team of five agronomy experts who used their smartphones to capture tomato leaf images. To prevent resolution degradation caused by image compression on messaging platforms, images were sent as documents through WhatsApp. Each team member was tasked with sending images of tomato leaves to our platform along with the corresponding disease label (e.g., tomato__early_blight). Automating the storage and labeling process of data collection, as shown in Figure 1, allowed us to focus on image acquisition, thereby maximizing the dataset's size and diversity.

2.1.2. Dataset preparation

Before using the dataset for training and validation, it was essential to preprocess and prepare the images to guarantee the deep learning model performances and efficiency. The images were subsequently adjusted to a consistent size (e.g., 224×224 pixels) to ensure compatibility with the input dimensions of the deep learning model. Additionally, data augmentation techniques, such as random rotations, flips, and zooming, were implemented to enhance the dataset's diversity and boost the model's capability to adapt to novel data. The following augmentation steps were performed, along with its respective parameters:

- Zooming (scale range): Each image was randomly scaled by a factor between 0.8 and 1.2 once.
- Flipping_H (horizontal flip): Every image was flipped horizontally with a probability of 50%.
- Flipping_V (vertical flip): Every image was flipped vertically with the probability of 50%.

– Random rotations (angle range): Each image was randomly rotated by an angle between -45 degrees and +45 degrees once.

After preprocessing the data set and applying data augmentation techniques, as illustrated in Table 3, the dataset was split into training and validation subsets following the standard practice of an 80% training and 20% validation split. The data splitting process involved randomly assigning each image to either the training set or the validation set. The training set, was used to train the deep learning model, which contained 80% of the images. The remaining 20% of the images were allocated for the validation set, which was used to evaluate the model's performance and assess its generalization capabilities on unseen data. By splitting the dataset into validation and training partitions, the model can be trained on a substantial portion of records while having an independent subset to measure its performance and detect any overfitting issues. This division allows for an objective evaluation of the model's accuracy, precision, recall, and other performance metrics.

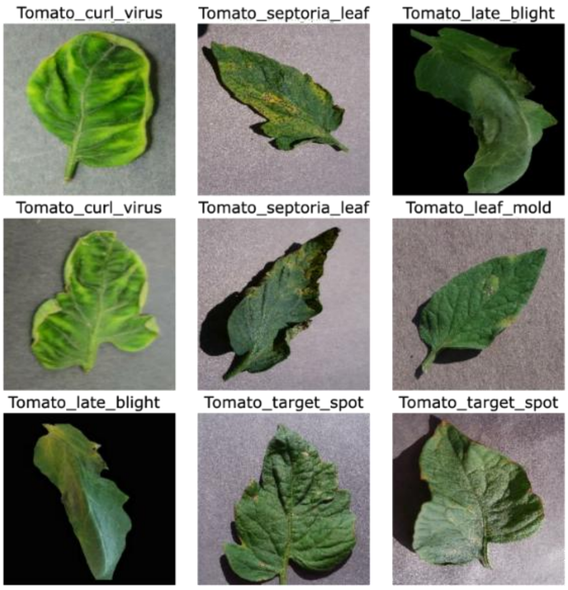


Figure 1. Annotated images examples from DeepLeafSet

Table 3. Number of images per split

Dataset split	Number of images	Percentage (%)
Training	116,218	80
Validation	29,054	20

2.2. Model architecture

The model architecture chosen for the tomato plant disease detection task is the ResNet architecture, specifically ResNet-50. ResNet is a popular deep learning architecture recognized for its capacity to train very deep neural networks without suffering from the vanishing gradient problem. The primary innovation behind ResNet is the introduction of skip connections or shortcut connections, which enable the network to learn residual functions in relation to the inputs of the layers. This design enables the efficient training of deeper networks while maintaining accuracy.

ResNet-50 stands out as a variant of the original ResNet architecture, consisting of 50 layers that include activation layers, batch normalization layers, convolutional layers, and pooling layers. The ResNet-50 architecture revolutionized image recognition tasks by introducing residual connections, which allow for the training of deeper networks. These residual connections enable gradients to propagate effectively during backpropagation, thereby alleviating the vanishing gradient problem associated with deep networks. The ResNet-50 architecture has gained widespread use as a framework in numerous computer vision applications, showcasing its effectiveness and impact in the field.

