IAES International Journal of Artificial Intelligence (IJ-AI)
Vol. 12, No. 1, March 2023, pp. 162~170
ISSN: 2252-8938, DOI: 10.11591/ijai.v12.i1.pp162-170

Low-cost convolutional neural network for tomato plant diseases classification

Soumia Bensaadi, Ahmed Louchene
Advanced Electronics Laboratory, Department of Electronics, Faculty of Technology, Batna2 University, Batna, Algeria

Article Info

Article history:

Received Aug 28, 2021
Revised Sep 1, 2022
Accepted Sep 30, 2022

Keywords:

Convolutional neural network
Deep learning
Low complexity architecture
Tomato plant disease classification

ABSTRACT

Agriculture is a crucial element to build a strong economy, not only because of its importance in providing food, but also as a source of raw materials for industry as well as source of energy. Different diseases affect plants, which leads to decrease in productivity. In recent years, developments in computing technology and machine-learning algorithms (such as deep neural networks) in the field of agriculture have played a great role to face this problem by building early detection tools. In this paper, we propose an automatic plant disease classification based on a low complexity convolutional neural network (CNN) architecture, which leads to faster on-line classification. For the training process, we used more than one 57,000 tomato leaf images representing nine classes, taken under natural environment, and considered during training without background subtraction. The designed model achieves 97.04% classification accuracy and less than 0.2 error, which shows a high accuracy in distinguishing a disease from another.

Corresponding Author:

Soumia Bensaadi
Advanced Electronics Laboratory, Department of Electronics, Faculty of Technology, Batna2 University
Rue Chahid Boukhlouf M El Hadi, Batna 05001, Algeria
Email: s.bensaadi@univ-batna2.dz

1. INTRODUCTION

Around 2.3 billion people in the world were moderately or severely food insecure in 2021, or nearly 30% of the global population -more than 350 million more people than in 2019, the year before the coronavirus disease of 2019 (COVID-19) pandemic unfolded. They are facing chronic hunger and most of them, 795 milions, live in Africa [1]. This continent is thus categorized as a food deficit nation. Thus, agriculture is the backbone of developing countries, it is an essential source of income that ensures the economic growth and food security, not only because it provides food for our daily life, but also, because of the dependence of all the industries on agriculture both directly and indirectly. Furthermore, agriculture is a major source of employment for people to face hunger, so it is crucial to develop the agriculture sector in any country. Agriculture always faces loss of crops caused by natural conditions, seeds quality and price in the international market, nutrition, or diseases.

Tomato is a popular vegetable crop in all countries. Under undesirable conditions: light, temperature, water, seeds quality and nutrition, this crop can be affected by a lot of pathogens like bacterial, fungal, viral, insects and nutrition deficiencies problems. These diseases are easily spreadable which causes big damages in quality and quantity. Some diseases have not visible symptoms and are identified only through sophisticated laboratory analysis, which may be too late to prevent any possible damage in crops. On the other hand, most of these diseases have visible symptoms that can be detected by the expert (plant pathologists) in early stages through visual perception, however, continuous monitoring costs highly in the case of large cultivated surfaces.

The great development and advances in computer vision and neural networks leads to design a technological support (automated system) for plant diseases identification, standing by their visual symptoms, and this will help simple farmers to achieve an accurate plant diseases diagnostics. In [2], two algorithms were used, the first employs visual geometry group (VGG16) as a feature extractor with support vector machine (SVM) as a classifier, the second is the original VGG16 model fine-tuned, to build a classification model. They found that the second model gives better result than the first with a classification accuracy of 89%. Ferentinos [3], authors used 87,848 photographs, captured in the fields. These photographs contain 58 distinct classes and were employed to train: AlexNet, AlexNetOWTBn, GoogLeNet, and VGG, the best result achieved is 99.53% accuracy (0.47% error), by VGG model.

Picon *et al.* [4] shows a new approach to detect three fungual diseases (septoria, tan spot and rust), on wheat images. Based on the previous work by Johannes *et al*. 2017, they develop a new algorithm with an adapted deep residual neural network, in the aime of detecting these three diseases in real time. The results show that the best result was obtained by the new approach: 87% of accuracy compared by the previous work of Johannes *et al*. which was 78%.

Ouhami *et al.* [5] compare three transfer learning models: dense convolutional network (DenseNet), (161 and 121 layers) and VGG16 networks. Their study was based on 666 infected plant leaves images, captured in the fields in Morocco, in addition to the images collected from the internet; all of them were categorized in 6 classes. The best result was 95.65%, obtained by DensNet161. Agarwal *et al.* [6] designed a convolutional neural network (CNN) based on three convolutional-maxpooling layers followed by two fully connected layers, for training process, they used 50,000 images (plant village) for 10 classes, the final results show the high performance of the designed model over VGG16, InceptionV3 and MobileNet pretrained models, with an accuracy of 91.2%. Morgan *et al.* [7], they used artificial neural network (ANN), Naïve Bayes, k-nearest neighbor (KNN), support vector machine (SVM), decision tree and random forest, to classify and predict soybean and mushroom diseases. In mushroom dataset, 8,124 hypothetical samples of 23 species were used to determine if there is a disease or not, the result achieved using all algorithms except Naïve Bayes, is 100%. In soybean dataset, 307 observations with 19 different diseases were used to determine which disease was present; they found that the ANN and KNN classifiers give the best result, which is more than 91% of accuracy.

