MACHINE LEARNING ALGORITHMS COMPARISON FOR IMAGE CLASSIFICATION ON ANTHRACNOSE INFECTED WALNUT TREE CANOPIES

Athanasios Anagnostis[1,2], Gavriela Asiminari[1], Georgios Dolias[1], Christos Arvanitis[1], Elpiniki Papageorgiou[1,3], Charalambos Myresiotis[1], and Dionysis Bochtis[1]

[1]Institute for Bio-economy and Agri-technology, Center for Research and Technology Hellas, Greece

[2]University of Thessaly, Department of Computer Science, Lamia, Greece

[3]University of Thessaly, Faculty of Technology, Larisa, Greece

a.anagnostis@certh.gr, athananagno@uth.gr, g.asiminari@certh.gr, g.dolias@certh.gr, c.arvanitis@certh.gr, e.papageorgiou@certh.gr, epapageor@cs.uth.gr, c.miresiotis@certh.gr, d.bochtis@certh.gr

ABSTRACT

Fungal diseases such as anthracnose, can be catastrophic to crops worldwide because it can destructively damage the canopies of trees and can also spread easily to nearby trees. Copper spaying, adequate pruning and proper sanitation, renders the treatment of such diseases as easy, however, the main concern in such cases is the spreading prevention by early detection systems. This can be dealt with automated procedures offered in precision agriculture such as automatic image collection and real-time classification by smart systems. Purpose of this study is to compare the most famous ML algorithms for classification, in order to investigate the applicability and effectiveness of an image-based classifier on anthracnose infected canopies. Various machine learning algorithms were employed, tested, evaluated and compared based on their abilities and limitations. The comparison is conducted based on several performance metrics and finally, the applicability of the best performing architecture is discussed for real-life applications.

Keywords*:* machine learning, image classification, anthracnose, walnut trees

1. INTRODUCTION

In agriculture, fungal infections can be catastrophic for entire crops, leading to diminished, and even destroyed yields. This problem affects directly the income of the farmers, and even though the disease can be treated easily, the aim is to prevent its spreading before it infects large areas. Autonomous systems that can identify the infection without any human input, can solve this problem by monitoring the status of each tree in the field, by collecting images of it. This approach can be built on machine learning algorithms (ML) which is used for this kind of applications (Bharate and Shirdhonkar, 2018; Liakos *et al.*, 2018).

Machine learning is a wide area of self-training algorithms that can learn to do tasks such as image classification. Several types of infections have been investigated for many types of plant leaves, leading to diverse approaches on image-based plant disease classification. Artificial neural networks (ANN), radial basis function networks (RBF) and learning vector quantization (LQV) were used by (Muthukannan *et al.*, 2015) achieving 56.7%-90.7% accuracy, support vector machines (SVM) and ANNs were used by (Ramya and Lydia, 2016) achieving 88%-92% accuracy, advanced deep neural networks (DNN) were used by (Picon *et al.*, 2018) achieving 96% accuracy, and convolutional neural networks (CNN) by (Mg *et al.*, 2017) that managed 96.3% accuracy.

More advanced variants of CNN have been able to achieve higher accuracies in more complicated, multi-class classification problems. Notably, (Sladojevic *et al.*, 2016) reached up to 98% for 13 types of plant diseases, (Ferentinos, 2018) achieved 99.53% for 25 plant types with 58 diseases, (Mohanty, Hughes and Salathé, 2016) achieved 99.35% for 14 plants and 26 diseases.

The classification of multiple types of infections was studied on tomato leaves by (Fuentes *et al.*, 2017), wheat by (Wang *et al.*, 2012), banana trees by (Amara, Bouaziz and Algergawy, 2017), on brinjal (aubergine) by (Anand, Veni and Aravinth, 2016) and on apple trees by (Wang, Sun and Wang, 2017) and (Liu *et al.*, 2018).

We focus on the proper classification of images containing leaves from walnut trees, that are both infected by anthracnose and healthy. We investigate the performance of the most famous classification ML algorithms on this problem, in order to understand the pros and cons of each approach.

The structure of this paper is as follows: in paragraph 2 we present the methodology that was followed for the acquisition and the preparation of the data, the ML algorithms that were tested and the performance metric that were used for the evaluation, in paragraph 3 we present the results of the investigatory study, and in paragraph 4 a short discussion as well as the conclusions of our study.

