



Automatic Detection and Classification of Apple Leaves Diseases Using MobileNet V2

Mohammed Boutalline^{1*}, Adil Tannouche², Hassan Faouzi³, Hamid Ouanan⁴, Malak Dargham⁵

¹ Systems Engineering Laboratory (LGS), National School of Applied Sciences (ENSA-BM), Sultan Moulay Slimane University (USMS), Beni Mellal 23000, Morocco

² Laboratory of Engineering and Applied Technology (LITA), Higher School of Technology (EST-BM), Sultan Moulay Slimane University (USMS), Beni Mellal 23000, Morocco

³ Systems Engineering Laboratory (LGS), Higher School of Technology (EST-FBS), Sultan Moulay Slimane University (USMS), Fkih Ben Salah 23200, Morocco

⁴ Information Processing and Decision Support Laboratory (TIAD), National School of Applied Sciences (ENSA-BM), Sultan Moulay Slimane University (USMS), Beni Mellal 23000, Morocco

⁵ Systems Engineering Laboratory (LGS), Sultan Moulay Slimane University (USMS), Beni Mellal 23000, Morocco

Corresponding Author Email: boutalline@gmail.com

<https://doi.org/10.18280/ria.360512>

ABSTRACT

Received: 19 September 2022

Accepted: 20 October 2022

Keywords:

apple leaves diseases classification, computer vision, artificial intelligence, deep learning, MobileNet V2

Apple orchards in the Imouzzar Kandar region (Morocco) suffer from numerous leaf diseases causing extreme yield losses. Early diagnosis favors the control of these diseases by optimizing the use of chemical products and reducing environmental impacts. In this context, we propose the deployment of an automated detection and classification system for these apple leaf diseases. We have pre-trained the convolutional neural network MobileNet V2 and evaluated its performance across several hyperlearning parameters to recognize the symptoms of the eight most common diseases in the region. The results show that MobileNet V2 is more than 98% effective in identifying these diseases. This encourages us to introduce this valuable tool to farmers looking to improve the quality of their crops.

1. INTRODUCTION

The Region of Fez-Meknes represents a favorable environment for the cultivation of apple trees. Its geographical location hinge between the foot of the Rif, the High and Middle Atlas, gives it an availability of sufficient surface and underground water, suitable soil types and a climate favorable to apple growing. During this decade, the average annual production of apples in the region of Fez-Meknes, including the area of Imouzzar Kandar known for its quality taste, size and good conservation, has experienced an expansion of 30%, exceeding the mark of 50 thousand tons.

This production expansion was accompanied, also, by an expansion of phytosanitary problems, pests and other diseases encountered by apple growers. The situation has deteriorated further following the adverse influence of climate change on the biology of plant and animal species. Production has suffered significant damage as a result. The impacts range from an unattractive aesthetic appearance and poor fruit quality to a decrease in yield or even complete loss of fruit or trees. Thus, they hinder marketing and cause huge economic losses [1].

Nowadays, apple orchards have a high density of plantations. As a result, pathogens spread rapidly and can destroy entire orchards [2].

The current state of the art is replete with risk prediction models for apple leaf pests and diseases. There are still several treatment programs depending on the severity, incidence and period of infection taking into account the weather conditions [1].

However, early detection promotes effective control of these diseases and pests [3]. While a late or incorrect diagnosis leads to a rapid spread of these diseases, improper use of phytosanitary products, increased production costs, impacts on the environment and human health.

Thus, automatic recognition and classification of plant diseases can help apple growers fight diseases in a timely manner and improve the productivity and quality of their apple orchards.

The rest of this paper is structured as follows. In Section 2, after presenting the problem and specifics related to our area, we present related work and our previous work in terms of using Deep Learning technology. Then, we introduce the material and the proposed method in Section 3, followed by the results and discussions in Section 4. Finally, Section 5 concludes the paper and presents our future work.

2. RELATED WORKS

Nowadays, machine learning and digital imaging have unveiled a significant interest in artificial intelligence. Deep learning is a technique based on the neural network model and is currently enjoying great success in several research fields.

In the field of precision agriculture, we evaluated the impact of this new technology in weed detection by analyzing and comparing the effectiveness of several convolutional neural networks [4]. The results obtained allowed us to improve the speed and efficiency of our previous solution, based on the combined use of Haar pseudo-features and the AdaBoost

algorithm, for real-time weed detection [5] and selection [6]. Then, we successfully used it to detect the weed and locate its location in the plot accurately [7].

In the textile industry, we have used this same technique to recognize and identify fabric defects in real time [8].