Zaki *et al.* [8], 4,621 tomato leaf images obtained from PlantVillage dataset was employed to fine tune MobileNet V2 model parameters; and classify four diseases classes, the experimental results show an accurate classification performance, with more than 90% of accuracy. Sladojevic *et al.* [9], the CNN approach was used to classify 13 different types of plant diseases as well as distinguish the leaves of the plant from their surroundings. The experimental results achieve an average accuracy of 86.3%. Aquil and Ishak [10] used CNN method to classify 54,306 images of 26 diseases across 14 species. The dataset was used to train different CNN models (DenseNet 120, ResNet (101, 50, 30, 18), SqueezeNet and VGGNet (19, 16), the best result achieved is 99.68% accuracy, by DenseNet 120. Kawasaki *et al.* [11], they used 800 leaf images to classify two viral diseases that affect cucumber plant: melon yellow spot virus (MYSV) and zucchini yellow mosaic virus (ZYMV). The proposed model achieved an average accuracy of 94.9%. Via tansfer learning with various architectures, [8], [12]–[24], demonstrate that the pre-trained networks pove a strong ability to generalize to new situations (images of plant leaf diseases), by fine-tuning, the pre-existing models.

Given the enormous complexity of the network structures used in the previously cited works, the proposed structure is optimally designed to have the least complex model in terms of network depth (number of layers) [25], [26]. This optimization is not only used to achieve computational efficiency, but also improves the generalizability to classify the diseases. Thus, the main contributions to this work are: i) an automatic tomato plant diseases classification method based on a low complexity CNN architecture, which leads to faster on-line classification; ii) the importance of data augmentation procedure, to avoid overfitting and achieving better results; iii) the influence of hyperparameters to accelerate learning and get a stable model; and iv) the results show a high accuracy in distinguishing a disease from another.

2. METHOD AND MATERIAL

2.1. CNN a theoretical background

The advances in computer vision and machine learning has been constructed and perfected with time, one of the methods that has shown an excellent performance is the CNN method. CNNs are regularized versions of multilayer perceptron (MLP) which is in turn a class of feedforward ANN [27]. In CNN architecture, the vector of weights and bias is called filter, each filter represents a particular feature of the input (e.g., shape in an image).

Figure 1 illustrates how a CNN runs layer by layer in a forward pass, where x^1 is the image (order 3 tensor); w^i parameters involved in the i^{th}layer; x^i the output of the i^{th-1}layer, which also acts as the input to the i^{th}layer; and w^L loss layer. The parameters of the CNN are optimized to minimize the loss z. To get the loss we need to compare the CNN model prediction x^L with the ground truth labels or target corresponding to x^1. On another hand and when the loss z is achieved, we use this supervision signal to update the parameters of the model such (1):

$$(w^i)^{l+1} = (w^i)^l - \eta \frac{\partial z}{\partial (w^i)^l} \tag{1}$$

Where: η is the learning rate and *l* is iteration number, *i* is the layer index [21], [28].

$$x^1 \longrightarrow \boxed{\boldsymbol{w^1}} \longrightarrow x^2 \longrightarrow \dots \longrightarrow x^{L-1} \longrightarrow \boxed{\boldsymbol{w^{L-1}}} \longrightarrow x^L \longrightarrow \boxed{\boldsymbol{w^L}} \longrightarrow z$$

Figure 1. CNN structure, running in a layer by layer; in a forward pass

"In standard gradient descent algorithm, the parameters of the model are updated using one gradient calculated using one training example. The stochastic gradient descent (SGD) algorithm evaluates the gradient and updates the parameters using a subset of the training set, this subset is called mini-batch where every gradient evaluation is an iteration, each iteration is one step for minimizing the loss function, and an epoch is the full pass of the training algorithm over all the training data set using mini-batches" [29], [30]. Note that when we use mini-batch strategy; the CNN input becomes an order 4 tensor (H, W, D, mini-batch size). The stochastic gradient descent algorithm oscillates towards the optimum. We use momentum method to help accelerating the gradient in the right direction and reducing oscillations, by adding a fraction η of the update vector of the past time step to the current update vector:

$$(w^i)^{l+1} = (w^i)^l - \eta \nabla z((w^i)^l) + \gamma((w^i)^l - (w^i)^{l-1}) \tag{2}$$

Where $\boldsymbol{l}$ is the iteration number,$\boldsymbol{\eta} > 0$ is the learning rate,$\boldsymbol{w}$ is the parameter vector, and $\boldsymbol{z}(\boldsymbol{w})$ is the loss function. The gradient of the loss function is $\boldsymbol{\nabla z}(\boldsymbol{w})$. $\boldsymbol{\gamma}$ determines the contribution of the previous gradient step to the current iteration [31], [32].