The diagram provided in Figure 2 illustrates the sequential flow of information within the ResNet-50 model in our approach. Beginning with an input image of a leaf, a series of layers is performed in order to process the information, including the first input layer, followed by convolutional layers (con2d), and subsequently passes through additional layers (e.g. batch normalization and activation layers). This intricate architecture allows the model to capture and comprehend complex representations of plant diseases, ultimately leading to more precise and reliable disease classification. By automating the detection process, our deep learning model offers a valuable tool for farmers, agronomists, and researchers, enabling them to quickly and accurately diagnose plant diseases, take timely preventive measures, and ensure optimal crop health.

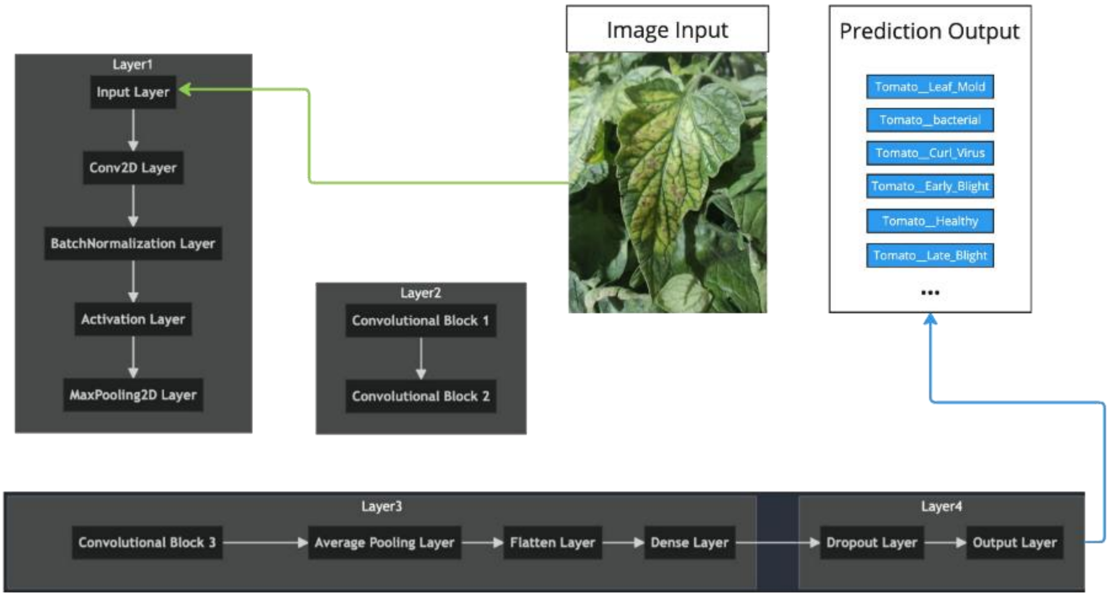


Figure 2. The ResNet-50 model layers architecture

2.2.1. Training process

The training procedure for the ResNet-50 model involves the following steps:

- Transfer learning: Instead of training the model from scratch, we use transfer learning to utilize the knowledge acquired from pre-training the ResNet-50 model on a large-scale dataset, such as ImageNet [27]. This approach helps reduces the training time and improve the performance of the model.
- Fine-tuning: The tomato leaf dataset is fine-tuned with ResNet-50 pre-trained model. The model's weights are updated using backpropagation with a binary cross randomness loss function with the SGD optimization algorithm, a batch size of 32 and 0.001 learning rate. we trained our model with 10 epochs.
- Model evaluation: After training, the validation was performed for model evaluation employing standard evaluation metrics such as precision, accuracy, F1 score and recall.

The ResNet-50 architecture, combined with the transfer learning approach and fine-tuning, enables the development of an efficient and accurate deep learning model for detecting tomato plant diseases. This model holds the promise of greatly enhancing disease detection and management in the agricultural sector.