In the field of plant diseases identification, several researchers have relied on machine learning algorithms known in the literature, namely: random forests, k-nearest neighbors or even wide-margin separators (SVM) [9-18], to automatically diagnose diseases and improve test times and accuracy of results. Other researchers have used convolutional neural networks (CNNs) to detect disease in crops [19-26]. These studies have shown that the use of CNNs reduces the demand for image preprocessing and improves detection accuracy.

In terms of leaf disease detection, the authors [27] used a CNN-based approach for the detection and distinction of speckle-infected leaves. The AlexNet and GoogLeNet CNNs have been used to detect plant diseases from images [19]. A mobile, interactive and semi-automated application has been developed to separate diseased tissue from healthy tissue [28]. A computer vision system named Cercospora leaf spot (CLS) Rater, was developed to assess the degree of CLS infection in sugar beet crop [29]. Another system has been studied to detect and evaluate disease symptoms in a variety of leaf shapes and sizes [30].

In contrast, according to the study [31], all these above-mentioned methods have difficulty in detecting a specific disease among a large number of symptoms due to different overlapping biotic and abiotic stresses.

Especially, in apple disease detection and classification. The combination of color, texture and shape features on apple images and multi-class wide-margin separators (MSVM) [32] gave better results.

On the other hand, the authors [33] trained a CNN convolutional neural network to detect and identify four common types of apple leaf diseases namely Mosaic, Rust, Brown spot and Alternaria leaf spot. The results obtained prove that this leaf disease identification approach using a CNN gives an overall accuracy of 97.62%.

In this article, we studied the influence of different training parameters, of the convolutional neural network MobileNet V2, for the classification of eight classes of biotic and abiotic diseases of common apple leaves in our region, namely: Scob, Alternaria, Powdery, MLB, Mosaic, Multiple, Necrosis and Insect.

The choice of this CNN is justified by its compactness, its speed and its lightness displayed in our previous study [4] and the study conducted on leaf disease classification of tomato using MobileNet v2 [34]. Thus, this lightweight model of CNN can be implemented in a low-cost board like the Raspberry board and also in the most widely used entry-level smartphones.

3. MATERIAL AND METHOD

In this work, we used a Dell laptop equipped with an Intel Core i5-1145G7 @ 1.5 GHz up to 2.60 GHz processor with 16 GB of DDR4-3200 MHz RAM.

Moreover, we used TensorFlow implemented on the Python platform requiring no GPU. Thus, we were able to test several MobileNet V2 hyperparameter configurations separately, namely: The batch size, the different optimizers and the learning rate, to find their optimal setting.

3.1 Presentation of MobileNet CNN

The MobileNet [35] is a scaled-down CNN (Figure 1), designed specifically for mobile applications and devices. The basic idea behind its success is the replacement of the classical convolution by two new distinct processes namely depthwise and pointwise separable convolution (Figure 1).

3.1.1 Depthwise convolution

Deep convolution is a subversion of classical convolution [36]. Thus, the standard convolution process is distributed over several steps depending on the number of channels in the image. Then, it is applied separately on each channel, ie the depth. The kernel size is also reduced according to the same grouping.

Similarly, the spatial characteristics are collected separately which leads to the reduction of the necessary parameters. Depthwise convolution does not increase image depth. The major difference between 2D convolution and depthwise convolution is that the former is performed on all or several input channels, whereas in depthwise convolution each channel is processed separately according to the following steps:

- (1) Input tensor of 3 dimensions is split into separate channels;
- (2) For each channel, the input is convolved with a filter (2D);
- (3) The output of each channel is then stacked together to get the output on the entire 3D tensor.

3.1.2 Pointwise convolution

Unlike depth convolution, point convolution sets the height and width of the filter to 1 [37], while the depth remains based on the input channels. This convolution immediately follows the depth convolution to form the total convolution while keeping few parameters.

The Pointwise Convolution consists of combining the outputs of the Depthwise Convolution, it is also called 1×1 convolution.

It performs a convolution on the three outputs of the Depthwise Convolution simultaneously. By using N kernels, we obtain at the output of the Pointwise Convolution an image of depth N. The pointwise convolution increases the depth of the image. It can be used to dimension the characteristics of the output channels.

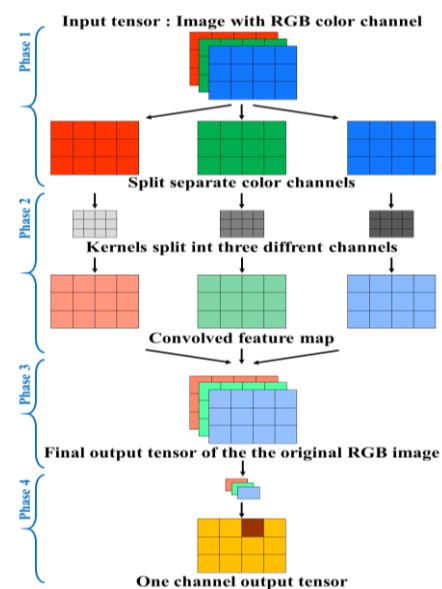


Figure 1. Depth-separable convolutions diagram

This approach allows MobileNet to benefit from a smaller size and higher performance compared to others CNNs (Table 1). Thanks to these performances, MobileNet is an ideal CNN for embedded systems.

