

## Original papers

## Deep learning models for plant disease detection and diagnosis

Konstantinos P. Ferentinos



Hellenic Agricultural Organization "Demeter", Institute of Soil &amp; Water Resources, Dept. of Agricultural Engineering, 61 Dimokratias Av., 13561 Athens, Greece

## ARTICLE INFO

## Keywords:

Convolutional neural networks  
Machine learning  
Artificial intelligence  
Plant disease identification  
Pattern recognition

## ABSTRACT

In this paper, convolutional neural network models were developed to perform plant disease detection and diagnosis using simple leaves images of healthy and diseased plants, through deep learning methodologies. Training of the models was performed with the use of an open database of 87,848 images, containing 25 different plants in a set of 58 distinct classes of [plant, disease] combinations, including healthy plants. Several model architectures were trained, with the best performance reaching a 99.53% success rate in identifying the corresponding [plant, disease] combination (or healthy plant). The significantly high success rate makes the model a very useful advisory or early warning tool, and an approach that could be further expanded to support an integrated plant disease identification system to operate in real cultivation conditions.

## 1. Introduction

Plant disease diagnosis through optical observation of the symptoms on plant leaves, incorporates a significantly high degree of complexity. Deep Learning (LeCun et al., 2015). Deep learning refers to the use of their existing phytopathological problems, even experienced agronomists and plant pathologists often fail to successfully diagnose specific diseases, and are consequently led to mistaken conclusions and treatments (LeCun et al., 1998; Dan et al., 2011), voice recognition. The existence of an automated computational system for the detection and diagnosis of plant diseases, would offer a valuable assistance to the agronomist who is asked to perform such diagnoses through optical observation of leaves of infected plants (Mohanty et al., 2016; Yang and Guo, 2017). If the system was simple to use and easily accessible through a simple mobile application, it could also be a valuable tool for farmers in parts of the world lacking the appropriate infrastructure for the provision of agronomic and phytopathological advice. In addition, in the case of large-scale cultivations, the system could be combined with autonomous agricultural vehicles, to accurately and timely locate phytopathological problems throughout the cultivation field, using continuous image capturing. All these are, performing pattern recognition in applications with large amount of course, valid under the condition that the system could achieve high levels of performance in detecting and diagnosing specific diseases. The basic deep learning tool used in this work is Convolutional Neural Networks (CNNs) (LeCun et al., 1998). CNNs constitute one of the most powerful techniques for modeling complex processes and data, like the one of pattern recognition in images. Lee et al. (2015) presented a CNNs system for the automated recognition of plants, based on leaves images. Grinblat et al. (2016) developed a relatively simple, operated through an appropriate, easy-to-use, and user-friendly mobile application (a first step towards that direction has been made by Johannes et al. (2017) for the specific case of wheat plants).

With the development of computational systems in recent years, established architectures of CNNs in the identification of 26 plant diseases, in particular Graphical Processing Units (GPU) embedded processor cases, using an open database of leaves images of 14 different plants. Machine Learning-related Artificial Intelligence applications have their results were very promising, with success rates in the automated

achieved exponential growth, leading to the development of novel methodologies and models, which now form a new category, that of

E-mail address: kp3@cornell.edu .

Table 1a  
Information and quantitative data of the database images.