2.2. Hardware and software environment

The training step requires graphics processing units (GPUs) because of the huge amount of computing. The deep learning was run under an Intel Core i7-7700 CPU @ 3.60GHz system, with the graphic card NVIDIA GeForce GTX 1080 Ti, powered by its GPU. This helped significantly to reduce the execution time, and allowed running many simulation scenarios.

2.2.1. Dataset

Besides the data downloaded from the internet (searched in terms of two keywords: the disease and plant name), the tomato leaf disease images have been taken from the green houses of the CRESTRA research center in Biskra-Algeria; where diseases are induced on tomato plant to be used later for biological experiments. Finally, 1210 images, in red-green-blue (RGB) color space, joint photographic experts' group (JPEG) standard format, for 9 classes shows in Figure 2, Table 1 were obtained, where each class is a disease label. Any tomato leaf image was captured in its own environment, i.e., with a random background.

2.2.2. Image preprocessing

It's important to utilize accurately classified images. We asked an agriculture-expert to examine the leaf images and label them with the appropriate disease acronym. Only higher resolution images with region of interest and containing all the needed information for feature learning were considered, they were resized to 256×256 to make feasible the training and reduce its time.

2.2.3. Augmentation process

We apply augmentation to reduce overfitting during training. It's about enlarging the data set by producing a slight distortion (transformation) to the images: rotation, translation, shearing, scaling and reflection [33]. Thus, we obtain the longer database Table 1.

Figure 2. Data-base classes. (upper left-to-lower right): canker, early blight, late blight, magnesium deficiency, healthy, leaf mould, leaf miner; septoria leaf spot, powdery mildew

Table1. Overview on the plant-diseases database [34]

Class	Number of original images	Total number of images	Leaf symptoms
Early blight	165	8244	Round small dark leaf spots (can be mor than ½ inch in diameter) with yellow tissue around the spots. The spots appear as concentric rings.
Late blight	82	5743	Dry dark brown and irregular spots, surrounded by green edge.
Septoria leaf spot	163	8172	Numerious dark brown circular spots that are the size of a pencil tip or larger. We can see in the center black specks.
Leaf mould	124	7256	Greenish to yellow spots (less than ¼ inch) on the upper leaf surface, and olive-green spots on the leaves upper side.
Leaf miner	149	1094	Irregular lines or tunnels on both sides of the leaf, inside these tunnels we can find a remarkable blotch looks like a black fecal material.
Powdery mildew	278	9667	We see powdery spots on the upper and lower sides of the leaf, in addition to yellow spots.
Bacterial canker	79	5574	At the margin of leaves, appear dark brown lesions, we can find also a round spots.
Magnesium deficiency	100	6307	Generally, plants develop an interveinal chlorosis on tomato plant leaves
Healthy	70	5148	-

2.2.4. Layer configuration and parameters of CNN

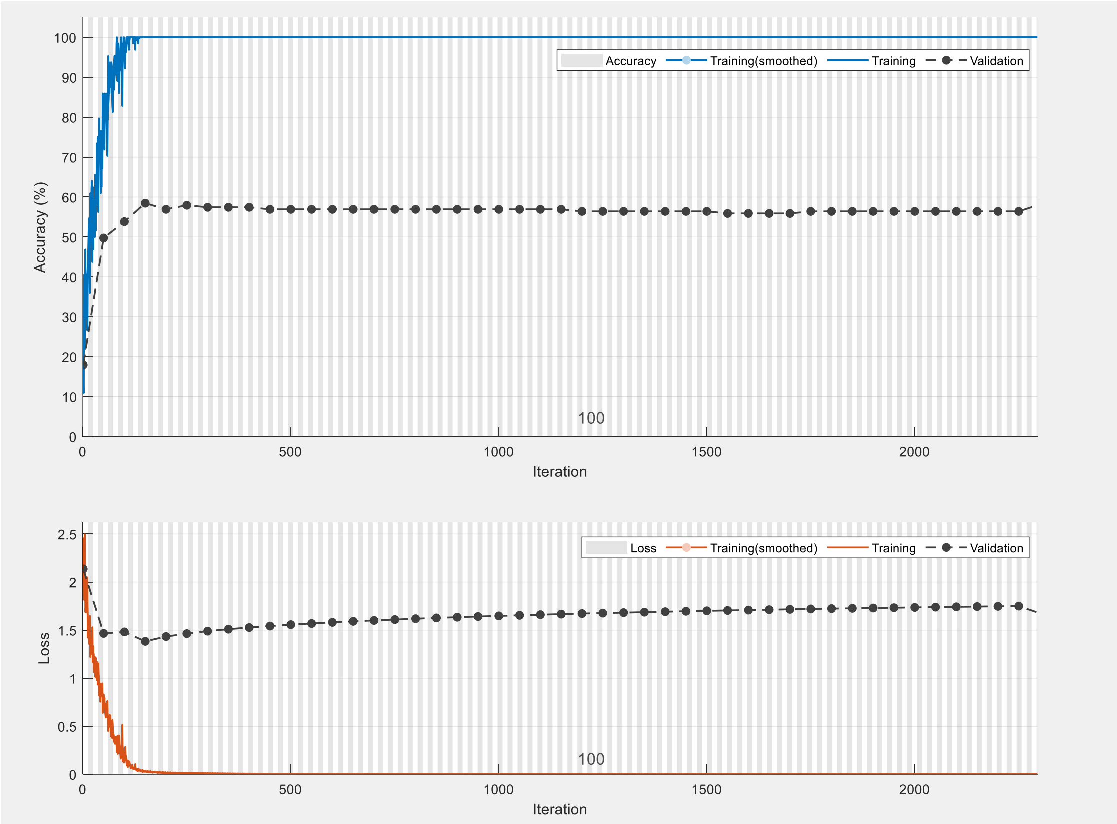
The proposed model consists of three convolution-maxpooling layers, the three last layers are: fully connected layer followed by softmax and classification layers. The model is trained, in Matlab environment, using SGD method with a constant learning rate equal to 0.0007. The network parameters were modified by trial and error. The database was divided into: 80% for training and 20% for validation.