Table 1. Performance comparison between some CNNs

Model	ImageNet Accuracy	Million Multi-Adds	Million Parameters
1.0 MobileNet-244	70.6%	569	4.2
GoogleNet	69.8%	1550	6.8
VGG 16	71.5%	15300	138

MobileNet V2 is its improved version introduced in 2018 [38] (Figure 2). Thanks to the improvement of the linear bottleneck and the inverted residual block, the size has been further reduced and the number of parameters has been reduced from 4.2 to 3.4 M while keeping the same performance as the first version.

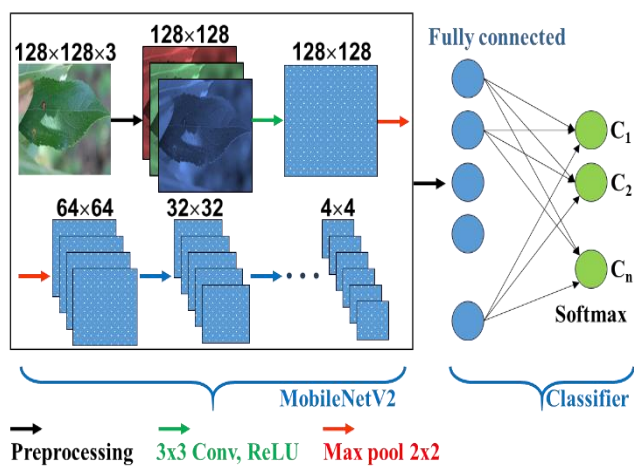


Figure 2. MobileNet V2 architecture

3.2 The MobileNet V2 approach

We trained convolution neural network “MobileNet V2” to detect and classify several apple leaf diseases (Figure 3). The training was performed on a dataset consisting of 3642 high quality images of different apple leaves [31] available free of charge at: <https://www.kaggle.com/competitions/plant-pathology-2020-fgvc7/>.

In addition, we used other images captured manually from different angles with varying illumination, surface and noise. Some images show healthy leaves, others show leaves showing actual symptoms of apple leaf disease, while the last category of images shows multiple diseases on a single leaf.

In general, we have defined nine classes for apple leaves, namely: Healthy, Scob, Alternaria, Powdery, MLB, Mosaic, Necrosis, Insect and Multiple diseases.

First, we annotated these images using the valuable help of local experts to determine the nine classes of apple leaves (Figures 4). Then we augmented the dataset by randomly adjusting the brightness, contrast, rotation and flipping the image processing software (IrfanView version 4.54).

Accordingly, we extracted 18000 images with 2000 images for each of the nine classes. All these images have been resized to 224×224 pixels according to the MobileNet V2 model input format. Then, these images were randomly divided into three groups relating to training, testing and validation.

