Computers and Electronics in Agriculture 127 (2016) 418–424

Contents lists available at ScienceDirect

Computers and Electronics in Agriculture

journal homepage: www.elsevier.com/locate/compag

Original papers

Deep learning for plant identification using vein morphological patterns

Guillermo L. Grinblat, Lucas C. Uzal *, Mónica G. Larese, Pablo M. Granitto

CIFASIS, French Argentine International Center for Information and Systems Sciences, UNR-CONICET, Argentina

ARTICLE INFO

Article history:
Received 20 November 2015
Received in revised form 9 June 2016
Accepted 5 July 2016
Available online 11 July 2016

Keywords:
Deep learning
Machine vision
Automatic plant identification
Leaf vein image

ABSTRACT

We propose using a deep convolutional neural network (CNN) for the problem of plant identification from leaf vein patterns. In particular, we consider classifying three different legume species: white bean, red bean and soybean. The introduction of a CNN avoids the use of handcrafted feature extractors as it is standard in state of the art pipeline. Furthermore, this deep learning approach significantly improves the accuracy of the referred pipeline. We also show that the reported accuracy is reached by increasing the model depth. Finally, by analyzing the resulting models with a simple visualization technique, we are able to unveil relevant vein patterns.

1. Introduction

Nowadays, in many typical applications of machine vision there is a tendency to replace classical techniques with deep learning algorithms (LeCun et al., 2015). In deep learning, handcrafted feature extractors are unnecessary: typically, classification results are better than those obtained with classical techniques. Some successful examples can be found in Krizhevsky et al. (2012), Cireşan et al. (2013), and Taigman et al. (2014).

Deep learning refers to training neural network architectures composed of several nonlinear processing layers. The success of deep learning is based on new model regularization techniques (Srivastava et al., 2014), improved nonlinearities design (Dahl et al., 2013), and current hardware capabilities, among others. In particular, for Machine Vision tasks, the success of deep learning is based on convolutional neural networks (CNN, LeCun et al., 1990) which have become the standard neural network variant for image processing (LeCun et al., 2015).

There are many agricultural problems currently addressed by classical machine vision techniques that may benefit from using a deep learning approach. We consider in this paper a successful example of this behavior by applying deep learning to automatic plant identification.

Automatic plant identification constitutes a challenging problem that has received increasing attention in recent years, in particular for identification based on leaf image analysis. Much of this work makes use leaf features that humans can perceive. The goal of automatization in this case is to avoid the use of human experts handling huge catalogs of plant species, and to reduce classification time. Some works are focused on leaf shape (Agarwal et al., 2006; Camargo Neto et al., 2006; Chaki and Parekh, 2012; Du et al., 2007; Gwo et al., 2013; Im et al., 1998; Solé-Casals et al., 2008), some use shape and texture (Husin et al., 2012), while others consider color and texture (Pydipati et al., 2006).

Recently, however, more attention has been payed to vein morphological patterns as a leaf fingerprint. A clear correlation has been established between vein characteristics and some properties of the leaf (such as damage and drought tolerance, among others) (Sack et al., 2008; Scoffoni et al., 2011). This suggests that vein morphology carries information suitable for plant classification when shape, color or texture differences are unobservable, as in the case of trying to separate different cultivars from the same species. This kind of features may not be easily spotted by a human observer, and automated recognition becomes indispensable.

Following this premise, Larese et al. (2014a) applied computer vision techniques to extract several vein morphological measures, and showed that it is possible to separate three different plant species by using only the extracted information and supervised machine learning algorithms. In a later work (Larese et al., 2014b), they used similar techniques to reach some degree of discrimination between plants belonging to different cultivars of the same species.

In this work we discuss the use of deep learning models for interesting agricultural problems. As a working example, we apply this new paradigm to the problem of plant identification based on vein morphology. We show that the application of a standard deep convolutional network yields better results than those obtained

* Corresponding author.
E-mail address: uzal@cifasis-conicet.gov.ar (L.C. Uzal).