Class	Plant common name	Plant scientific name	Disease common name	Disease scientific name	Images (number)	Laboratory conditions (%)	Field conditions (%)
c_0	Apple Malus domestica	—	1835	89.7	10.3		
c_1	Apple Malus domestica	Apple scab	Venturia inaequalis	630	100.0	0.0	
c_2	Apple Malus domestica	Cedar apple rust	Gymnosporangium juniperi-virginianae			276	100.0
c_3	Apple Malus domestica	Black rot	Botryosphaeria obtusa	712	87.2	12.8	
c_4	Banana Musa paradisiaca	—	1643	0.0	100.0		
c_5	Banana Musa paradisiaca	Black sigatoka	Mycosphaerella fi jensis	240	0.0	100.0	
c_6	Banana Musa paradisiaca	Banana speckle	Mycosphaerella musae	3284	0.0	100.0	
c_7	Blueberry Vaccinium spp.	—	1735	86.7	13.3		
c_8	Cabbage Brassica oleracea	—	420	0.0	100.0		
c_9	Cabbage Brassica oleracea	Black rot	Xanthomonas campestris	64	0.0	100.0	
c_10	Cantaloupe Cucumis melo	—	1055	0.0	100.0		
c_11	Cassava (manioc) Manihot esculenta	Brown leaf spot	Cercosporidium henningsii	43	100.0	0.0	
c_12	Cassava (manioc) Manihot esculenta	Cassava green spider mite	Mononychellus tanajoa & progresivus			892	100.0
c_13	Celery Apium graveolens	Early blight, Cercospora	Cercospora apii	1204	0.0	100.0	
c_14	Cherry (& sour) Prunus spp.	—	854	100.0	0.0		
c_15	Cherry (& sour) Prunus spp.	Powdery mildew	Podosphaera spp.	1052	100.0	0.0	
c_16	Corn (maize) Zea mays	—	4450	26.1	73.9		
c_17	Corn (maize) Zea mays	Cercospora leaf spot	Cercospora zeae-maydis	1457	35.2	64.8	
c_18	Corn (maize) Zea mays	Common rust	Puccinia sorghi	1614	73.9	26.1	
c_19	Corn (maize) Zea mays	Northern Leaf Blight	Exserohilum turcicum	985	100.0	0.0	
c_20	Cucumber Cucumis sativus	—	267	0.0	100.0		
c_21	Cucumber Cucumis sativus	Downy mildew	Pseudoperonospora cubensis	1318	0.0	100.0	
c_22	Eggplant Solanum melongena	—	515	0.0	100.0		
c_23	Gourd Langenaria spp.	Downy mildew	Pseudoperonospora cubensis	114	0.0	100.0	
c_24	Grape Vitis vinifera	—	613	69.0	31.0		
c_25	Grape Vitis vinifera	Black rot	Guignardia bidwellii	1180	100.0	0.0	
c_26	Grape Vitis vinifera	Esca (Black measles)	Phaeomoniella chlamydospora	1384	100.0	0.0	
c_27	Grape Vitis vinifera	Leaf blight	Pseudocercospora vitis	1076	100.0	0.0	
c_28	Onion Allium cepa	—	154	0.0	100.0		
c_29	Orange Citrus sinensis	Huanglongbing	Candidatus Liberibacter	5507	100.0	0.0	

identifi cation up to 99.35%. However, a main drawback was that the Materials and methods entire photographic material included solely images in experimental

(laboratory) setups, not in real conditions in the cultivation field.

#### 2.1. Convolutional neural network models

Sladojevic et al. (2016) developed a similar methodology for plant

disease detection through leaves images using a similar amount of data. Artificial neural networks are mathematical models that mimic the available on the Internet, which included a smaller number of diseases. General principles of brain function with their neurons and synapses (13) and different plants (5). Success rates of their models were between 91% and 98%, depending on the testing data. More recently, trained through the process of supervised learning. During that process, Pawara et al. (2017) compared the performance of some conventional neural networks are “trained” to model some system with the use of pattern recognition techniques with that of CNN models, in plants existing data that contain specific matchings of inputs and outputs of identifi cation, using three different databases of (a rather limited) the system to be modelled. CNNs (LeCun et al., 1998) are an evolution number of images of either entire plants and fruits, or plant leaves of traditional artificial neural networks, focused mainly on applications concluding that CNNs drastically outperform conventional methods with repeating patterns in different areas of the modeling space, especially in fruit and vegetable diseases. Finally, Fuentes et al. (2017) developed CNN models for the detection of tomato diseases and pests, with satisfactory performance. In their methodology used in their layering, they drastically reduce the required number of structural elements (number of artificial neurons) in comparison to traditional feedforward neural networks. For image recognition based on simple images of leaves of healthy and diseased plants. The recognition applications, several baseline architectures of CNNs have been developed, which have been successfully applied to complicated laboratory) setups and real cultivation conditions in the field. The tasks of visual imagery.

In this work, specific CNN architectures were trained and assessed, number of structural elements (number of artificial neurons) in comparison to form an automated plant disease detection and diagnosis system. The five basic CNN architectures that were tested in the problem shallow approaches, which learn with less data but are specific to investigated in this work concerning the identification of plant diseases crops (e.g., Pantazi et al., 2016). The next section presents the basic principles of the tested models, the datasets used for training and testing, and the experimentations that were designed for the investigation of the factors that affect the performance and robustness of the developed system. Section 3 presents the results of the application and testing processes, were implemented using Torch<sup>1</sup> machine of the proposed models for plant disease detection and diagnosis, which uses the LuaJIT<sup>2</sup> computational framework, which uses the LuaJIT<sup>2</sup> the paper closes with some concluding remarks and directions for future research towards the evolution and enhancement of the developed system.

<sup>1</sup> <http://torch.ch>

<sup>2</sup> <http://www.lua.org>

Table 1b  
Information and quantitative data of the database images (continued).