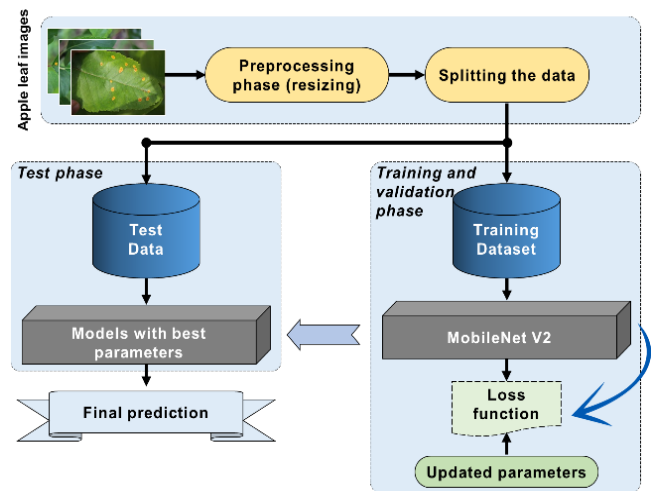


Figure 3. MobileNet approach

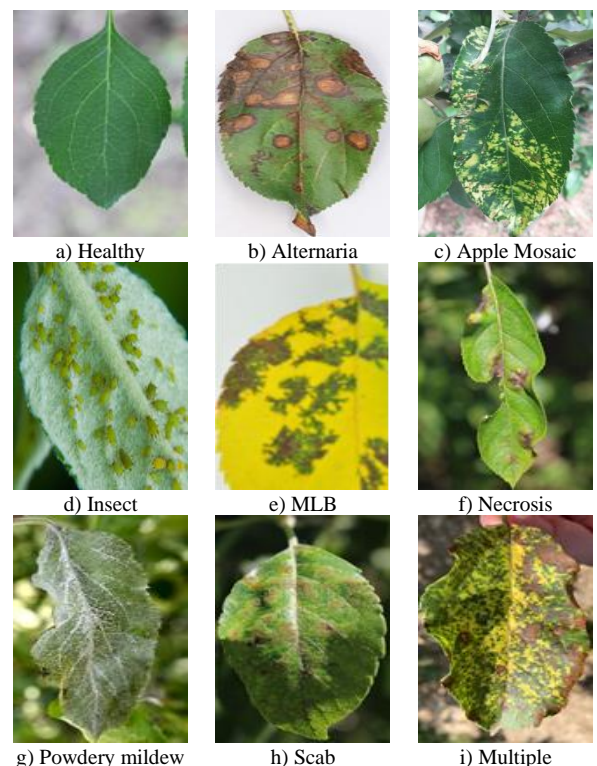


Figure 4. Apple leaf diseases

Finally, we fine-tuned the convolution neural network “MobileNet V2”. The performance in terms of detection and classification of apple leaf diseases is given by the measured average accuracy (1) according to the study [39].

$$accuracy = \frac{\sum_{i=1}^k \frac{TP_i + TN_i}{TP_i + TN_i + FP_i + FN_i}}{k} \quad (1)$$

The performance evaluation of a multiclass classification is done in the same way as a simple binary classification composed of two classes namely: "positive" and "negative". Metrics are obtained from the following five measures:

k: Total number of classes;

TP: True Positives: Number of positive elements correctly predicted;

FP: False positives: Number of negative elements incorrectly predicted to be positives;

TN: True Negatives: number of negative elements correctly predicted;

FN: False Negatives: Number of positives elements incorrectly predicted to be negative.

4. RESULTS AND DISCUSSION

In this part, we perform an analysis of the effectiveness of apple leaf disease detection using the CNN MobileNet V2. According to several hyper learning parameters, namely:

- The optimizers;
- The learning rate;
- The data subset ratios;
- The batch size.

4.1 Influences of optimizers on classification performance

When training a deep learning model, the weights are changed and the loss function is minimized at each epoch. An optimizer is a function or algorithm that modifies the attributes of the neural network, such as weights and learning rate. Thus, it helps to reduce overall loss and improve accuracy.

The problem of choosing the right weights for the model is a daunting task because a deep learning model usually consists of millions of parameters. This raises the need to choose an optimization algorithm well suited to each application.

In this step, we studied the influence of the following optimizers: Adagrad [37], Adam [38], SGD [40], RMSprop and Nadam [41-42], on the classification of apple leaf diseases. The other hyperparameters remain fixed, namely:

- Batch size=32;
- Learning rate=0.1;
- Data subset ratios=9:1.

The results (Figure 5) show that the Adagrad optimizer obtained the best score with an accuracy of 95%. Next comes Adam with an accuracy of 90% and SGD with 87.32%. While the RMSProp and Nadam optimizers achieved very low accuracy compared to the previous ones (Table 2).

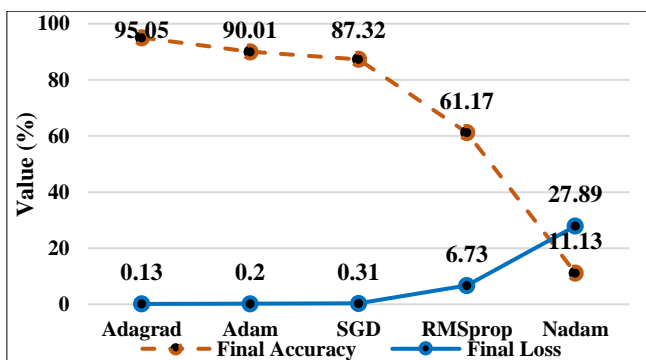


Figure 5. Influences of optimizers

Table 2. Final accuracy and loss of used optimizers

Optimizers	Final Accuracy	Final Loss
Adagrad	0.95	0.1554
Adam	0.90	0.2587
SGD	0.87	0.4306
RMSprop	0.61	4.3899
Nadam	0.11	20.5444

4.2 Influences of learning rate on classification performance

The learning rate is a multiplicative factor applied to the gradient in order to vary the gain of the gradient. The gradient descent algorithm multiplies the learning rate by the gradient at each iteration. The product thus generated is called gradient gain.

Thus, the learning rate is a hyperparameter that plays on the speed of the gradient descent: a greater or lesser number of iterations is necessary before the algorithm converges, that is to say that an optimal learning of the network is achieved.