2. Methodology

Anthracnose appears as brown or brown-yellowish marks on the leaves of the walnut tree. The marks usually appear as spots on the surface, or they cover the perimeter of the leaf. For the human eye, the symptoms of anthracnose (or of similar illnesses) are easy to detect. The aim is to identify the machine learning methodology that can perform as well as this human perception. The methodology that has been followed in this study, is presented in the next paragraphs.

2.1 Data acquisition

Image-based classification requires a large number of data, with distinct features in order to be able to train successfully a model to predict a class. A total number of 2.000 leaf images was collected from a walnut crop field located in Rizomylos Volos, Greece. The dataset was balanced with half of the images to contain leaves which are infected by anthracnose, and half that are healthy. In Figure 1 (a) we can see an indicative image from a healthy leaf, and in Figure 1 (b) a leaf infected by anthracnose.

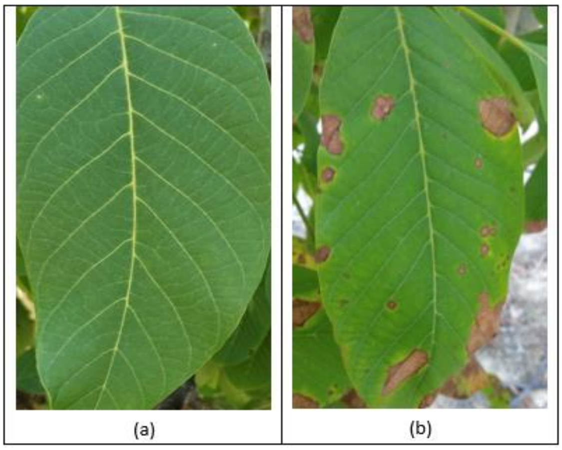

Figure 1. Walnut tree leaves without (a) and with anthracnose (b).

2.2 Data preparation

Each image was originally resized into a 128x128 pixel resolution. This resolution is large enough to keep the characteristics of the leaf, and also small enough to be able train the model within a desired time. Since colour information is important for the identification of anthracnose, all images are collected as RGB and as such as are used in the algorithms. Each image is then reshaped, by transforming from a 128x128x3 shape, into a 1x49,152 vector. This was conducted for each image, leading to a 2,000x49,152 matrix of pixel-wise features. One last column is being added to the matrix, containing the label of the image (i.e. 'Anthracnose' or 'Healthy') therefore the matrix becomes 2,000x49,153. The label column which now is categorical, is being encoded into binary representation, '0' for anthracnose and '1' for healthy. Finally, the rows of the dataset are being shuffled twice, with two different random-generated methods in order to avoid a biased sampling, that would lead to improperly trained models.