Class	Plant common name	Plant scienti c name	Disease common name	Disease scienti c name	Images (number)	Laboratory conditions (%)	Field conditions (%)
c_30	Peach Prunus persica	—	360	100.0 0.0			
c_31	Peach Prunus persica	Bacterial sport Xanthomonas campestris	2297	100.0 0.0			
c_32	Pepper, bell Capsicum annuum	—	2029	72.8 27.2			
c_33	Pepper, bell Capsicum annuum	Bacterial spot Xanthomonas campestris	997	100.0 0.0			
c_34	Potato Solanum tuberosum	—	152	100.0 0.0			
c_35	Potato Solanum tuberosum	Late blight Phytophthora infestans	1000	100.0 0.0			
c_36	Potato Solanum tuberosum	Early blight Alternaria solani	3167	31.6 68.4			
c_37	Pumpkin Cucurbita spp.	Cucumber mosaic virus (CMV)	2387	0.0 100.0			
c_38	Raspberry Rubus spp.	—	371	100.0 0.0			
c_39	Soybean Glycine max	—	6235	81.6 18.4			
c_40	Soybean Glycine max	Downy mildew Peronospora manshurica	851	0.0 100.0			
c_41	Soybean Glycine max	Frogeye leaf spot Cercospora sojina	2023	0.0 100.0			
c_42	Soybean Glycine max	Septoria Leaf Blight Septoria glycines	3565	0.0 100.0			
c_43	Squash Cucurbita spp.	—	264	0.0 100.0			
c_44	Squash Cucurbita spp.	Powdery mildew Erysiphe cichoracearum, Sphaerotheca fuliginea	1835	100.0 0.0			
c_45	Strawberry Fragaria spp.	—	456	100.0 0.0			
c_46	Strawberry Fragaria spp.	Leaf scorch Diplocarpon earlianum	3396	29.7 70.3			
c_47	Tomato Lycopersicum esculentum	—	1592	100.0 0.0			
c_48	Tomato Lycopersicum esculentum	Bacterial spot Xanthomonas campestris pv. Vesicatoria	2127	100.0 0.0			
c_49	Tomato Lycopersicum esculentum	Early blight Alternaria solani	2579	38.8 61.2			
c_50	Tomato Lycopersicum esculentum	Late blight Phytophthora infestans	1910	100.0 0.0			
c_51	Tomato Lycopersicum esculentum	Septoria leaf spot Septoria lycopersici	1771	100.0 0.0			
c_52	Tomato Lycopersicum esculentum	Spider mites Tetranychus urticae	1653	100.0 0.0			
c_53	Tomato Lycopersicum esculentum	Tomato mosaic virus (ToMV)	373	100.0 0.0			
c_54	Tomato Lycopersicum esculentum	virus					
c_55	Tomato Lycopersicum esculentum	Leaf Mold Fulvia fulva	952	100.0 0.0			
c_56	Tomato Lycopersicum esculentum	Target spot Corynespora cassiicola	1404	100.0 0.0			
c_57	Watermelon Citrullus lanatus	T YLCV Begomovirus (Fam. Geminiviridae)	5357	100.0 0.0			
TOTAL: 87,848		62.7 37.3					

programming language. Training algorithms were implemented on the GPU of an NVIDIA ® GTX1080 card, using the CUDA ® parallel programming platform, in Linux environment (Ubuntu 16.04 LTS operating system).

## 2.2. Training and testing datasets

An open database containing 87,848 photographs of leaves of healthy and infected plants was used for the training and testing of CNN models. A preliminary version of the database, containing a smaller number of images, is described in Hughes and Salathé (2015). The database that was used here, includes 58 different classes, where each class is defined as a pair of plant and a corresponding disease. and 1b presents information on the 58 classes, including some statistical data, such as the number of available images per class, and the percentages of images taken in laboratory setups or in real conditions in cultivation fields. These 58 classes comprise 25 different healthy or diseased plants. As shown in Table 1, more than 1/3 of the available images (37.3%) have been captured in real cultivation conditions in the field. Fig. 1 shows samples of a random class, containing two representative images in laboratory conditions, and two in real conditions. The increased complexity of the latter images is obvious, with several aspects contributing to it, such as multiple leaves and other parts of the plants, irrelevant objects (e.g., shoes), different ground textures, shading effects, etc.

The entire database was initially divided into two datasets, the training set, and the testing set, by randomly splitting the 87,848 images so that 80% of them formed the training set, and 20% formed the testing set. The 80/20 splitting ratio of training/testing datasets is the most commonly used one in neural network applications, and other similar splitting ratios (e.g., 70/30) should not have a significant impact on the performance of the developed models ( Fine, 2006 ). Thus, for the training of the CNN models, 70,300 images were used, while the rest 17,548 images were kept for testing the performance of the models in classifying new, previously “unseen” images. A Python script was developed for the creation of the two datasets, producing uniformly distributed pseudorandom numbers for the random selection of the images, thus the percentages of “laboratory conditions” images and “real conditions” images for both datasets (training and testing) were kept similar to those presented in Table 1. In addition, another approach to the development of the training and testing data was also considered, by pre-processing of the images, which included size reduction and cropping to a 256 × 256 pixels size, while the training/testing ratio was kept the same (80/20). The alternative of using grayscale versions of the images for training was not considered, as previous works (e.g., Mohanty et al., 2016) have indicated that this approach does not improve the final classification performance of deep learning models in similar applications. The same holds for segmentation of the leaves from the background of the images, thus this additional step in the process was also not considered. This holds because deep learning systems like CNNs have the ability to identify the



Fig. 1. Samples of images in laboratory conditions (up) and in field conditions (down). [From class c\_49 – Tomato with Early blight]

Table 2  
Final CNN training parameters values.