3. RESULT AND DISCUSSION

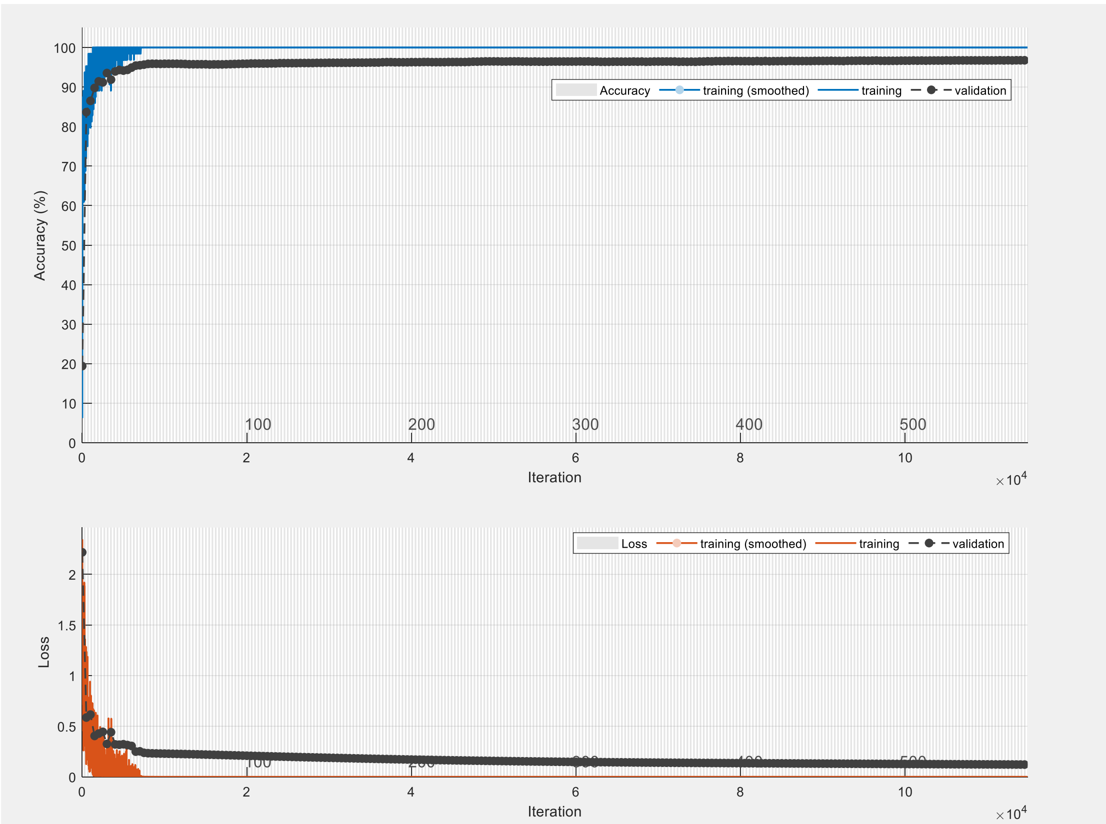
3.1. Data augmentation

We showed earlier the importance of data augmentation and its influence to avoid overfitting which is one of the current problems that occur during learning. In Figure 3, classification accuracy and loss on the training dataset is marked in blue and red for the training dataset; and in black dotted line for the validation dataset. When we use a small dataset (1,210 images), we notice that the accuracy or the error curves obtained on the training set, goes twards 100% of accuracy (or 0.001 error), while the accuracy or error curves obtained on validation, stops improving after a certain number of epochs and begins to be stable twards 58%. This shows that the model works more poorly as Figure 3(a), this is explained by the fact that the training examples were memorized by the network, but it has not the ability to generalize to new cases.

Several ways are used to reduce overfitting, the best of them is to get more training data. When we use a big dataset (57,205 images), that was obtained by applying image data augmentation procedure, we noticed that the model fits well on training dataset and generalizes well to the validation dataset with a high validation accuracy, 97.04%, and a very small error shows in Figure 3(b). In this case we have a good fit model.