http://dx.doi.org/10.1016/j.compag.2016.07.003
0168-1699/

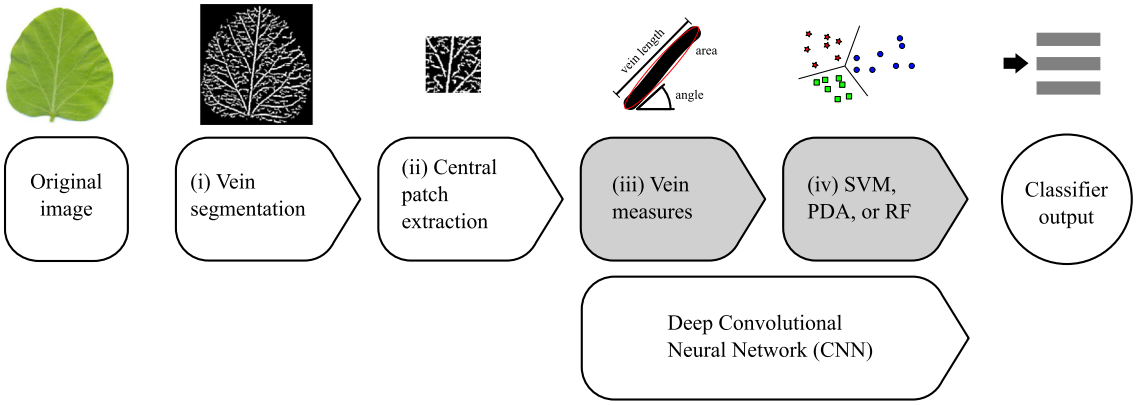

Fig. 1. Adopted pipeline (as in Larese et al. (2014b)). In this work, the grayed stages were replaced by a deep convolutional network. Stages (i) and (ii) were kept in order to allow a fair comparison with Larese et al. results. By design, these two stages filter color and leaf shape information. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

with a standard machine vision pipeline. Furthermore, the utilization of a simple model visualization technique allows us to identify meaningful vein patterns. The obtained results on plant classification from leaf vein morphology are not only valuable by themselves, but as a first step for motivating further research on the use of deep learning in agriculture.

The rest of the paper is organized as follows. In Section 2 we review the task-specific approach for the problem at hand as proposed by Larese et al. (2014b). In Section 3 we introduce the proposed deep learning methodology and explain the performed experiments in Section 4. Results are presented in Section 5, and then in Section 6 we show which patterns were deemed relevant for classification. Finally, we draw some conclusions in Section 8. In Appendix A, more detailed information about data acquisition and processing can be found.

2. Task-specific approach

Many successful Machine Learning applications make intensive use of specific knowledge about the task provided by human experts. In this section we summarize the approach considered in Larese et al. (2014b) for plant classification based on leaf veins, which makes use of expert knowledge.

The processing pipeline is divided into four stages as shown in Fig. 1. The starting point is the set of images of first foliage leaves acquired with a standard flatbed scanner (see Appendix A for more details). These images are processed according to the following stages:

(i) Vein Segmentation: first, an unconstrained version of the Hit or Miss Transform (UHMT) (Soille, 1999) is applied in order to extract vein morphological patterns. The output of this transform is a binary image—it therefore eliminates color information.
(ii) Central patch extraction: a central patch (100×100 pixels) of the binary image is cropped and the rest of the image is discarded. The purpose is to eliminate possible influences of the leaf shape.
(iii) Vein measures: at this stage, a set of features of interest was extracted with the help of LEAF GUI (Price et al., 2011). This set includes measures such as the total number of veins, total number of nodes, and mean vein width, among others.
(iv) Classification. Three different Machine Learning algorithms were tested: Support Vector Machines (SVM), Penalized Discriminant Analysis (PDA) and Random Forests (RF) (Hastie et al., 2009). These models were trained using the features obtained in the previous step.

There are two main observations about this pipeline. First, in order to highlight different levels of vein details, Larese et al. applied the UHMT to resized versions of the leaf image. The scale factors considered were 100% (no resize), 80%, and 60%. The processed images were resized back to the original size. With these three output images, two alternatives were studied. In the first one, a single combined image was obtained by adding them. For the second alternative, the three output images and the combined one were kept. We will refer to these two setups as S1 and S2 respectively. Fig. 2 shows some example images after stage (ii) for the S2 setup. The S1 setup correspond to selecting only the first column for each sample.

The second point to notice is that stage (iii) is the only one that requires specific domain knowledge. All considered measures can be automatically extracted but were specifically designed by experts to characterize vein patterns.

Also, it is important to remark the difference in the number of features extracted at stage (iii). Larese et al. extracted 52 features from each patch image. This means 52 features in the S1 setup, in contrast to 208 features in the S2 setup.