In this step, we varied the learning rate for each optimizer in the previous step. We have noticed that the Adam optimizer is the one that best responds to the variation of this parameter, while the other optimizers remain less sensitive to the variation of the learning rate in the domain of variation chosen by this study. The other hyperparameters remain fixed, namely:

- Batch size=32;
- Data subset ratios=9:1.

We have chosen to test the following values for this parameter.

Learning rate = {0.1, 0.01, 0.001, 0.0001, 0.00001}.

The results (Figure 6) show that the value 0.001 of the learning rate achieved the best accuracy of 89.79% (Table 3).

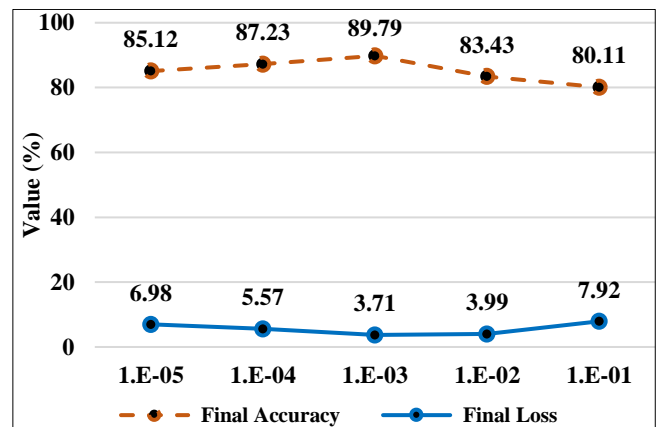


Figure 6. Influences of learning rate

Table 3. Final accuracy and loss of used learning rates

Learning Rate	Final Accuracy	Final Loss
0.00001	85.12	6.98
0.0001	87.23	5.57
0.001	89.79	3.71
0.01	83.43	3.99
0.1	80.11	7.92

4.3. Influences of data subset ratios on classification performance

A successful learning model depends above all on quality data: It is therefore necessary to pre-process the data collected in order to extract its full potential.

The dataset translates to set or collection of data. It is a coherent set of data that can be presented in different formats: numerical data, text, video, image or even sound.

The dataset is a cornerstone of machine learning. It will be used to teach a model to perform a task or make a prediction.

The training dataset is divided into three subsets namely training subset used for MobileNet V2 training, test subset for

training performance evaluations and validation subset, which is used to validate the result of the training.

In this step, we changed the ratios of the training subset and the test subset. Inspired by Kadim et al. [42], four ratios namely, 9:1; 4:1; 7:3 and 3:2 is used. The other hyperparameters remain fixed, namely:

- Batch size=32;
- Learning rate=0.1;
- Optimizer=Adam.

The classification results (Figure 7) showed that the 4:1 ratio achieved the best accuracy of 95.33% (Table 4).

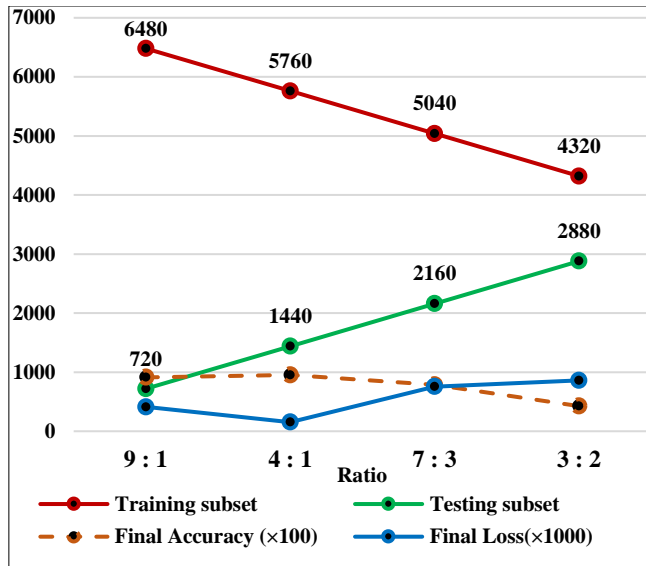


Figure 7. Influences of data subset ratios

Table 4. Final accuracy and loss of used data subset ratios

Training	Testing	ratio	Final Accuracy	Final Loss
6480	720	9:1	91.27	0.4130
5760	1440	4:1	95.33	0.1535
5040	2160	7:3	78.55	0.7551
4320	2880	3:2	42.61	0.8613

Figure 7 shows that the rate of data sets can have a remarkable influence on model performance. The greatest influence was observed for the Adam optimizer. However, the Adagrad optimizer, selected in section 4.1, showed a minimal difference of about 1% in accuracy. The best accuracy obtained was 98% with the 4:1 rate.