(a)



(b)

Figure 3. Accuracy and loss of classification (a) without augmentation and (b) with augmentation process

3.2. Momentum (γ)

Here, we show the influence of momentum on the model performance. In order to gain time, we just used four diseases classification model (powdery mildew, early blight, septoria and leaf mould). In Figure 4 we show line plots for different momentum (γ) values, and compare their effect to the SGD algorithm, with a constant learning rate. Classification accuracy and loss on the training dataset are marked respectively in blue and red, whereas accuracy and loss on the validation dataset are marked in black dotted line. The proposed method shows the ability to learn very well with all momentum values (γ=0, γ=0.5, γ=0.75, γ=0.9), which explains the stability of the model. On the other hand, we see that adding momentum can accelerate training and gives a more stable model.

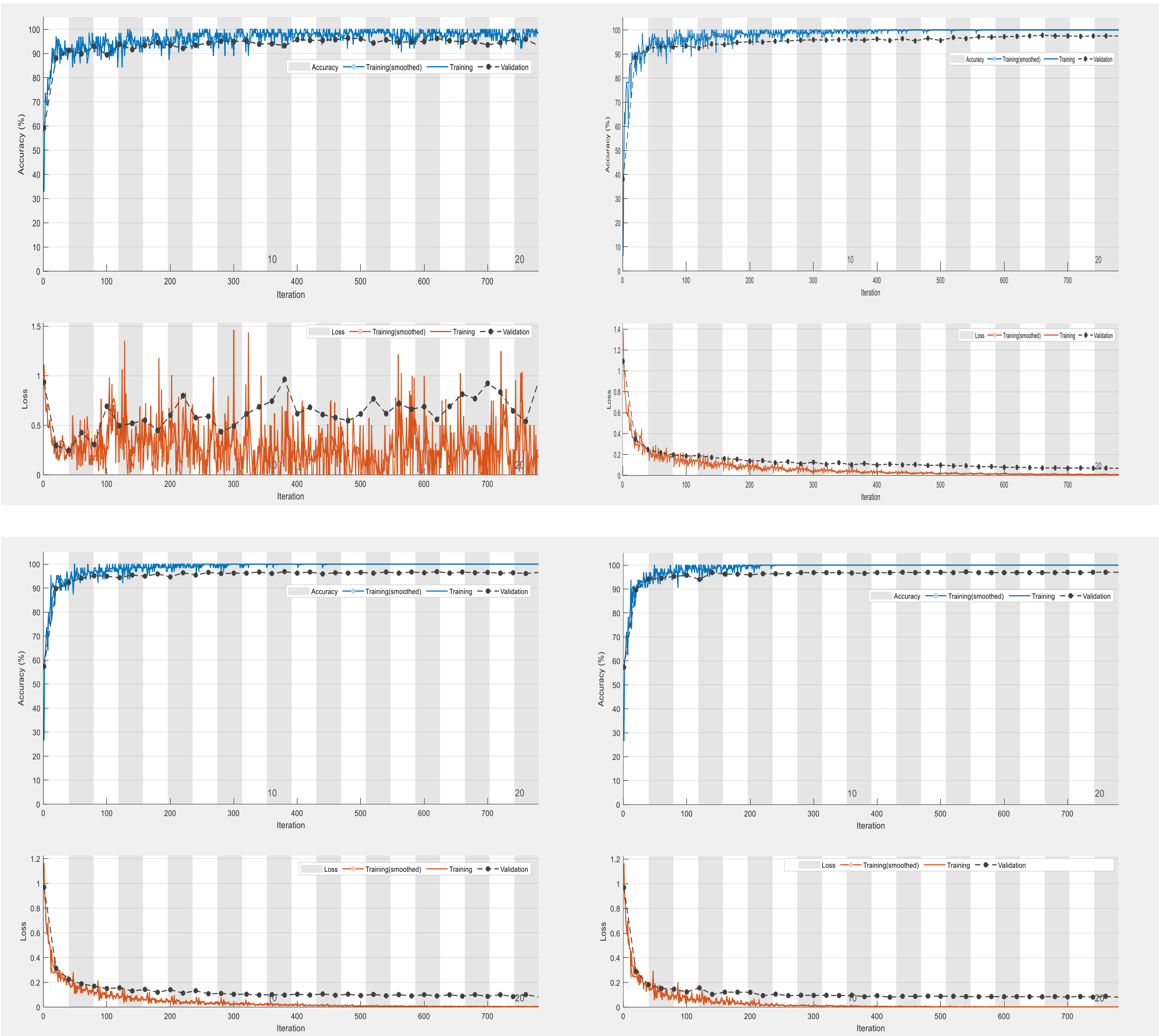


Figure 4. Accuracy and loss of classification with different momentum values, (upper left-to-lower right): γ=0; γ=0.5; γ=0.75; γ=0.9

3.3. Learning rate (η)

The learning rate is a configurable hyperparameter which refers to the amount that the weights are updated during training. Figure 5 shows line plots for different learning rate values (η), and compares its effect on the SGD algorithm, with a momentum=0.9. Classification accuracy and loss on the training dataset are marked in blue and red, whereas accuracy and loss on the validation dataset are marked in black dotted line. We can see for $\eta = 7 \cdot 10^{-1}$, and $\eta = 7 \cdot 10^{-7}$, oscillations, and inability of the model to learn; for $\eta = 7 \cdot 10^{-3}$, $\eta = 7 \cdot 10^{-4}$, and $\eta = 7 \cdot 10^{-5}$, less oscillations, and ability of the model to learn well. The best accuracy and loss obtained with $\eta = 7 \cdot 10^{-4}$. Decreasing the value of η, leads to a slow model.