2.2.2. Tuning parameters and metrics

Tuning parameters, also known as hyperparameters, are the adjustable settings of the training process that can influence the deep learning model performances. Selecting the appropriate values for these parameters is crucial to ensure the model's convergence, generalization, and overall performance. In this part, we discuss the key tuning parameters used in training the ResNet-50 model for tomato plant disease detection and the metrics employed to evaluate the model's performance. The following are the key tuning parameters used in training the ResNet-50 model on the tomato leaf dataset:

- Batch size: the number of images used in each iteration of the training process. In our training, we used a batch size of 32. The choice of batch size can impact both the training speed and the final model's quality.
- Learning rate: the step size used to update the neural network weights during the training. A 0.001 learning rate was used. the speed at which the model masters the data is determined by the learning rate and must be chosen carefully to balance convergence speed and stability.
- Number of epochs: the complete dataset is fed into the model a total number of epochs (e.g., times) throughout the training process. We trained the model for 10 epochs. The choice of the number of

epochs can influence the performance of the model to generalize to new data, an excessive number of epochs may cause overfitting, whereas insufficient epochs can lead to underfitting.

- Optimization algorithm: we aimed to compare the performance of both SGD and Adam optimizers in the context of tomato disease detection. We trained our ResNet-50 model using both optimization algorithms and then evaluated their respective impacts on convergence speed, stability, and overall accuracy.

2.2.3. Method and environment of training

To train our ResNet-50 model on the DeepLeafSet dataset, we utilized Azure machine learning, which provided a powerful and scalable platform for deep learning tasks. The training process leveraged compute instances equipped with Nvidia Tesla GPU clusters with various specifications, as depicted in Table 4. These specifications enabled efficient processing and accelerated model training.

The DeepLeafSet dataset, consisting of a diverse collection of leaf images, was stored in Azure storage blob. This cloud-based storage solution offered robust and reliable data management capabilities, ensuring easy access and seamless integration with the training pipeline. Furthermore, utilizing Azure's scalable infrastructure allowed for efficient handling of large datasets, facilitating continuous updates and improvements to the dataset as new images were collected.

Considering the capabilities of Azure machine learning studio, we implemented the ResNet-50 architecture well-recognized for its outstanding effectiveness in tasks related to image classification. Throughout the training process, we monitored the performance of the model assessed through evaluation metrics including precision, F1-score, accuracy, and recall. This allowed us to assess the model's progress, identify potential areas for improvement, and fine-tune the hyperparameters to optimize its performance.

Table 4. Specifications of compute instance used to train our ResNet-50 model

Specification	Value
GPU Model	Nvidia Tesla V100
GPU Memory	Up to 32 GB or 16 GB HBM2
CUDA Cores	5,120 CUDA cores
Memory Bandwidth	Up to 900 GB/s
Tensor Cores	640 Tensor Cores

2.2.4. Evaluation metrics

To assess the effectiveness of our trained ResNet-50 model using both SGD and Adam optimizers, we employed common evaluation metrics commonly used for binary classification tasks. These metrics include precision, recall, and F1-score. They are defined as follows:

- Precision:

$$Precision = \frac{True\ Positives}{True\ Positives + False\ Positives}$$

True positives indicate the count of instances accurately identified as positive (i.e., correctly classified as a tomato plant disease), while false positives denote the count of instances inaccurately classified as positive (i.e., misclassified as a tomato plant disease). Precision measures the proportion of instances classified as positive (i.e., tomato plant diseases) that are truly positive (i.e., true positives) out of all instances classified as positive.

- Recall:

$$Recall = \frac{True\ Positives}{True\ Positives + False\ Negatives}$$

True positives refer to the count of instances accurately identified as positive (i.e., correctly classified as a Tomato plant disease), while false negatives indicate the count of instances erroneously classified as negative (i.e., misclassified as a different class or not classified as one of the tomato plant diseases). Recall, commonly known as the true positive rate or sensitivity, assesses the ratio of positive instances (i.e., tomato plant diseases) that are correctly classified as positive (i.e., true positives) from all true positive instances.