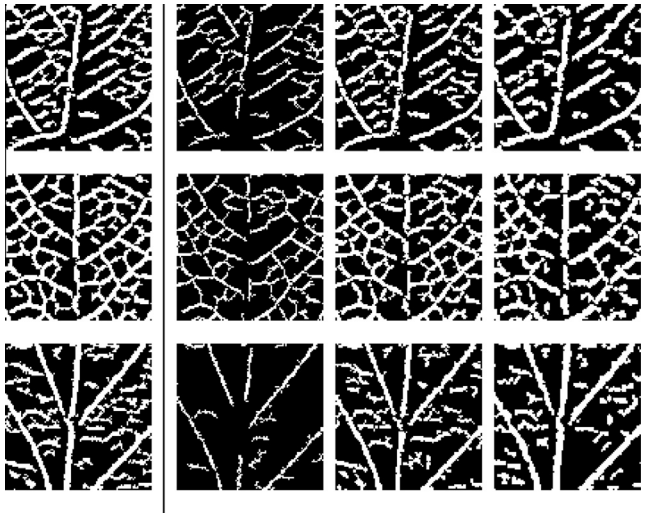

Fig. 2. Image samples obtained after processing stages (i) and (ii). The first column corresponds to preprocessing S1, while setup S2 is formed by all columns. These images are the input to the CNN.

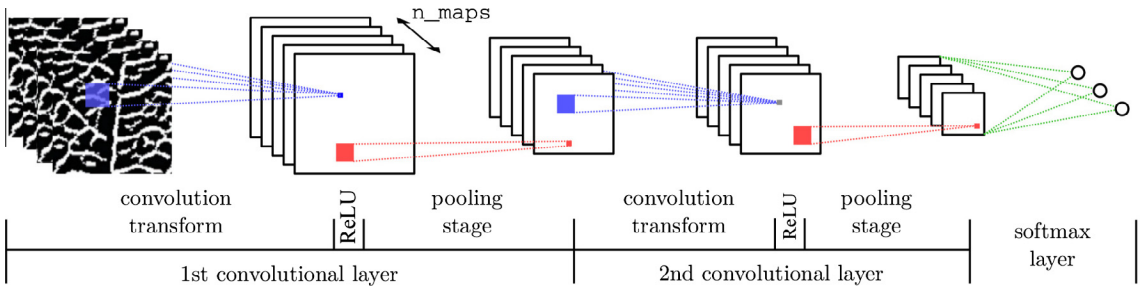

Fig. 3. Diagram of a three layer version of the CNN considered in this work. The first two are convolutional layers and the last one is a softmax layer. We evaluate CNNs up to 6 layers.

3. Deep Learning proposal

In this work we replace stages (iii) and (iv) of the previous pipeline (see Fig. 1) with a CNN (explained in the next subsection). We expect that the CNN can automatically learn from the training set the appropriate features to solve the classification problem. This implies that we can put aside the expert knowledge for feature design.

It should be noticed that stages (i) and (ii) ensure that color and leaf shape information is deleted. The reason for this is that the main objective in Larese et al. (2014b) was to show that it is possible to classify plant species only from vein morphological patterns. It should be remarked that this setup mimics the case where it is expected to find only morphological differences, as in the case of different cultivars from the same species.

In order to fairly compare our results with those from Larese et al. (2014b), we also include stages (i) and (ii) in our pipeline. That is, we consider the classification problem using only the vein morphological patterns, feeding the CNN with images as those shown in Fig. 2.

3.1. Convolutional neural networks

CNNs were first introduced by LeCun et al., 1990. They have an architecture specially designed to process images. In this section we give a brief introduction to CNNs, mainly with the purpose of defining the hyperparameters considered in this work. For a more detailed description of this kind of models, we refer the reader to LeCun et al. (2010) and references therein.

Fig. 3 depicts a diagram of the considered CNN model. Each layer is composed of three transforms. First, there is a convolution operation between the input image and a filter bank of size `n_maps`. Each filter has a bounded size associated to a small receptive field in the input image (blue squares in Fig. 3). We use square filters of width `filter_size`. For each filter in the bank, the convolution produces a feature map.

The second transform is an element-wise nonlinear function applied to all feature maps. Typically this function is a ReLU.[1]

Finally, there is a subsampling transform. At this pooling stage, each map is divided into a set of non-overlapping square neighborhoods of width `pool_size` (red squares). From each neighborhood, this transform only retains the maximum value.