4.4 Influences of batch size on classification performance

The batch size is the number of samples sent to the network for one training iteration. This approach allows multiple images to be analyzed instead of just one at a time, which reduces fluctuation in training error. On the other hand, a value that is too large causes an excessive generalization of the model formed.

This hyperparameter is usually fixed during the learning and inference processes. However, TensorFlow supports dynamic lot sizes.

In this step, we evaluated the performance of the classification according to three values of lot sizes namely, 16, 32 and 48. The other hyperparameters remain fixed, namely:

- Data subset ratios=9:1;

- Learning rate=0.1;
- Optimizer=Adagrad.

The results (Figure 8) show that the value 16 gives the best precision of 95.11% (Table 5).

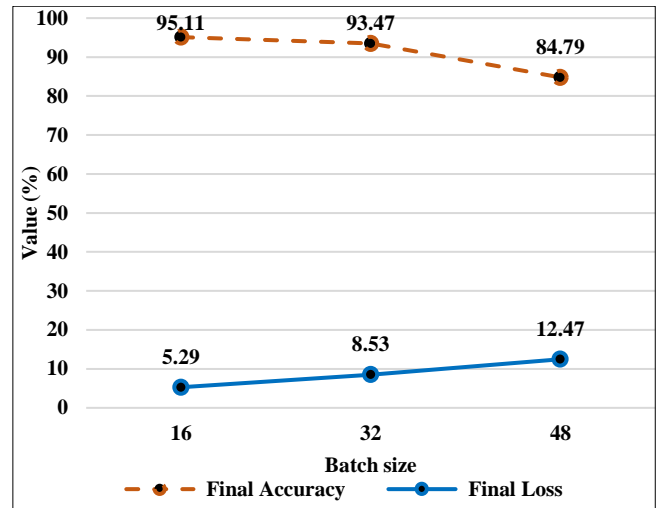


Figure 8. Influences of batch sizes

Table 5. Final accuracy and loss of used batch sizes

Batch Size	Final Accuracy	Final Loss
16	95.11	0.1229
32	93.47	0.4153
48	84.79	0.7012

Similarly, Figure 8 shows that the Batch Size hyperparameter can have a remarkable influence on the model performance. The greatest influence was observed for the following values (Data subset ratios = 9:1, Learning rate = 0.1 and Optimizer = Adagrad).

On the other hand, taking into account the optimal values from the previous sections (Data subset ratios = 4:1, Learning rate = 0.001 and Optimizer = Adagrad) the best accuracy obtained was 98% with a batch size = 16.

5. CONCLUSIONS

MobileNet V2 shows better performance in apple leaf disease classification by combining the Adagrad optimizer with a batch size of 16. More accurate results can be obtained by choosing a learning rate of 0.001 and a 4:1 data subset ratio obtained between the number of training images and the number of test images. Thus, the accuracy obtained is greater than 98%.

We conclude that MobileNet V2 can successfully detect and classify various apple leaf diseases in our region. Given these results, MobileNet V2 remains our team's favorite CNN for embedded applications due to its high performance and compact size.

For future work, we plan to integrate it into an android application to detect and classify all leaf and fruit diseases in the region. This will allow arborists to benefit from the latest technical advances in order to detect diseases early, treat them at the right time and improve the yield and quality of their products accordingly.