- F1-score:

$$F1\ Score = \frac{2 * Precision * Recall}{Precision + Recall}$$

The harmonic mean of precision and recall is presented by the F1 score, which provides an equitable assessment of the model's performance by equally considering both metrics. The F1-score use a range from 0 to 1, where a better performance is known with higher values. A score of 1 reflects perfect recall and precision, while a score of 0 demonstrates that the model has not achieved any correct predictions.

3. RESULTS AND DISCUSSION

In this study, two optimizers based on the ResNet architecture were implemented in our proposed transfer learning-based model, utilizing residual blocks with skip connections. These optimizers employ different optimization methods that had an impact on their performance and complexity. The SGD optimizer updates the model's parameters by computing gradients based on a limited sample of the training data. The indicated optimization method is well-known for its simplicity and effectiveness in training deep networks. However, Adam optimizer is a variant of SGD that adapts the learning rate for each parameter individually. It is particularly favored for its ability to handle sparse gradients and demonstrate strong performance on large-scale datasets. While both optimizers contribute to the successful training of ResNet-50, they differ in terms of learning rate adaptation and update mechanisms. SGD is a straightforward and widely used optimization method, whereas Adam offers adaptability and efficiency in terms of learning rate adjustment. The choice between the two optimizers is contingent upon the specific requirements of the task and the characteristics of the dataset.

To compare the performance of the proposed model trained with both SGD and Adam optimizers, several metrics were evaluated. These metrics offer insights into the models' accuracy and loss during the training process. Figure 3 presents the training accuracy and loss curves across different epochs. It can be observed that the SGD optimizer achieves a higher accuracy compared to the Adam optimizer, indicating its effectiveness in improving the model's performance over time.

The results reveal that the SGD optimizer consistently outperforms the Adam optimizer across various metrics and classes in our ResNet-50 model (see Table 5 for the comparison). Specifically, the SGD optimizer achieves higher F1-Score, precision, and recall values for most classes, as illustrated in Figures 4 and 5 indicating its superior ability to make accurate predictions and capture meaningful patterns in the dataset. This comparison highlights the effectiveness of the SGD optimizer in training the ResNet-50 model and its potential to achieve better performance in our plant disease classification tasks.

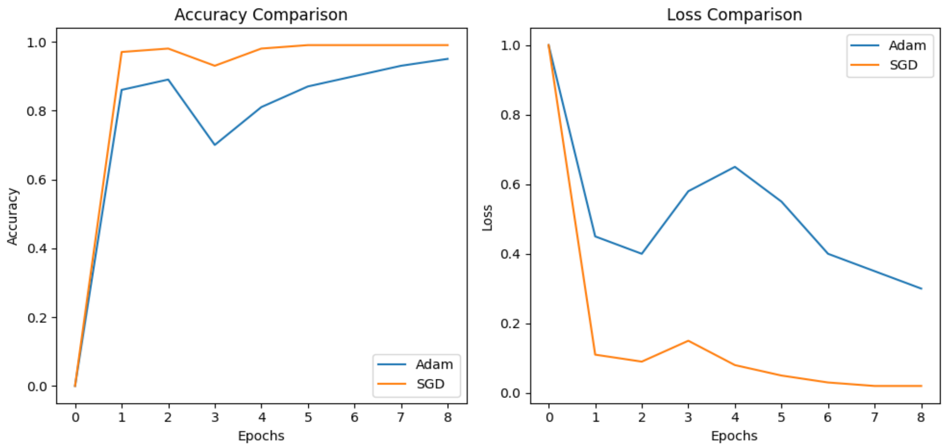


Figure 3. The training accuracy and loss per number of epochs comparison between Adam and SGD

Table 5. Comparison of evaluation metrics for the proposed model using adam and SGD optimizers