The last layer in the network is a softmax function. It returns the estimated probability of each class, given a concrete sample. This layer is fully connected to all output feature maps of the last convolutional layer.

The described topology produces a huge reduction in the number of free trainable parameters in comparison to a standard (fully connected) artificial neural network. This is due to its sparse neural connectivity (restricted to small receptive fields) and to the sharing of filter values along image locations exploiting translational invariance.

A last relevant comment on CNNs is about visualization techniques. Recently published procedures (Zeiler and Fergus, 2014) allow the visualization of which patterns are detected at each layer of the deep network, partially removing the CNNs from the category of black-box models. In this paper we use a simple procedure considered in Zeiler and Fergus (2014) for highlighting the most relevant input image regions for the network output probabilities (see Section 6).

4. Experiments

We evaluated the proposed pipeline with a CNN stage by performing a set of numerical experiments in which several models were trained in order to estimate the test error. Below we give details of the performed experiments.

4.1. Datasets

We consider the leaf images already processed with stages (i) and (ii) of the original pipeline (Fig. 1) with the two setup variations S1 and S2 (as explained in Section 2) as input of our CNN without any further processing. For the S2 setup, in which we have four images per sample, they are considered as four different input channels for the CNN (as it is standard with the three channels in a RGB image). We have therefore two series of experiments associated with these two setups S1 and S2, as in Larese et al. (2014b). In summary, the input of the CNN is a $100 \times 100 \times 1$ binary image in the S1 setup and a $100 \times 100 \times 4$ one in S2.

4.2. Models

We trained CNN models of increasing depth: from 2 layers (1 conv. layer + 1 softmax) to 6 layers (5 conv. layers + 1 softmax). The architecture is the same for all convolutional layers in each model. This means fixed `filter_size` and `n_maps` for the convolution transform followed by a 2×2 pooling and a ReLU element-wise nonlinearity.[2]

4.3. Training algorithm

The parameters were optimized using stochastic gradient descent (SGD) over a training set using mini-batches of 20 samples. We used a 50% dropout rate (Srivastava et al., 2014) in the training stage for regularization. After some preliminary training experiments we set the learning rate at 0.01 and momentum to zero.

[1] $ReLU(x) = \max(0, x)$.

[2] The best model configuration obtained by cross validation for a 5-layer CNN has `filter_size` = `12` and `n_maps` = `10`. See Section 5.

4.4. Error estimation

For each model depth, the final test errors were estimated, as in Larese et al. (2014b), by averaging over 100 runs (10 independent runs of 10-fold cross-validation). The hyperparameters `filter_size` and `n_maps` were chosen such as to minimize the classification error over a validation set. We performed early stopping on the number of SGD iterations by monitoring the validation error.

4.5. Implementation

All experiments were carried out using Pylearn2 (Goodfellow et al., 2013).

5. Results

Figs. 4 and 5 show that the final accuracy consistently improves with the depth of the model, irrespective of the selected setup (S1 or S2).

For the S1 setup (Fig. 4) the best performance is reached at a depth of 5 layers, attaining a mean accuracy of $(92.6 \pm 0.2)\%$. This value surpasses the best mean accuracy reported in Larese et al. (2014b), which corresponds to a PDA classifier (90.4 ± 0.3, horizontal dashed line in Fig. 4). The reached accuracy does not improve upon addition of extra layers: there is no clear difference between 5- and 6-layer results.

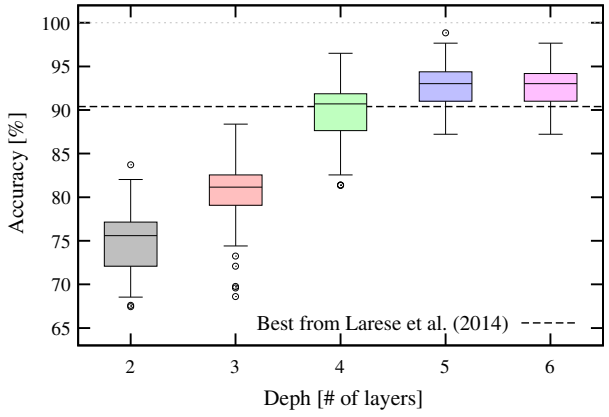

Fig. 4. Classification accuracy as a function of CNN depth for setup S1. Each box resumes the results from 100 runs (10 runs of 10-fold cross validation) for the corresponding depth. The box size covers the two central quartiles, while the whiskers span the central 95% of the runs. The horizontal dashed line indicates the best mean value obtained by a task-specific approach (Larese et al., 2014b) for this setup.