Parameter	Value
Batches/epoch	10,000
Batch size	32
Momentum	0.9
Weight decay	0.0005
Learning rate	0.01 – 0.0001

important and non-important features of a set of images, and in some way, ignore the latter. Thus, the additional step of segmentation of the objects of interest, which can also become very problematic in complex images like the field-images in the current application, can be avoided.

Finally, a third approach to the separation of the database into training and testing datasets was implemented, focusing on assessing the importance of the type and place of capture of the leaves images in the cultivation field. Thus, from the 58 available classes of the form [plant, disease], the 12 which contained images of both types

Table 4  
Final models' performance on the testing dataset, trained with the original images.

Model	Success rate	Average error	Epoch	Time (s/epoch)
AlexNetOWTBn	99.49%	0.0174	121	6647
VGG	99.53%	0.0223	67	7034

were selected (the rest of the classes solely consist either of laboratory conditions images or field conditions images, as shown in Table 1). With these 12 classes, two experiments were conducted and two CNN models were developed, respectively: one which was trained with solely laboratory conditions images and tested on field conditions images, and another which was trained and tested reversely, i.e. it was trained with field conditions images and tested on laboratory conditions images. In both cases, the training/testing ratio was not ideal, because, in those 12 classes, the percentage of laboratory conditions images was 55.8% and of field conditions images was 44.2%, thus the training/testing ratio was far from the 80/20 ratio that was used in the development of the basic model. In the second case in particular, the training set was smaller than the testing set, something that, in general, is not acceptable. However, the results, as presented in the following section, are quite indicative.

### 3. Results

#### 3.1. Optimal deep learning model

All different CNN models presented in Section 2.1 were trained using the training parameters shown in Table 2. After appropriate experimentation, these values gave the best results during training. The learning rate followed a specific annealing schedule, starting from 0.01 and descending every 20 epochs by 1/2 or by 1/5, alternately, down to 0.0001. Comparison of the models was based on their performance on the testing set (all models achieved 100% accuracy on the training set). Table 3 presents successful classification percentages during testing of the various models, for the two basic training/testing approaches of the 58 classes (i.e., training/testing in a 80/20 ratio, with the original images in the first case, and the pre-processed, down-scaled, squared images in the second case). Presented metrics include:

(i) the percentages of successful classification of [plant, disease] classes: this is the number of correctly classified images over the total number of images, where "classification" is the first choice of the models, meaning that the successful classification is the opposite of the top-1 error,

(ii) the corresponding average errors of the models: this is the average losses per batch, over all batches of the testing set,

(iii) the training epoch at which best performances were achieved, the importance of the type and place of capture of the leaves images (iv) the average time per epoch (in seconds) required for the training of each model.

The results presented in Table 3 indicate that all models achieve better

Table 3

Performance of different CNN model architectures for the identification of [plant, disease] classes on the testing dataset.

Model	Original images		Pre-processed images		Success rate	Average error	Epoch	Time (s/epoch)
	Success rate	Average error	Success rate	Average error				
AlexNet	99.06%	0.0354	47	7034	98.64%	0.0658	50	1022
AlexNetOWTBn	99.44%	0.0192	46	7520	99.07%	0.0332	45	1125
GoogLeNet	97.27%	0.0957	45	7845	97.06%	0.0984	40	2670
Overfeat	98.96%	0.0412	45	6204	98.26%	0.0848	49	1570
VGG	99.48%	0.0223	48	7294	98.87%	0.0542	49	4208