Class	Precision (ADAM)	Precision (SGD)	Recall (ADAM)	Recall (SGD)	F1-Score (ADAM)	F1-Score (SGD)	Support (ADAM)	Support (SGD)
tomato_bacterial	0.86	0.97	0.92	1.00	0.89	0.99	425	425
tomato_curl_virus	0.93	1.00	0.89	0.99	0.91	0.99	1071	1071
tomato_eb	0.48	0.64	0.82	0.98	0.61	0.78	200	200
tomato_healthy	0.92	0.99	0.94	1.00	0.93	1.00	318	318
tomato_lb	0.82	0.98	0.62	0.73	0.70	0.84	381	381
tomato_leaf_mold	0.95	1.00	0.89	0.98	0.92	0.99	190	190
tomato_mosac_virus	0.96	1.00	0.72	0.89	0.82	0.94	74	74
tomato_septoria_leaf	0.89	0.99	0.91	0.99	0.90	0.99	354	354
tomato_spider_mite	0.90	0.98	0.92	0.99	0.91	0.98	335	335
tomato_targ_spot	0.92	0.99	0.90	0.99	0.91	0.99	280	280

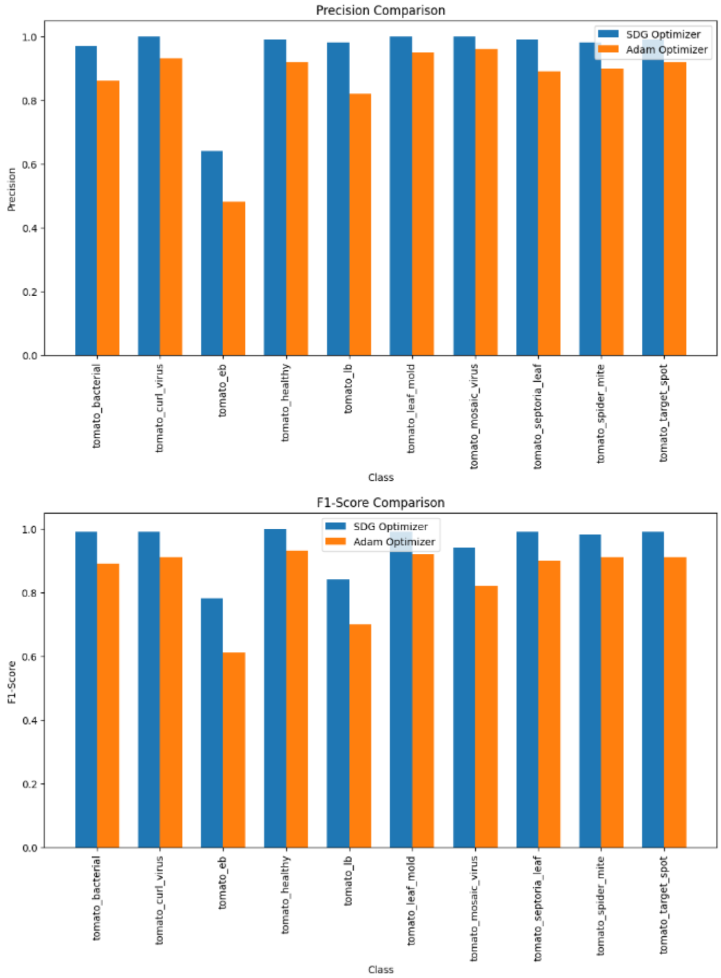


Figure 4. Precision and F1-score evaluation metric comparison of ResNet-50 model with SGD and Adam optimizers