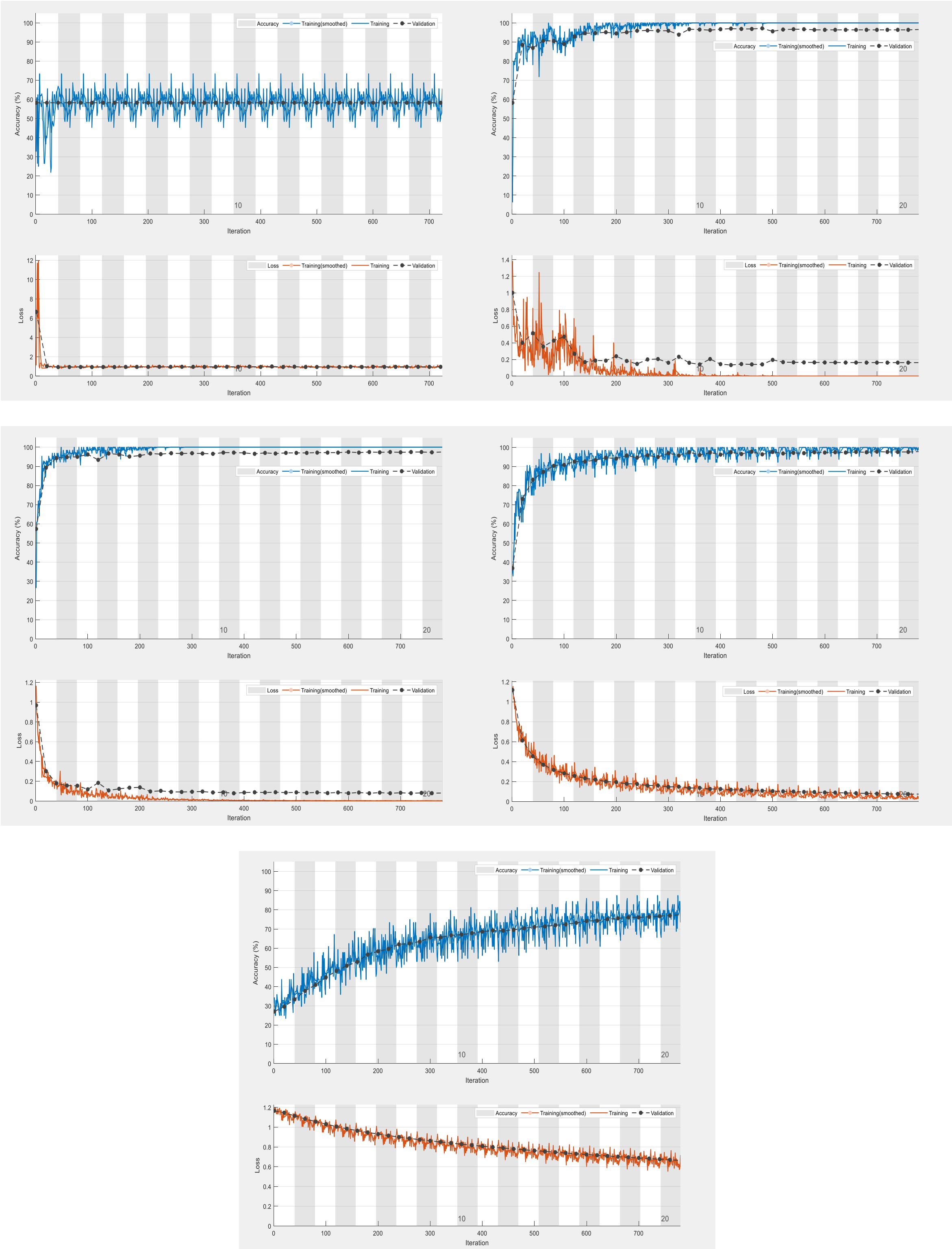


Figure 5. Accuracy and loss of classification with different learning rate values, (upper left-to-lower right): $\eta = 7 \cdot 10^{-1}, \eta = 7 \cdot 10^{-3}, \eta = 7 \cdot 10^{-4}, \eta = 7 \cdot 10^{-5}, \eta = 7 \cdot 10^{-3}$

4. CONCLUSION

In this study we tried to focus on tomato plant diseases detection and classification using leaf images which are captured in the field, or downloaded from the internet, in their environment without background subtraction. The proposed CNN architecture involves only three convolutional layers, which allows a reduced computational complexity, and this is very useful for applications that require fast on-line decision making. Thus, we achieved 97.04% accuracy and less than 0.2 error, by this we meant that one should not be impressed when a complex model fits a dataset well, with enough parameters we can fit any dataset, and it will be easy to reach ~99% accuracy by a continuous tuning process. The first part of experiments shows the importance of data augmentation procedure to avoid overfitting and achieve better results. In the second part of this study, we improved: The considerable effect of learning rate to the stochastic gradient descent algorithm, this hyperparameter should be the first to adjust; and the importance of using SGD algorithm with momentum to accelerate learning and get a more stable model.