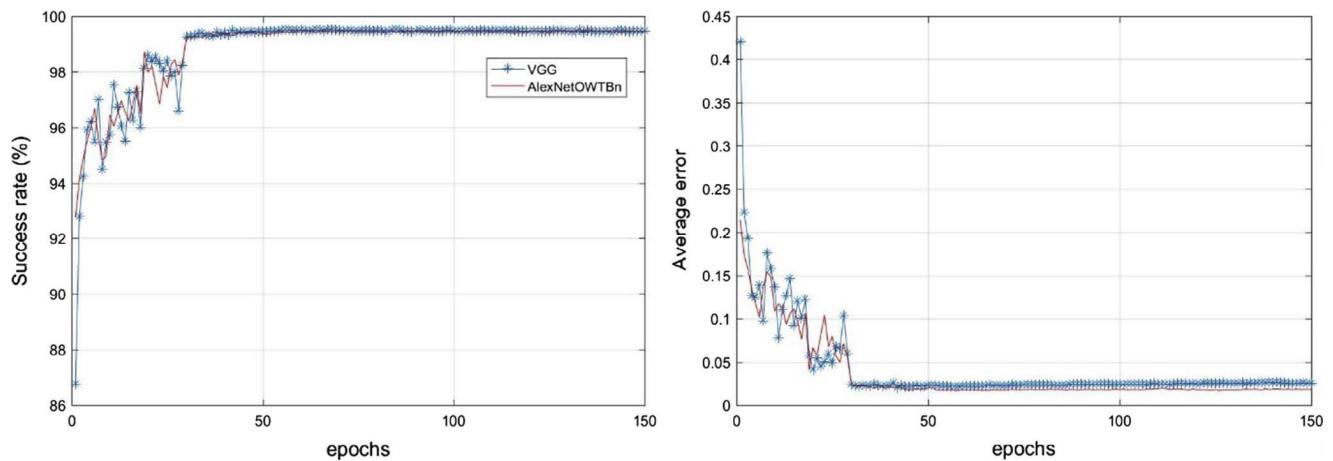


Fig.

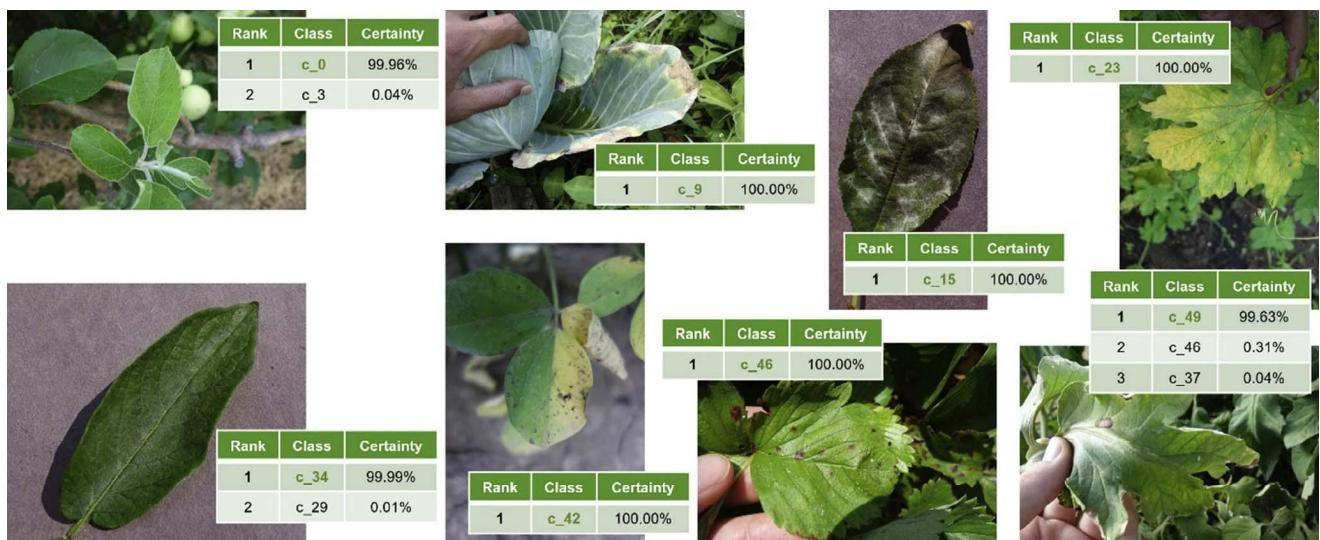


Fig. 3. Examples of correct classifications of various images of the testing dataset.

Table 5  
Final models' performance with different training/testing scenarios in respect to laboratory-conditions and field-conditions images.

	Training: Laboratory - Testing: Field	Training: Field - Testing: Laboratory				
Model	Success rate	Average error	Epoch Time (s/epoch)	Success rate	Average error	Epoch Time (s/epoch)
AlexNetOWTBn	32.23%	3.5484	53 4375	62.57%	1.9369	104 –
VGG	33.27%	7.8541	54 4901	65.69%	2.6786	134 –