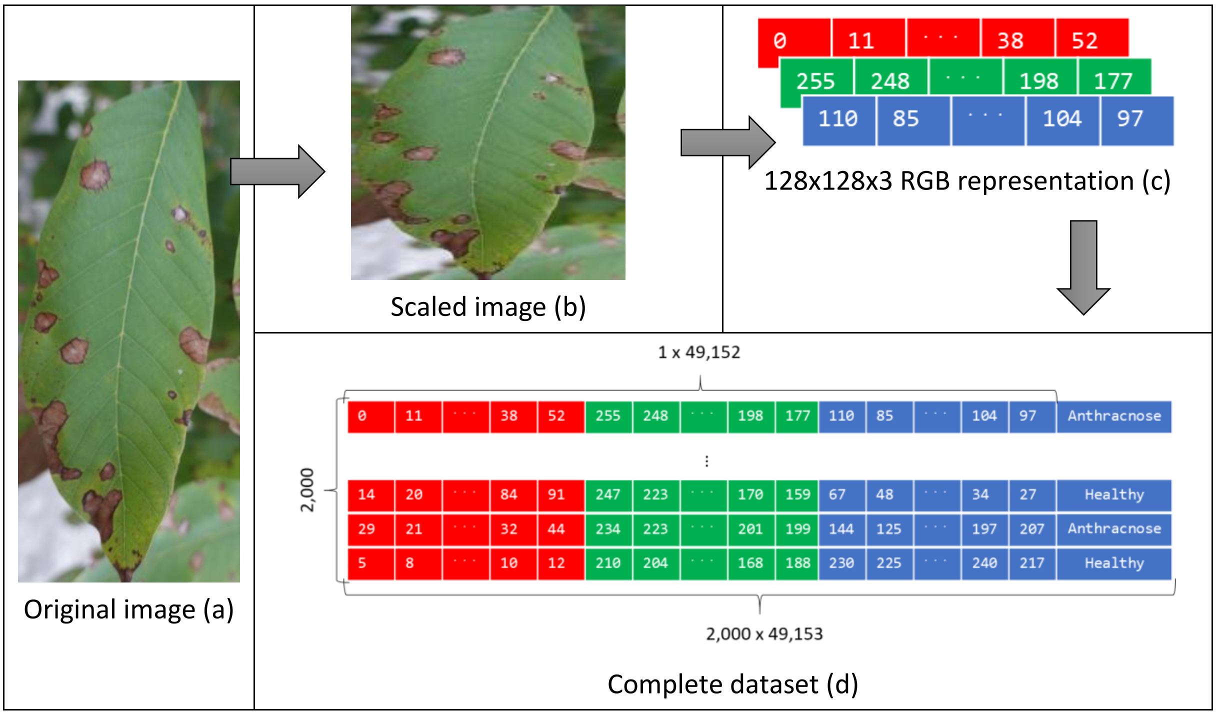

Figure 2. Data preparation from original image (a) to scaled image (b), to RGB tables (c), and finally to pixel-wise dataset (d).

2.3 Data split

Before any implementation to the models, the dataset is being split into three sections. Initially, the dataset is divided into training and testing with an 80/20 rate respectively. The testing portion is completely hidden from the model training process (hold-out), in order to obtain predictions that are unknown to the trained classifier. The second split regards the training dataset, of which the 20% is used for validation. The purpose of that is to make sure that the algorithms do not overfit during training, in order to keep good generalization for the model. Once the model is trained, the testing dataset will be used to make predictions, and based on that, the accuracy of the model will be obtained.

2.4 ML Methods

For our study, we have chosen 11 of the most famous ML algorithms in order to see how well each of these different methodologies can achieve reliant performances. These ML algorithms and the particular implementations are described shortly.

The Bayesian algorithms, Gaussian Naive Bayes (GNB) (Russell and Norvig, 2002) and Linear Discriminant Analysis (LDA) (Büyüköztürk and Çokluk-Bökeoğlu, 2008), suited for high-dimensionality problems, were implemented with no prior probabilities for the classes, and with the singular value decomposition (SVD) solver for the LDA. The k-Nearest Neighbours (kNN) (Altman, 1992), an instant-based learning algorithm where the function is only approximated locally, and all computation is deferred until classification, was implemented with 5 neighbours (*k*). The Decision Tree (DT) (Breiman, 1984), which builds a classification model in a tree structure, learns simple if-then decision rules, which offer interpretability and require minimal data preparation. Three ensemble classifiers, Random Forest (RF) (Breiman, 2001), Adaptive- (Adaboost, Ada-b) (Freund and Schapire, 1996) and Gradient-Boosting (Grad-B) (Mason *et al.*, 1999), methods that create strong classifiers by combining weak learners such as DT and bootstrap aggregating to obtain better predictive performance than could be achieved from the constituent algorithms on their own, were implemented with DT as weak learners, and 100 estimators each. Support Vector Machine (SVM) (Vapnik, 1999), a supervised learning algorithm with main aim to plot data is higher dimensionality spaces and find a hyperplane that differentiates classes by maximizing the distance between the hyperplane and the closest data point from each group, was implemented both with a radial basis function kernel, and with a *nu* parameter, set to 0.5, which is used to control the number of vectors. Finally, two Artificial Neural Networks (ANN) (McCulloch and Pitts, 1943), computing systems inspired by the biological neural networks were implemented. A multi-layer perceptron (MLP) with one hidden layer of 100 nodes, and a deep neural network (DNN) with a 5-layer architecture, an 8, 16, 16, 16, 16 node configuration and a 50% dropout rate per hidden layer. Both were implemented with 'ReLU' activation in all hidden layers, and sigmoid function for the output layer, since we have binary classification. 'Adam' optimizer and early stopping with 10 epochs patience were used for the training.

2.5 Performance metrics

The performance metrics used in this study are described in this paragraph. After a classifier is trained, it predicts the class of a new entry from the testing set. Depending on the prediction and the actual class it belongs, this prediction can be true positive (TP) or true negative (TN) if it is classified correctly, or false positive (FP) or false negative (FN) if it is misclassified.

Accuracy is the most intuitive metric, considering symmetric datasets, and is defined as the ratio of correctly predicted observation to the total observations (TP+TN/TP+FP+FN+TN). Precision is the ratio of correctly predicted positive observations to the total predicted positive observations and is defined as (TP/TP+FP). Recall is the ratio of correctly predicted positive observations to the all observations in actual class (TP/TP+FN). F1 Score, which is preferred when the class distribution is unbalanced, is the weighted average of Precision and Recall (2*(Recall * Precision) / (Recall + Precision)).