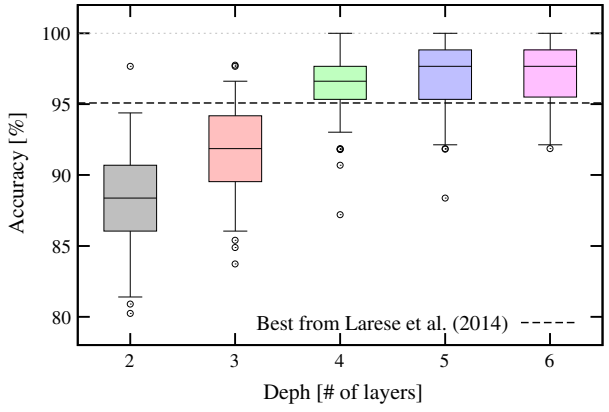

Fig. 5. Classification accuracy as a function of CNN depth for setup S2. See Fig. 4 caption for details.

Table 1
Per class and total accuracy (mean ± stdev). Values correspond to our CNN model and best accuracies reached in Larese et al. (2014b) for the S1 and S2 setups.

Model	White bean (%)	Red bean (%)	Soybean (%)	Total (%)
Combined veins only (S1 setup)				
PDA, best of Larese et al. (2014b)	82.7 ± 0.9	85.8 ± 0.6	96.5 ± 0.3	90.4 ± 0.3
CNN 5 layers	77.6 ± 1.1	93.8 ± 0.5	98.8 ± 0.2	93.0 ± 0.3
Combined veins with 3 scales (S2 setup)				
PDA, best of Larese et al. (2014b)	90.9 ± 0.6	91.7 ± 0.5	99.0 ± 0.1	95.1 ± 0.2
CNN 5 layers	90.2 ± 1.0	98.3 ± 0.3	98.8 ± 0.2	96.9 ± 0.2

For the S2 setup (Fig. 5) we find similar results. The best performance is also reached with the 5-layer model, with no clear difference with respect to the 6-layer results. As in Larese et al. (2014b) the accuracy levels with setup S2 are consistently higher than with setup S1. In this case the 5-layer model accuracy reaches a mean value of $(96.9 \pm 0.2)\%$, surpassing the $(95.1 \pm 0.2)\%$ best accuracy in Larese et al. (2014b).

Table 1 shows the mean accuracy for each class attained by the best models. The improved performance of the 5-layer model under both setups, S1 and S2, mainly originates from a better classification of red bean leaves.

For the S1 setup the accuracy of the 5-layer model over white bean leaves drops significantly with respect to the best model in Larese et al. (2014b). However, as white bean is the least numerous class, this drop is insufficient to change the overall result. This behavior vanishes for the S2 setup, where the accuracy of the 5-layer model is similar or better for all classes.

Finally, for completeness, we show Receiver Operating Characteristic (ROC) curves for the S2 setup in order to further evaluate classifier output quality. Fig. 6 shows one-vs-all ROC curves for each class, for the different models (from 2 to 6 layers). Consistently, white beans are harder to classify for all depths. However, we found that all curves improve with depth up to the 5-layer model. The bottom left panel shows average curves for different depth models where the 6-layer model falls below the 5-layer model curve. It is worth to mention that CNNs typically respond with nearly saturated probabilities, and ROC curves are less useful in these cases.

6. Visualizing relevant patterns

In order to gain an insight into which input patterns an already trained CNN is sensitive to for class labeling we perform an experiment along the lines of Zeiler and Fergus (2014). We occlude different parts of an input image with a 10×10 black patch and observe the variation in the output probability for the correct class.[3] The result is a heatmap indicating the locations of the image the output is more sensitive to. Fig. 7 shows some example images for the three classes. Red colored regions correspond to a decrease in the output probability while green indicates an increase. As all shown examples were correctly classified, this occlusion procedure typically lowers the probability of the correct class, biasing the heatmap results to red as observed. We can see that different parts of the vein patterns are taken into account for each class. For example, in the case of the soybean samples, the most informative regions are

[3] We used a trained 5-layer CNN with `filter_size = 12` and `n_maps = 10` in the S2 setup.