the inevitable consequence of requiring more training time. The highest certainty level (i.e., Rank 1) is considered to be the final success rates were achieved by the VGG (highest success rate) and outcome of the model's prediction). All images shown in Fig. 3 were AlexNetOWTBn architectures (lowest final average testing error). Directly classified, i.e., their corresponding classes were those returned by the two models were further trained, using solely the original images, for the highest certainty by the model (Rank 1). In most cases, no larger number of training epochs. As shown in Table 4, the final highest ranking exists, as the correct classification was returned with a successful classification percentage of 99.53% (i.e., a top-1 error of certainty level of about 100%). 0.47% was achieved by the VGG model, which constituted the final model of plant disease detection. Total training time for that mode was about 5.5 days (on a single GPU, as described in Section 2.1). Fig. 2 presents the performance on the testing dataset for both models, during their training process. Fig. 3 shows some classification cases of raw images capturing type, as analyzed in Section 2.2. The corresponding image takes on average about 2 ms on the same GPU that was used. Results are presented in Table 5. As expected, success rates are significantly lower than those achieved by using a combination of laboratory and field conditions images. The two most successful CNN models (VGG and AlexNetOWTBn) were further tested for the investigation of the importance of training images type, as analyzed in Section 2.2. The corresponding classification results on the tables next to each image are significantly lower than those achieved by using a combination of laboratory and field conditions images. The two most successful CNN models (VGG and AlexNetOWTBn) were further tested for the investigation of the importance of training images type, as analyzed in Section 2.2. The corresponding classification results on the tables next to each image are significantly lower than those achieved by using a combination of laboratory and field conditions images.

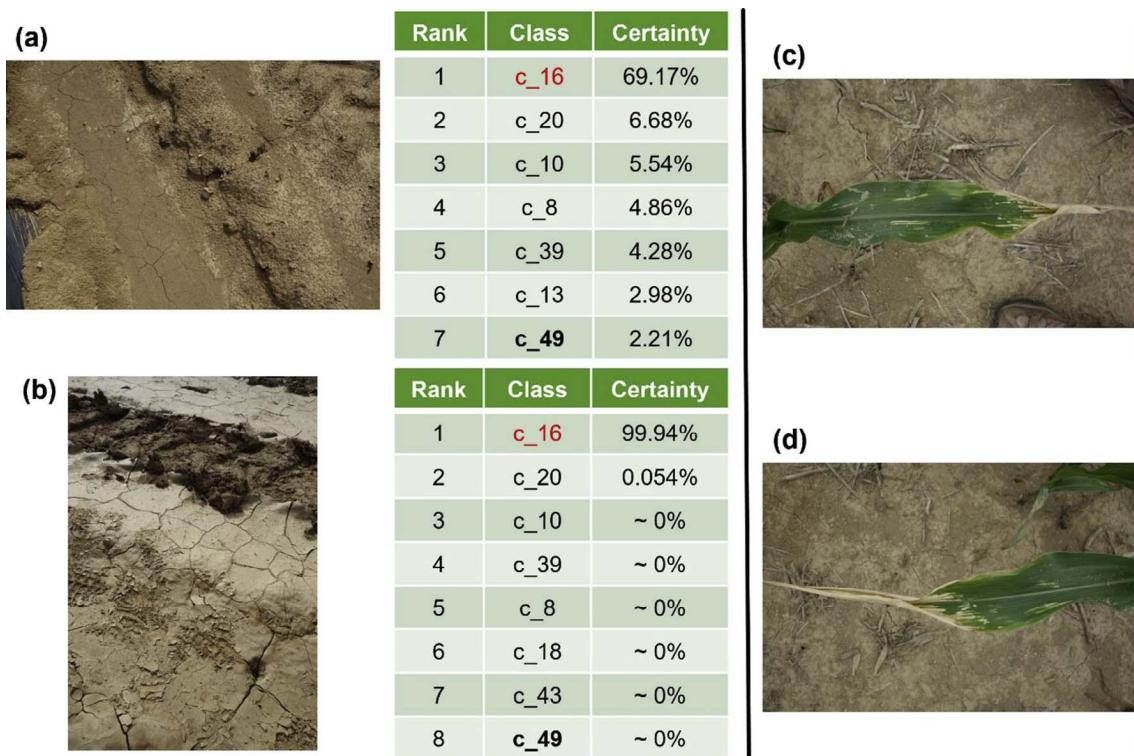


Fig. 4. (a) and (b): Faulty images registered in class c\_49 and their corresponding “classification” results. (c) and (d): Representative images of class c\_16, “results



Fig. 5. Representative examples of correct (green and yellow rectangles) and incorrect (red rectangles) classifications of testing images of the class c\_5 (Banana interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

available classes in the classification process (12 instead of 58). The importance of the existence of images captured in actual cultivation results show that models achieve better performance when trained with images for the development of useful and successful systems for the field-conditions images and asked to identify laboratory-conditions automated detection and diagnosis of plant diseases. images (success rates up to almost 68%). On the other hand, when trained solely on laboratory-conditions images and asked to identify field-conditions images, success rates are significantly lower (about 33%). This demonstrates the fact that image identification in actual cultivation conditions is a much more difficult and complex task than in the case of laboratory-conditions images, and proves the high success rate of 99.53% that the final model achieved on the testing dataset of 17,548 images, corresponds to correct classification of 17,466