Finally, logarithmic loss measures the performance of the classification model where the prediction input is a probability value between 0 and 1. Log loss increases as the predicted probability diverges from the actual label with ideal goal to minimize this value to 0. For the calculation, a probability is assigned to each class rather instead of yielding the most likely class. Mathematically, log loss for binary classification is defined as:

$$-(y\log(p) + (1-y)\log(1-p)) \quad (1)$$

where y is a binary indicator (0 or 1) of whether a class label is the correct classification for a given observation, and p is the model's predicted probability that the given observation belongs to a certain class.

3. Results

The results of the ML algorithms are presented in this section. All algorithms were trained on a Nvidia Titan 1080 Ti, and were programmed on Python's Sci-Kit Learn, Keras (with Tensorflow), Pandas, and Numpy libraries. In Table 1, the results of all the tested ML algorithms are presented

Table 1. Performance metrics for the ML algorithms' comparison.

Algorithms	Accuracy	Precision	Recall	F1	Log Loss	Fitting time (sec)
GNB	81.25%	0.81	0.81	0.81	6.479	2.08
LDA	88.50%	0.89	0.89	0.88	0.532	10.2

kNN	71.25%	0.8	0.71	0.69	4.503	3.02
DT	81.75%	0.82	0.82	0.82	6.303	36.7
RF	88.00%	0.88	0.88	0.88	0.369	1.44
Ada-B	87.25%	0.87	0.87	0.87	0.631	165
Grad-B	89.25%	0.89	0.89	0.89	0.247	328
Nu-SVM	89.25%	0.89	0.89	0.89	0.263	462
SVM	85.00%	0.85	0.85	0.85	0.338	472
MLP	87.75%	0.88	0.88	0.88	3.256	84.3
DNN	**90.25%**	**0.9**	**0.9**	**0.9**	**0.295**	**74**

Deep neural network, nu-SVM and gradient boosting achieved the best performance. These three algorithms were able to achieve a low logarithmic loss, however the training time of the DNN was significantly less compared to the others. Deep neural network seems to be the best performing algorithm for this application.

4. Discussion & Conclusions

A comparative analysis was conducted on the applicability and the performance of machine learning algorithms for image-based classification of anthracnose in walnut tree leaves. An adequate number of 2.000 images was obtained, with equal portions of healthy and anthracnose-infected leaves. A total of 11 ML algorithms were tested and evaluated based on their performance and the accuracy they achieved over the collected dataset. The algorithms' accuracies ranged from a 71.25% to 90.25%, with most of them reaching over 85%. The algorithm that achieved the best performance, was the deep neural network implementation, which also achieved a low logarithmic loss (0.295) and a average training time (74 sec) compared to the other algorithms. Considering the performance of the deep neural network architecture, image-based anthracnose detection of walnut tree leaves, is a viable option for application in a real-life field.

This exploratory analysis points the way towards a more in-depth study on the algorithms that are based on deep neural structures, such as convolutional neural networks (CNN) and recurrent neural networks (RNN). Future plans include investigation of more complex derivatives, as well as meta-architectures of CNN that can conduct object detection, as well as instance aware image segmentation. Additionally, pre-processing methods such as fast Fourier transformation and wavelet decomposition, should be considered for investigation.

As far as the author's knowledge, other studies have reached similar performances in the particular problem of walnut trees/anthracnose (Gamal *et al.*, 2017). Aim for this study and its continuation is to build a ground-up, high-accuracy (>99%) classifier, based on real-conditions dataset of walnut trees' canopies. This way, robust, autonomous systems will be able to detect the disease on leaves, with high confidence, without human supervision.

Acknowledgement

The work was supported by the project "Research Synergy to address major challenges in the nexus: energy-environment-agricultural production (Food, Water, Materials)"—NEXUS, funded by the Greek Secretariat for Research and Technology (GSRT)—Pr. No. MIS 5002496.

References

Altman, N. S. (1992) 'An introduction to kernel and nearest-neighbor nonparametric regression', *American Statistician*. doi: 10.1080/00031305.1992.10475879.

Amara, J., Bouaziz, B. and Algergawy, A. (2017) 'A Deep Learning-based Approach for Banana Leaf Diseases Classification', in *BTW*.

Anand, R., Veni, S. and Aravinth, J. (2016) 'An application of image processing techniques for detection of diseases on brinjal leaves using k-means clustering method', in *2016 International Conference on Recent Trends in Information Technology, ICRTIT 2016*. doi: 10.1109/ICRTIT.2016.7569531.