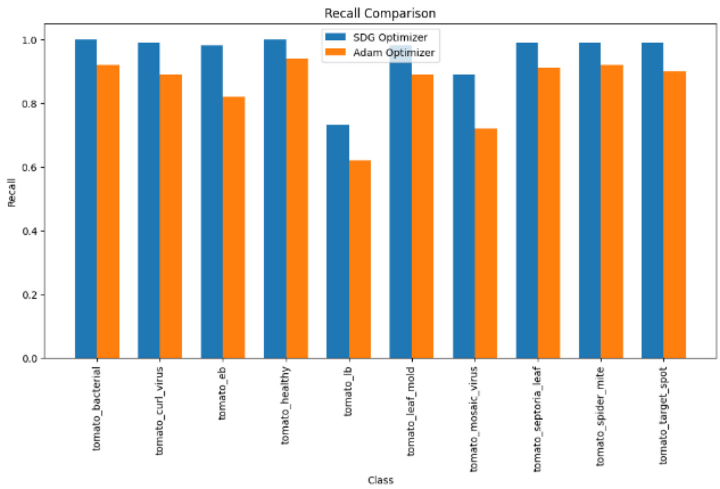
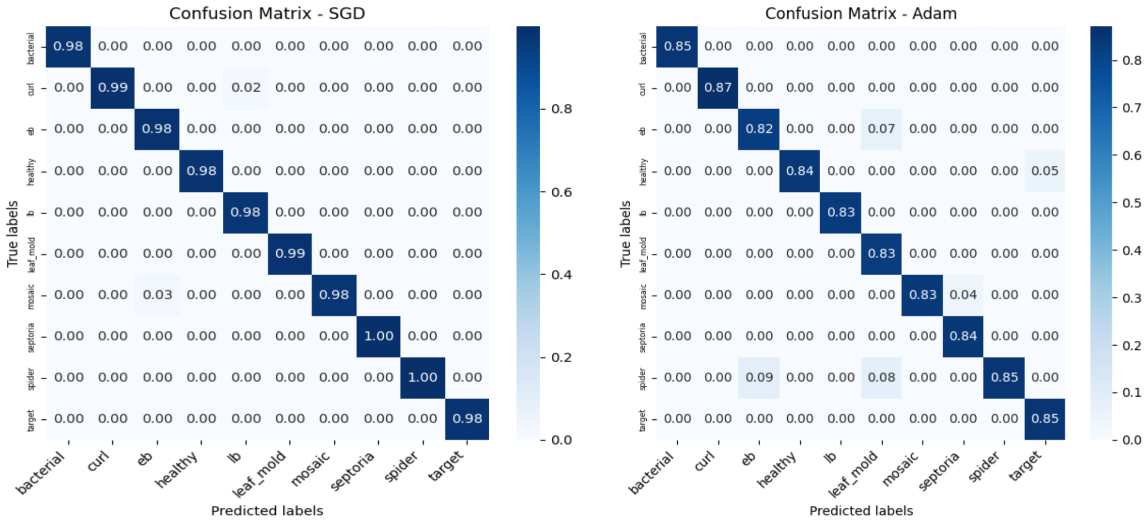


Figure 5. Recall evaluation metric comparison of ResNet-50 model with SGD and Adam optimizers

Additionally, the confusion matrix of the model with Figure 6(a) the SGD optimizer and Figure 6(b) the Adam optimizer, as depicted in Figure 6 shows the classification results for each class. Each cell in the matrix represents the model's prediction accuracy for a given true label and predicted label. Correct classifications are represented by the diagonal elements, whereas the off-diagonal elements indicate misclassifications. As a result, the SGD model in Figure 6(a) showed fewer misclassifications compared to the Adam model in Figure 6(b).

In conclusion, we observed that in our comparison, the SGD optimizer consistently generated higher recall, precision, and F1-scores across most classes compared to the Adam optimizer. Our experiments demonstrate that while both optimizers can achieve high accuracy in tomato plant disease detection, ResNet-50 with the SGD optimizer showed superior performance. It achieves high F1 score of 0.96 with an accuracy of 97%, representing its effectiveness in detecting tomato plant diseases. The real-time classification capability of this model will allow farmers to promptly identify diseased tomato plants and take immediate action, thereby mitigating the impact of diseases on crop yields.



(a) (b)

Figure 6. Confusion matrices of the model with (a) the SGD optimizer and (b) the Adam optimizer

4. CONCLUSION

This study presents a transfer learning-based model designed to detect and manage tomato plant diseases, which significantly impact tomato production in Morocco and Africa. We collected a dataset of 18,159 annotated tomato leaf images and trained a deep learning model using transfer learning with the ResNet-50 architecture based on the SGD optimizer. The model achieved high F1 score of 0.96 with an accuracy of 97%, demonstrating its effectiveness in disease detection compared to the Adam optimizer. Future directions include expanding the model to detect and manage other crop diseases or pests, offering a comprehensive solution for agriculture disease management. Integrating additional data sources, such as weather data or soil condition information, could further enhance the accuracy and effectiveness of our approach. Building on the existing work, future research may also explore the development of a user-friendly chatbot interface. Incorporating the deep learning model into a chatbot could simplify access to technology for farmers, potentially promoting widespread adoption within the farming community.