REFERENCES

- [1] Lahlali, R., Boulif, M., Moimina, A. (2021). Pratiques phytosanitaires des pomiculteurs: Cas de la région Fès-Meknès. *Revue Marocaine des Sciences Agronomiques et Vétérinaires*, 9(2).
- [2] Peil, A., Bus, V.G., Geider, K., Richter, K., Flachowsky, H., Hanke, M.V. (2009). Improvement of fire blight resistance in apple and pear. *Int J Plant Breed*, 3(1): 1-27.
- [3] Bessin, R.T., McManus, P.S., Brown, G.R., Strang, J.G. (1998). *Midwest tree fruit pest management handbook*. Univ. Ky., Lexington.
- [4] Tannouche, A., Gaga, A., Boutalline, M., Belhouideg, S. (2022). Weeds detection efficiency through different convolutional neural networks technology. *International Journal of Electrical and Computer Engineering*, 12(1): 1048. <http://doi.org/10.11591/ijece.v12i1.pp1048-1055>
- [5] Tannouche, A., Sbai, K., Rahmoune, M., Agounoune, R., Rahmani, A., Rahmani, A. (2016). Real time weed detection using a boosted cascade of simple features. *International Journal of Electrical & Computer Engineering*, 6(6). <http://doi.org/10.11591/ijece.v6i6.pp2755-2765>
- [6] Tannouche, A., Sbai, K., Rahmoune, M., Zoubir, A., Agounoune, R., Saadani, R., Rahmani, A. (2016). A fast and efficient shape descriptor for an advanced weed type classification approach. *International Journal of Electrical and Computer Engineering*, 6(3): 1168. <http://doi.org/10.11591/ijece.v6i3.pp1168-1175>
- [7] Habib, M., Tannouche, A., Ounejjar, Y. (2016). Weed detection in pea cultivation with the faster RCNN ResNet 50 convolutional neural network. *Revue d'Intelligence Artificielle*, 36(1): 13-18. <https://doi.org/10.18280/ria.360102>
- [8] Beljadid, A., Tannouche, A., Balouki, A. (2022). Automatic fabric defect detection employing deep learning. *International Journal of Electrical & Computer Engineering*, 12(4). <http://doi.org/10.11591/ijece.v12i4.pp4129-4136>
- [9] Es-saady, Y., El Massi, I., El Yassa, M., Mammass, D., Benazoun, A. (2016). Automatic recognition of plant leaves diseases based on serial combination of two SVM classifiers. In 2016 International Conference on Electrical and Information Technologies (ICEIT), pp. 561-566. <https://doi.org/10.1109/EITech.2016.7519661>
- [10] Wang, G., Sun, Y., Wang, J. (2017). Automatic image-based plant disease severity estimation using deep learning. *Computational intelligence and neuroscience*, 2017: 2917536. <https://doi.org/10.1155/2017/2917536>
- [11] Padol, P.B., Yadav, A.A. (2016). SVM classifier based grape leaf disease detection. In 2016 Conference on advances in signal processing (CASP), pp. 175-179. <https://doi.org/10.1109/CASP.2016.7746160>
- [12] Gavhale, K.R., Gawande, U. (2014). An overview of the research on plant leaves disease detection using image processing techniques. *IOSR Journal of Computer Engineering (IOSR-JCE)*, 16(1): 10-16. <http://dx.doi.org/10.9790/0661-16151016>
- [13] Sannakki, S.S., Rajpurohit, V.S., Nargund, V.B., Kulkarni, P. (2013). Diagnosis and classification of grape leaf diseases using neural networks. In 2013 Fourth International Conference on Computing, Communications and Networking Technologies (ICCCNT), pp. 1-5. <https://doi.org/10.1109/ICCCNT.2013.6726616>
- [14] Dhakate, M., Ingole, A.B. (2015). Diagnosis of pomegranate plant diseases using neural network. In 2015 fifth national conference on computer vision, pattern recognition, image processing and graphics (NCVPRIPG), pp. 1-4. <https://doi.org/10.1109/NCVPRIPG.2015.7490056>
- [15] Qin, F., Liu, D., Sun, B., Ruan, L., Ma, Z., Wang, H. (2016). Identification of alfalfa leaf diseases using image recognition technology. *PLoS One*, 11(12): e0168274. <https://doi.org/10.1371/journal.pone.0168274>
- [16] Gupta, T. (2017). Plant leaf disease analysis using image processing technique with modified SVM-CS classifier. *Int. J. Eng. Manag. Technol*, 5, 11-17.
- [17] Rothe, P.R., Kshirsagar, R.V. (2015). Cotton leaf disease identification using pattern recognition techniques. In 2015 International conference on pervasive computing (ICPC), pp. 1-6. <https://doi.org/10.1109/PERVASIVE.2015.7086983>
- [18] Islam, M., Dinh, A., Wahid, K., Bhowmik, P. (2017). Detection of potato diseases using image segmentation and multiclass support vector machine. In 2017 IEEE 30th canadian conference on electrical and computer engineering (CCECE), pp. 1-4. <https://doi.org/10.1109/CCECE.2017.7946594>
- [19] Mohanty, S.P., Hughes, D.P., Salathé, M. (2016). Using deep learning for image-based plant disease detection. *Frontiers in Plant Science*, 7: 1419. <https://doi.org/10.3389/fpls.2016.01419>
- [20] Lu, Y., Yi, S., Zeng, N., Liu, Y., Zhang, Y. (2017). Identification of rice diseases using deep convolutional neural networks. *Neurocomputing*, 267: 378-384. <https://doi.org/10.1016/j.neucom.2017.06.023>
- [21] Fuentes, A., Yoon, S., Kim, S.C., Park, D.S. (2017). A robust deep-learning-based detector for real-time tomato plant diseases and pests recognition. *Sensors*, 17(9): 2022. <https://doi.org/10.3390/s17092022>
- [22] Mohanty, S.P., Hughes, D., Salathe, M. (2016). Inference of plant diseases from leaf images through deep learning. *Front. Plant Sci*, 7: 1419. <https://doi.org/10.3389/fpls.2016.01419>
- [23] Kawasaki, Y., Uga, H., Kagiwada, S., Iyatomi, H. (2015). Basic study of automated diagnosis of viral plant diseases using convolutional neural networks. In International symposium on visual computing, pp. 638-645. https://doi.org/10.1007/978-3-319-27863-6_59
- [24] Sladojevic, S., Arsenovic, M., Anderla, A., Culibrk, D., Stefanovic, D. (2016). Deep neural networks based recognition of plant diseases by leaf image classification. *Computational intelligence and neuroscience*, 2016: 3289801. <https://doi.org/10.1155/2016/3289801>
- [25] Tan, W.X., Zhao, C.J., Wu, H.R. (2016). CNN intelligent early warning for apple skin lesion image acquired by infrared video sensors. *High Technol. Lett.* 22: 67-74. <https://doi.org/10.3772/j.issn.1006-6748.2016.01.010>
- [26] Hanson, A.M.G.J., Joel, M.G., Joy, A., Francis, J. (2017). Plant leaf disease detection using deep learning and convolutional neural network. *International Journal of Engineering Science*, 5324: 2-4.
- [27] Amara, J., Bouaziz, B., Algergawy, A. (2017). A deep learning-based approach for banana leaf diseases classification. *Datenbanksysteme für Business, Technologie und Web (BTW 2017)-Workshopband*.
- [28] Pethybridge, S.J., Nelson, S.C. (2015). Leaf doctor: A