Bharate, A. A. and Shirdhonkar, M. S. (2018) 'A review on plant disease detection using image processing', in *Proceedings of the International Conference on Intelligent Sustainable Systems, ICISS 2017*, pp. 103–109. doi: 10.1109/ISS1.2017.8389326.

Breiman, L. (1984) 'Classification and regression trees Regression trees', *Encyclopedia of Ecology*. doi: 10.1007/s00038-011-0315-z.

Breiman, L. (2001) 'Random Forrest', *Machine Learning*. doi: 10.1023/A:1010933404324.

Büyüköztürk, Ş. and Çokluk-Bökeoğlu, Ö. (2008) 'Discriminant function analysis: Concept and application', *Egitim Arastirmalari - Eurasian Journal of Educational Research*, (33), pp. 73–92.

Ferentinos, K. P. (2018) 'Deep learning models for plant disease detection and diagnosis', *Computers and Electronics in Agriculture*. doi: 10.1016/j.compag.2018.01.009.

Freund, Y. and Schapire, R. R. E. (1996) 'Experiments with a New Boosting Algorithm', *International Conference on Machine Learning*. doi: 10.1.1.133.1040.

Fuentes, A. *et al.* (2017) 'A robust deep-learning-based detector for real-time tomato plant diseases and pests recognition', *Sensors (Switzerland)*. doi: 10.3390/s17092022.

Gamal, A. *et al.* (2017) 'A New Proposed Model for Plant Diseases Monitoring Based on Data Mining Techniques', in Hakeem, K. R. et al. (eds) *Plant Bioinformatics: Decoding the Phyta*. Cham: Springer International Publishing, pp. 179–195. doi: 10.1007/978-3-319-67156-7_6.

Liakos, K. G. *et al.* (2018) 'Machine learning in agriculture: A review', *Sensors (Switzerland)*. Multidisciplinary Digital Publishing Institute, p. 2674. doi: 10.3390/s18082674.

Liu, B. *et al.* (2018) 'Identification of apple leaf diseases based on deep convolutional neural networks', *Symmetry*. doi: 10.3390/sym10010011.

Mason, L. *et al.* (1999) 'Boosting algorithms as gradient descent in Function space', *Nips*. doi: 10.1109/5.58323.

McCulloch, W. S. and Pitts, W. (1943) 'A logical calculus of the ideas immanent in nervous activity', *The Bulletin of Mathematical Biophysics*. doi: 10.1007/BF02478259.

Mg, A. *et al.* (2017) 'Plant Leaf Disease Detection using Deep Learning and Convolutional Neural Network', *International Journal of Engineering Science and Computing*.

Mohanty, S. P., Hughes, D. P. and Salathé, M. (2016) 'Using Deep Learning for Image-Based Plant Disease Detection', *Frontiers in Plant Science*. doi: 10.3389/fpls.2016.01419.

Muthukannan, K. *et al.* (2015) 'Classification of diseased plant leaves using neural network algorithms', *ARPN Journal of Engineering and Applied Sciences*.

Picon, A. *et al.* (2018) 'Deep convolutional neural networks for mobile capture device-based crop disease classification in the wild', *Computers and Electronics in Agriculture*. doi: 10.1016/j.compag.2018.04.002.

Ramya, V. and Lydia, M. A. (2016) 'Leaf Disease Detection and Classification using Neural Networks', *International Journal of Advanced Research in Computer and Communication Engineering*, 5(11), pp. 207–210. doi: 10.17148/IJARCCE.2016.51144.

Russell, S. and Norvig, P. (2002) *Artificial Intelligence: A Modern Approach (2nd Edition)*, *Prentice Hall*. doi: 10.1017/S0269888900007724.

Sladojevic, S. *et al.* (2016) 'Deep Neural Networks Based Recognition of Plant Diseases by Leaf Image Classification', *Computational Intelligence and Neuroscience*, 2016. doi: 10.1155/2016/3289801.

Vapnik, V. N. (1999) 'An overview of statistical learning theory', *IEEE Transactions on Neural Networks*. doi: 10.1109/72.788640.

Wang, G., Sun, Y. and Wang, J. (2017) 'Automatic Image-Based Plant Disease Severity Estimation Using Deep Learning', *Computational Intelligence and Neuroscience*. doi: 10.1155/2017/2917536.

Wang, H. *et al.* (2012) 'Application of neural networks to image recognition of plant diseases', in *2012 International Conference on Systems and Informatics, ICSAI 2012*. doi: 10.1109/ICSAI.2012.6223479.