([plant, disease] combinations) of just 82 images (0.47%). Among these could be either smartphones, to be used by growers or agro-82 misclassified images, there were some few “faulty” ones that economists, or drones and other autonomous agricultural vehicles, to be tained no plant leaves at all, like for example, those shown in Fig. 4(a) for real-time monitoring and dynamic disease detection on large-and (b). In that specific case, the “faulty” images were registered on open-field cultivations. In the former case in particular, except for class c\_49 of the database (Tomato with Early blight), while the main fact that a farmer at a remote location could have an incipient classified them in class c\_16 (healthy Corn), as shown on the classification about a possible threat for his/her cultivation, and an agro- cation tables in Fig. 4. These tables present the ranked outputs of the model could have a valuable advisory tool in his/her disposal, a future final CNN model for the images on their left. These images were possibility could be the development of an automated pesticide pre- counted as misclassified in the performance estimation of the system that would require a confirmation by the automated model (a “correct” classification would be the class c\_49, even though diagnosis system to allow the purchase of appropriate pesticides they do not actually belong to any class, as they do not contain any by the farmers. That would drastically limit the uncontrolled acquisition- plant leaves). They were both classified as class c\_16 probably due to of pesticides that leads to their overuse and misuse, with the the fact that the images of that class (two representative ones are subsequent catastrophic effects on the environment.

Other problematic situations regarding the field-conditions images constitutes, to our knowledge, the largest plant disease identification of the database included: (i) images with extensive partial shading task tackled with deep learning methodologies at the moment. the leaves, (ii) images with multiple objects in addition to the picture However, the expansion of the existing database to incorporate a wider leave or leaves, such as fingers, entire hands, shoes, parts of shirts, variety of plant species and diseases should be the next immediate fu- and (iii) images with the leave occupying a very small and non-centre step, a process that can be challenging in several aspects, as well as part of the frame. Judging from the high performance of the final time demanding. Another important issue that should be noted and model, these problems were overcome by the learning process in most should be resolved, is that the testing dataset used for the assessment of cases. An indicative case was that of class c\_5 (Banana with Blacktip models, was part of the same database that constituted the training gatoka), where 4 out of a total of 48 testing images were mistakenly set. This is common practice in machine learning models, but the real classifed (i.e., misclassified reached 8.33%, in contrast to the value of the developed system, especially for being able to be used in overall misclassification rate of 0.47%). Fig. 5 shows the classification situations, should be proven in testing data that would come from results of the model on 8 representative images of the c\_5 class, in various different sources and/or databases. Some preliminary experiments including the 4 misclassified ones (inside red rectangles). The first tends towards that direction, in a limited however amount of data, images inside green rectangles were classified correctly with a certainty a substantial reduction in model performance, in the range of practically 100%, while the image inside the yellow rectangle was 25 – 35%, depending on the data source, which is similar to correctly with a certainty of about 80% (the second corresponding performances reported in the literature (e.g., Mohanty et al. with a certainty level of 19.4% was class c\_6, which corresponds to (2016) reported a model accuracy of 31% in such data, for a problem banana plant with a different disease). The 4 misclassified images with 38 plant disease classes). For an improvement towards that direction the red rectangles suffered mainly from extensive partial shading reflection, a much wider variety of training data should be collected, from effects, which probably confused the CNN model, even though in two several sources of different geographic areas, cultivation conditions, them, the correct classification was the second choice of the model and image capturing modes and sets. The proposed deep learning approach with the first choice being the similar class c\_6 (Banana plants without banana speckle disease), thus the model identified correctly the plurality of available data to improve the system, and make it wider (in terms of plant species and diseases that can be identified) and more robust in real cultivation conditions.

#### 4. Conclusions

## References

- In this work, specialized deep learning models were developed, based on specific convolutional neural networks architectures, for the identification of plant diseases through simple leaves images of healthy or diseased plants. The training of the models was performed using an openly available database of 87,848 photographs, taken in both laboratory conditions and real conditions in cultivation fields. The database comprises 25 plant species in 58 distinct classes of [plant, disease] combinations, including some healthy plants. The most successful model architecture, a VGG convolutional neural network, achieved a success rate of 99.53% (top-1 error of 0.47%) in the classification of 17,548 previously unseen by the model plant leaves images (testing set). Based on that high level of performance, it becomes evident that convolutional neural networks are highly suitable for the automated detection and diagnosis of plant diseases through the analysis of simple leaves images. In addition, the high importance of the existence of real-conditions images (captured in the cultivation fields) in the training data, which was indicated by the presented results, suggests that, in the development of such models, focus should be given in the maximization of the ratio of real-conditions images in the training data. Furthermore, the low computational power required by the trained model to classify a given image (about 2 ms on a single GPU), makes feasible its integration into mobile applications for their use in mobile devices. Such

Baranza-Rojas, J., Goeau, H., Bonnet, P., Mata-Montero, E., Joly, A., 2017. Going beyond the automated identification of Herbarium specimens. *BMC Evol. Biol.* <http://doi.org/10.1186/s12862-017-1014-z>.