REFERENCES

[1] "Report of the Food and Agriculture Organization of the United Nations (FAO)," in *The state of Food Security and Nutrition in the world*, Rome, 2022.

[2] J. Shijie, J. Peiyi, H. Siping, and Sl. Haibo, "Automatic detection of tomato diseases and pests based on leaf images," in *2017 Chinese Automation Congress (CAC)*, Oct. 2017, pp. 2510–2537, doi: 10.1109/CAC.2017.8243388.

[3] K. P. Ferentinos, "Deep learning models for plant disease detection and diagnosis," *Computers and Electronics in Agriculture*, vol. 145, pp. 311–318, Feb. 2018, doi: 10.1016/j.compag.2018.01.009.

[4] A. Picon, A. Alvarez-Gila, M. Seitz, A. Ortiz-Barredo, J. Echazarra, and A. Johannes, "Deep convolutional neural networks for mobile capture device-based crop disease classification in the wild," *Computers and Electronics in Agriculture*, vol. 161, pp. 280–290, Jun. 2019, doi: 10.1016/j.compag.2018.04.002.

[5] M. Ouhami, Y. Es-Saady, M. El Hajji, A. Hafiane, R. Canals, and M. El Yassa, "Deep transfer learning models for tomato disease detection," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2020, pp. 65–73.

[6] 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.

[7] M. Morgan, C. Blank, and R. Seetan, "Plant disease prediction using classification algorithms," *IAES International Journal of Artificial Intelligence (IJ-AI)*, vol. 10, no. 1, p. 257, Mar. 2021, doi: 10.11591/ijai.v10.i1.pp257-264.

[8] S. Z. M. Zaki, M. Asyraf Zulkifley, M. Mohd Stofa, N. A. M. Kamari, and N. Ayuni Mohamed, "Classification of tomato leaf diseases using MobileNet v2," *IAES International Journal of Artificial Intelligence (IJ-AI)*, vol. 9, no. 2, p. 290, Jun. 2020, doi: 10.11591/ijai.v9.i2.pp290-296.

[9] 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, pp. 1–11, 2016, doi: 10.1155/2016/3289801.

[10] M. A. Ihsan Aquil and W. H. Wan Ishak, "Evaluation of scratch and pre-trained convolutional neural networks for the classification of Tomato plant diseases," *IAES International Journal of Artificial Intelligence (IJ-AI)*, vol. 10, no. 2, p. 467, Jun. 2021, doi: 10.11591/ijai.v10.i2.pp467-475.

[11] Y. Kawasaki, H. Uga, S. Kagiwada, and H. Iyatomi, "Basic study of automated diagnosis of viral plant diseases using convolutional neural networks," in *Advances in Visual Computing*, 2015, pp. 638–645.

[12] A. S. Hamzah and A. Mohamed, "Classification of white rice grain quality using ANN: a review," *IAES International Journal of Artificial Intelligence (IJ-AI)*, vol. 9, no. 4, p. 600, Dec. 2020, doi: 10.11591/ijai.v9.i4.pp600-608.

[13] A. Jadon, A. Varshney, and M. S. Ansari, "Low-Complexity high-performance deep learning model for real-time low-cost embedded fire detection systems," *Procedia Computer Science*, vol. 171, pp. 418–426, 2020, doi: 10.1016/j.procs.2020.04.044.

[14] S. Wallelign, M. Polceanu, and C. Buche, "Soybean plant disease identification using convolutional neural network," in *Proceedings of the 31st International Florida Artificial Intelligence Research Society Conference, FLAIRS 2018*, 2018, pp. 146–151.

[15] Z. Iqbal, M. A. Khan, M. Sharif, J. H. Shah, M. H. ur Rehman, and K. Javed, "An automated detection and classification of citrus plant diseases using image processing techniques: A review," *Computers and Electronics in Agriculture*, vol. 153, pp. 12–32, Oct. 2018, doi: 10.1016/j.compag.2018.07.032.

[16] K. Zhang, Q. Wu, A. Liu, and X. Meng, "Can deep learning identify tomato leaf disease?," *Advances in Multimedia*, vol. 2018, pp. 1–10, Sep. 2018, doi: 10.1155/2018/6710865.

[17] M. Brahimi, M. Arsenovic, S. Laraba, S. Sladojevic, K. Boukhalfa, and A. Moussaoui, "Deep learning for plants diseases: detection and saliency map visualisation," in *in Human and Machine Learning*, 2018, pp. 93–117.

[18] H. Durmus, E. O. Gunes, and M. Kirci, "Disease detection on the leaves of the tomato plants by using deep learning," in *2017 6th International Conference on Agro-Geoinformatics*, Aug. 2017, pp. 1–5, doi: 10.1109/Agro-Geoinformatics.2017.8047016.