REFERENCES

[1] K. Benabderrazik, B. Kopainsky, L. Tazi, J. Jörin, and J. Six, “Agricultural intensification can no longer ignore water conservation–A systemic modelling approach to the case of tomato producers in Morocco,” *Agricultural Water Management*, vol. 256, 2021, doi: 10.1016/j.agwat.2021.107082.
[2] WHO and FAO, *The second global meeting of the FAO/WHO international food safety authorities network*, World Health Organization: Geneva, Switzerland, 2019.
[3] A. F. Fuentes, S. Yoon, J. Lee, and D. S. Park, “High-performance deep neural network-based tomato plant diseases and pests’ diagnosis system with refinement filter bank,” *Frontiers in Plant Science*, vol. 9, 2018, doi: 10.3389/fpls.2018.01162.
[4] A. Abbas, S. Jain, M. Gour, and S. Vankudothu, “Tomato plant disease detection using transfer learning with C-GAN synthetic images,” *Computers and Electronics in Agriculture*, vol. 187, 2021, doi: 10.1016/j.compag.2021.106279.
[5] A. M. Tronsmo, D. B. Collinge, A. Djurle, L. Munk, J. Yuen, and A. Tronsmo, *Plant pathology and plant diseases*, Oxford, United Kingdom: CABI Digital Library, 2020.
[6] N. N. Misra, Y. Dixit, A. Al-Mallahi, M. S. Bhullar, R. Upadhyay, and A. Martynenko, “IoT, big data, and artificial intelligence in agriculture and food industry,” *IEEE Internet of things Journal*, vol. 9, no 9, pp. 6305-6324, 2020, doi: 10.1109/JIOT.2020.2998584.

[7] Y. Ampatzidis, L. D. Bellis, and A. Luvisi, "iPathology: robotic applications and management of plants and plant diseases," *Sustainability*, vol. 9, no 6, 2017, doi: 10.3390/su9061010.

[8] İ. Yağ, and A. Altan, "Artificial intelligence-based robust hybrid algorithm design and implementation for real-time detection of plant diseases in agricultural environments," *Biology*, vol. 11, no 12, 2022, doi: 10.3390/biology11121732.

[9] A. Kamilaris, and F. X. Prenafeta-Boldú, "Deep learning in agriculture: A survey," *Computers and Electronics in Agriculture*, vol. 147, pp. 70-90, 2018, doi: 10.1016/j.compag.2018.02.016.

[10] K. Huang, A. Hussain, Q. F. Wang, and R. Zhang, *Deep learning: fundamentals, theory and applications*, Switzerland: Springer Nature, 2019, doi: 10.1007/978-3-030-06073-2.

[11] I.H. Sarker, "Machine learning: Algorithms, real-world applications and research directions," *SN computer science*, vol. 2, no 3, 2021, doi: 10.1007/s42979-021-00592-x.

[12] J. Naranjo-Torres, M. Mora, R. Hernández-García, R.J. Barrientos, C. Fredes, and A. Valenzuela, "A review of convolutional neural network applied to fruit image processing," *Applied Sciences*, vol. 10, no 10, 2020, doi: 10.3390/app10103443.

[13] M. Tripathi, "Analysis of convolutional neural network-based image classification techniques," *Journal of Innovative Image Processing*, vol. 3, no. 2, pp. 100-117, 2021, doi:10.36548/jiip.2021.2.003.

[14] J. Lu, L. Tan, and H. Jiang, "Review on convolutional neural network (CNN) applied to plant leaf disease classification," *Agriculture*, vol. 11, no 8, 2021, doi: 10.3390/agriculture11080707.

[15] L. Li, S. Zhang, and B. Wang, "Plant disease detection and classification by deep learning—a review," *IEEE Access*, vol. 9, pp. 56683-56698, 2021, doi: 10.1109/ACCESS.2021.3069646.