Dan, C., Meier, U., Masci, J., Gambardella, L.M., Schmidhuber, J. 2011. Flexible, high-performance convolutional neural networks for image classification. *Proceedings of the 22nd International Joint Conference on Artificial Intelligence*, vol. 2, pp. 1237 – 1242.

Fine, T.L., 2006. *Feedforward Neural Network Methodology*. Springer Science Business Media .

Fuentes, A., Yoon, S., Kim, S.C., Park, D.S., 2017. A robust deep-learning-based detector for real-time tomato plant diseases and pest recognition. *Sensors* 17, 2022. <http://doi.org/10.3390/s17092022> .

Grinblat, G.L., Uzal, L.C., Larese, M.G., Granitto, P.M., 2016. Deep learning for plant identification using vein morphological patterns. *Comput. Electron. Agric.* 127, 418 – 424 .

Hinton, G., Deng, L., Yu, D., Dahl, G.E., Mohamed, A.R., Jaitly, N., Senior, A., et al. 2012. Deep neural networks for acoustic modeling in speech recognition: The shared vocabulary approach. *IEEE Signal Process Mag.* 29 (6), 82 – 97 .

Hughes, D.P., Salathé, M. 2015. An open access repository of images on plant health enable the development of mobile disease diagnostics. [arXiv:1511.08060](http://arXiv:1511.08060).

Jolani, A., Picon, A., Alvarez-Gila, A., Echazarra, J., Rodriguez-Vaamonde, S., Ortiz-Barredo, A., 2017. Automatic plant disease diagnosis using mobile cameras, applied on a wheat use case. *Comput. Electron. Agric.* 138, 200 – 209 .

Krizhevsky, A. 2014. One weird trick for parallelizing convolutional neural networks arXiv:1404.5997.

Krizhevsky, A., Sutskever, I., Hinton, G.E. 2012. Imagenet classification with deep convolutional neural networks. In: *Advances in Neural Information Processing Systems*

- Eds. F. Pereira, C.J.C. Burges, L. Bottou, and K.Q. Weinberger (Curran Associates, Inc.), pp. 1097 – 1105.
- LeCun, Y., Bengio, Y. 1995. Convolutional networks for images, speech, and time series recognition. In: The handbook of brain theory and neural networks, vol. 3361(10). (ICPRAM 2017).
- LeCun, Y., Bengio, Y., Hinton, G., 2015. Deep learning. *Nature* 521, 436 – 445. [doi.org/10.1038/nature14539](https://doi.org/10.1038/nature14539).
- LeCun, Y., Bottou, L., Bengio, Y., Haaffner, P., 1998. Gradient-based learning applied to document recognition. *Proc. IEEE* 86 (11), 2278 – 2324 .
- Lee, S.H., Chan, C.S., Wilkin, P., Remagnino, P. 2015. Deep-plant: Plant identification with convolutional neural networks. 2015 IEEE Intl Conf. on Image Processing (ICIP 2015).
- Mohanty, S.P., Hughes, D.P., Salathé, M., 2016. Using deep learning for image-based plant disease detection. *Front. Plant Sci.* 7 <http://dx.doi.org/10.3389/fpls.2016.0289801>. Article ID: 01419. Article: 1419.
- Pantazi, X.E., Moshou, D., Tamouridou, A.A., Kasderidis, S. 2016. Leaf disease recognition in vine plants based on local binary patterns and one class support vector machine. *Pattern Recognition*.
- 12th IFIP International Conference on Artificial Intelligence Applications (AIAI 2016), Sept. 16 – 18, Thessaloniki Greece, Proceedings, Springer 6 – 9. <http://dx.doi.org/10.18088/ejbm.3.1.2016.pp6-9> . pp. 319 – 327.
- Simonyan, K., Zisserman, A. 2014. Very deep convolutional networks for large-scale image recognition. *arXiv:1409.1556*.
- Singovic, S., Arsenovic, M., Anderla, A., Culibrk, D., Stefanovic, D., 2016. Deep neural networks based recognition of plant diseases by leaf image classification. *Computational Intelligence Neurosci.* <http://dx.doi.org/10.1155/2016/3289801>. Article ID: 01419. Article: 1419.
- Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., et al. 2015. GoogLeNet: An ImageNet Winner Takes All.
- Ying, X., Guo, T., 2017. Machine learning in plant disease research. *Europ. J. BioInnovations (EJBMR)* 6 – 9. <http://dx.doi.org/10.18088/ejbm.3.1.2016.pp6-9> .