[19] E. Suryawati, R. Sustika, R. S. Yuwana, A. Subekti, and H. F. Pardede, "eep structured convolutional neural network for tomato diseases detection," in *2018 International Conference on Advanced Computer Science and Information Systems (ICACSIS)*, Oct. 2018, pp. 385–390, doi: 10.1109/ICACSIS.2018.8618169.

[20] 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, Aug. 2018, doi: 10.3389/fpls.2018.01162.

[21] J. G. M. Esgario, P. B. C. de Castro, L. M. Tassis, and R. A. Krohling, "An app to assist farmers in the identification of diseases and pests of coffee leaves using deep learning," *Information Processing in Agriculture*, vol. 9, no. 1, pp. 38–47, Mar. 2022, doi: 10.1016/j.inpa.2021.01.004.

[22] P. Tm, A. Pranathi, K. SaiAshritha, N. B. Chittaragi, and S. G. Koolagudi, "Tomato leaf disease detection using convolutional neural networks," in *2018 Eleventh International Conference on Contemporary Computing (IC3)*, Aug. 2018, pp. 1–5, doi: 10.1109/IC3.2018.8530532.

[23] A. K. Rangarajan, R. Purushothaman, and 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.

[24] V. Singh and A. K. Misra, "Detection of plant leaf diseases using image segmentation and soft computing techniques," *Information Processing in Agriculture*, vol. 4, no. 1, pp. 41–49, Mar. 2017, doi: 10.1016/j.inpa.2016.10.005.

[25] D. Justus, J. Brennan, S. Bonner, and A. S. McGough, "Predicting the computational cost of deep learning models," in *2018 IEEE International Conference on Big Data (Big Data)*, Dec. 2018, pp. 3873–3882, doi: 10.1109/BigData.2018.8622396.

[26] M. H. Shah and X. Dang, "Low-complexity deep learning and RBFN architectures for modulation classification of space-time block-code (STBC)-MIMO systems," *Digital Signal Processing*, vol. 99, p. 102656, Apr. 2020, doi: 10.1016/j.dsp.2020.102656.

[27] S. Bensaadi, “A visual perception system for a mobile robot,” Batna University, Batna, Algeria, 2014.

[28] J. Wu, “Introduction to convolutional neural networks,” *Introduction to Convolutional Neural Networks*. pp. 1–31, 2017.

[29] Bharadwaj, K. B. Prakash, and G. R. Kanagachidambaresan, “Pattern recognition and machine learning,” *Journal of Information Technology Case and Application Research*, vol. 21, no. 2, pp. 105–144, 2021.

[30] Y. Lecun and M. Zeiler, “Visualizing and understanding convolutional networks,” *Computer Vision - Eccv 2014*, vol. 8689, pp. 1–15, 2015.

[31] I. Gitman, H. Lang, P. Zhang, and L. Xiao, “Understanding the role of momentum in stochastic gradient methods,” *Advances in Neural Information Processing Systems*, vol. 32, 2019.

[32] A. Mustapha, L. Mohamed, and K. Ali, “An overview of gradient descent algorithm optimization in machine learning: application in the ophthalmology field,” in *Smart Applications and Data Analysis*, Springer International Publishing, 2020, pp. 349–359.

[33] R. Poojary, R. Raina, and A. Kumar Mondal, “Effect of data-augmentation on fine-tuned CNN model performance,” *IAES International Journal of Artificial Intelligence (IJ-AI)*, vol. 10, no. 1, p. 84, Mar. 2021, doi: 10.11591/ijai.v10.i1.pp84-92.

[34] A. Fuentes, S. Yoon, Y. Lee, H. Youngki, and D. S. Park, “Characteristics of tomato plant diseases, a study for tomato plant disease identification,” in *ISITC International Symposium on Information Technology Convergence*, (Shanghai, China), 2016, pp. 226–230.

BIOGRAPHIES OF AUTHORS

Soumia Bensaadi was born in Batna, Algeria. She received the state-engineer degree in Electronics and Control Systems in 2009; and the M.Sc. degree in Robotics in 2015, from the department of Electronics, University of Batna, Algeria. She is, since 2012, a contract-lecturer with the faculties of (Technology; Maths and Computer Science; HSE), invloved in courses and Labs related to (Computer science; Maths; Robotics; PLCs; Physics). She is presently in the process of completing a Ph.D. degree, in the same field and same university at the Laboratory of Advanced Electronics (LEA). Her research interests include (robotics, control theory and computer vision). She can be contacted at email: s.bensaadi@univ-batna2.dz.

Ahmed Louchene was born in Ain-Touta, Algeria. He received the degree in electronics engineering from the University of Technology USTOran in 1981. He joined the Department of Electrical Engineering, University of Batna, Algeria, in 1981 as an assistant. He received a master’s in microprocessor engineering in 1987 at Bradford University, Great Britain and a PhD in electronics engineering from the University of Batna in 2004. He is currently a full-time lecturer and is the head of the robotic research group at the Laboratory of Advanced Electronics (LEA). His research and teaching interests are in the areas of agricultural production and protection automation, systems control theory. He can be contacted at email: a.louchene@univ-batna2.dz.