[16] S. Sladojevic, M. Arsenovic, A. Anderla, D. Culibrk, and D. Stefanovic, "Deep neural networks based recognition of plant diseases by leaf image classification," *Computational Intelligence and Neuroscience*, vol. 2016, no 1, 2016, doi: 10.1155/2016/3289801.

[17] X. Cheng, Y. Zhang, Y. Chen, Y. Wu, and Y. Yue, "Pest identification via deep residual learning in complex background," *Computers and Electronics in Agriculture*, vol. 141, pp. 351–356, 2017, doi: 10.1016/j.compag.2017.08.005.

[18] M. M. Ozguven and K. Adem, "Automatic detection and classification of leaf spot disease in sugar beet using deep learning algorithms," *Physica A: statistical mechanics and its applications*, vol. 535, 2019, doi: 10.1016/j.physa.2019.122537.

[19] A. Karlekar, A. Seal, "SoyNet: Soybean leaf diseases classification," *Computers and Electronics in Agriculture 2020*, vol. 172, 2020, doi: 10.1016/j.compag.2020.105342.

[20] M. Brahimi, K. Boukhalfa, and A. Moussaoui, "Deep learning for tomato diseases: classification and symptoms visualization," *Applied Artificial Intelligence*, vol. 31, no 4, pp. 299-315, 2017, doi: 10.1080/08839514.2017.1315516.

[21] A. K. Rangarajan, R. Purushothaman, A. Ramesh, "Tomato crop disease classification using pre-trained deep learning algorithm," *Procedia computer science*, vol. 133, pp. 1040-1047, 2018, doi: 10.1016/j.procs.2018.07.070.

[22] M. Agarwal, A. Singh, S. Arjaria, A. Sinha, and S. Gupta, "ToLeD: Tomato leaf disease detection using convolution neural network," *Procedia Computer Science*, vol. 167, pp. 293-301, 2020, doi: 10.1016/j.procs.2020.03.225.

[23] A. Fuentes, S. Yoon, S.C. Kim, and D.S. Park, "A robust deep-learning-based detector for real-time tomato plant diseases and pests' recognition," *Sensors*, vol. 17, no. 9, 2022, doi: 10.3390/s17092022.

[24] K. Zhang, Q. Wu, A. Liu, and X. Meng, "Can deep learning identify tomato leaf disease," *Advances in multimedia*, vol. 2018, no 1, 2018, doi: 10.1155/2018/6710865.

[25] R. Thangaraj, S. Anandamurugan, and V. K. Kaliappan, "Automated tomato leaf disease classification using transfer learning-based deep convolution neural network," *Journal of Plant Diseases and Protection*, vol. 128, no 1, pp. 73-86, 2021, doi: 10.1007/s41348-020-00403-0.

[26] J. Gressel *et al*., "Major heretofore intractable biotic constraints to African food security that may be amenable to novel biotechnological solutions," *Crop Protection*, vol. 23, no 8, pp. 661-689, 2004, doi: 10.1016/j.cropro.2003.11.014.

[27] A. Krizhevsky, I. Sutskever, and G.E. Hinton, "ImageNet classification with deep convolutional neural networks," *Communications of the ACM*, vol. 60, no 6, pp. 84-90, 2017, doi: 10.1145/306538.

BIOGRAPHIES OF AUTHORS

Fadwa Lachhab is an Assistant Professor at Ibn Zohr University (UIZ), Agadir, Morocco. She received a Ph.D. in Computer Science in 2018 at the Mohamed V University, at the National School of Computer Science and Systems Analysis (ENSIAS) and the International University of Rabat (UIR). Her research interests include various aspects that are related to the design and implementation of distributed and context-aware systems using big data and artificial intelligent techniques. She can be contacted at email: f.lachhab@uiz.ac.ma.

El Mahdi Aboulmanadel holds an engineering degree in computer science from Polytechnic School at the International University of Agadir (in September 2023). He founded DeepLeaf, an agritech startup focused on developing deep learning models for plant disease detection. His work has been recognized with awards, including winning the second prize in the AgriYoung Innovate competition by ADA and UM6P, the first prize in U-founders by UM6P and the first prize in Pitch AgriHack by GoGettaz Africa in Tanzania. He can be contacted at email: mahdi@depleaf.io.