- new portable application for quantifying plant disease severity. *Plant disease*, 99(10): 1310-1316. <https://doi.org/10.1094/PDIS-03-15-0319-RE>
- [29] Atoum, Y., Afridi, M.J., Liu, X., McGrath, J.M., Hanson, L.E. (2016). On developing and enhancing plant-level disease rating systems in real fields. *Pattern Recognition*, 53: 287-299. <https://doi.org/10.1016/j.patcog.2015.11.021>
- [30] Barbedo, J.G.A. (2014). An automatic method to detect and measure leaf disease symptoms using digital image processing. *Plant Disease*, 98(12): 1709-1716. <https://doi.org/10.1094/pdis-03-14-0290-re>
- [31] Thapa, R., Zhang, K., Snively, N., Belongie, S., Khan, A. (2020). The plant pathology challenge 2020 data set to classify foliar disease of apples. *Applications in Plant Sciences*, 8(9)L e11390. <https://doi.org/10.1002/aps3.11390>
- [32] Dubey, S.R., Jalal, A.S. (2016). Apple disease classification using color, texture and shape features from images. *Signal, Image and Video Processing*, 10(5): 819-826. <https://doi.org/10.1007/s11760-015-0821-1>
- [33] Liu, B., Zhang, Y., He, D., Li, Y. (2017). Identification of apple leaf diseases based on deep convolutional neural networks. *Symmetry*, 10(1): 11. <https://doi.org/10.3390/sym10010011>
- [34] Zaki, S.Z.M., Zulkifley, M.A., Stofa, M.M., Kamari, N.A.M., Mohamed, N.A. (2020). Classification of tomato leaf diseases using MobileNet v2. *IAES International Journal of Artificial Intelligence*, 9(2): 290. <http://doi.org/10.11591/ijai.v9.i2.pp290-296>
- [35] Howard, A.G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Adam, H. (2017). Mobilenets: Efficient convolutional neural networks for mobile vision applications. arXiv preprint arXiv:1704.04861. <https://doi.org/10.48550/arXiv.1704.04861>
- [36] Yoo, B., Choi, Y., Choi, H. (2018). Fast depthwise separable convolution for embedded systems. In *International Conference on Neural Information Processing*, pp. 656-665. https://doi.org/10.1007/978-3-030-04239-4_59
- [37] Duchi, J., Hazan, E., Singer, Y. (2011). Adaptive subgradient methods for online learning and stochastic optimization. *Journal of machine learning research*, 12(7).
- [38] Kingma, D.P., Ba, J. (2014). Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980. <https://doi.org/10.48550/arXiv.1412.6980>
- [39] Grandini, M., Bagli, E., Visani, G. (2020). Metrics for multi-class classification: An overview. arXiv preprint arXiv:2008.05756.
- [40] Ruder, S. (2016). An overview of gradient descent optimization algorithms. arXiv preprint arXiv:1609.04747. <https://doi.org/10.48550/arXiv.1609.04747>
- [41] Dozat, T. (2016). Incorporating nesterov momentum into adam. <https://openreview.net/>, accessed on Jun. 24, 2022.
- [42] Kadim, Z., Zulkifley, M.A., Hamzah, N. (2020). Deep-learning based single object tracker for night surveillance. *International Journal of Electrical & Computer Engineering*, 10(4). <http://doi.org/10.11591/ijece.v10i4.pp3576-358>