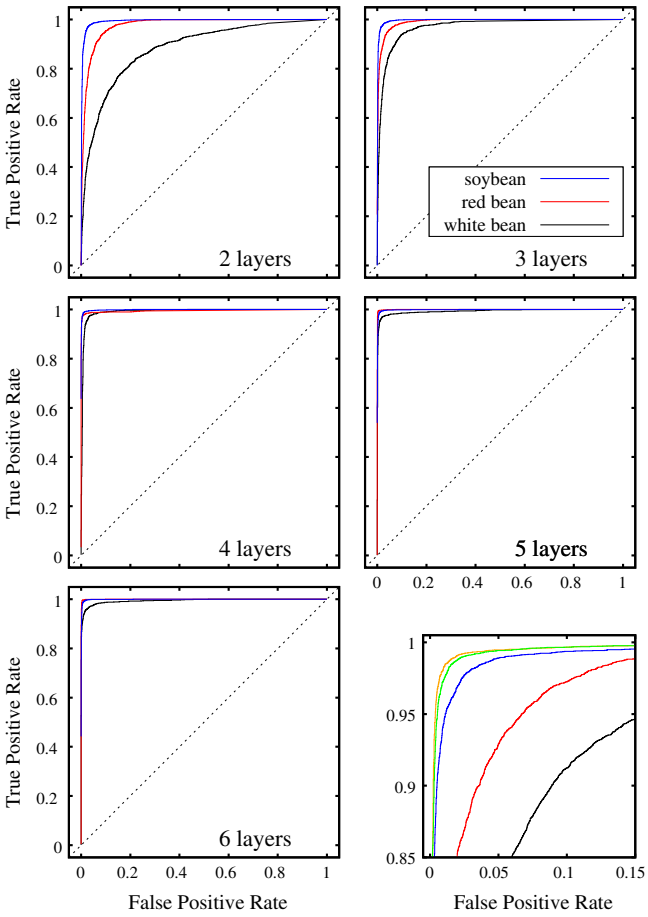

Fig. 6. Receiver Operating Characteristic (ROC) curves for the S2 setup. The first 5 panels show one-vs-all ROC curves for each class for the different models (from 2 to 6 layers). The bottom left panel shows average curves for different depths (black, 2 layers; red, 3 layers; blue, 4 layers; orange, 5 layers; and green, 6 layers). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

located in the central vein and, in particular, in the places where the secondary veins join to it. However, for white and red bean classes, the results suggest that the outer and smaller veins are also relevant for the network output. In these highlighted regions we expect to find the patterns relevant for the task at hand.

Some selected samples shown in Fig. 8 help us understand the tradeoff between the output probabilities for the three classes. Rows (a) and (b) show two sample images from white bean leaves, while rows (c) and (d) correspond to a single sample image from red bean and soybean, respectively. Each column shows the variation of the output probabilities of the model associated to each class when performing the same procedure as in Fig. 7. Given that these probabilities sum up to 1, a decrease in one class (red regions) is associated to an increase in the others (green regions).

The first observation we can derive from rows (a) and (b) is that by occluding the higher order veins in the left side of the image the correct class output probability drops (first column) in favor of soybean probability (third column). This is because the higher order veins in soybean leaves are sparser than in the case of red and white beans (see samples in Fig. 7). Another observation we can make concerns the central vein in row (b). For this particular white bean, lower order veins look similar to those of a soybean with respect to its ramification angle. This is the reason why when we occlude parts of the central vein the probability of the correct class increases at the expense of the soybean probability.

Concerning row (c), the selected red bean sample shows a pattern typically found in this class. The cell-like pattern of the higher order veins seems to constitute a subtle difference with respect to the white bean class. Therefore, when occluding these patterns the probability of red bean decreases (second column) in favor of the white beans.

Rows (c) and (d) also show a typical behavior of the red bean and soybean classes. Both classes interact with the white bean class in the probability exchange but they do not interact with each other. This suggests that these two classes do not have a frontier in common in the representation computed by the deep network at the input to the softmax layer.

7. Overall efficiency

When comparing two approaches for solving a specific task it is advisable to consider the overall efficiency, i.e., the ratio between the quality measure of the results and the cost of the resources employed to achieve them. In Section 5 we present a comparative summary in terms of classification accuracy. The counterpart of resources spent is harder to compare given that it requires to measure the time and effort spent in the design and training of neural models vs. the manual design and selection of feature extractors, and issuing a classification. However, there are a few points we can make in order to ease the comparison.

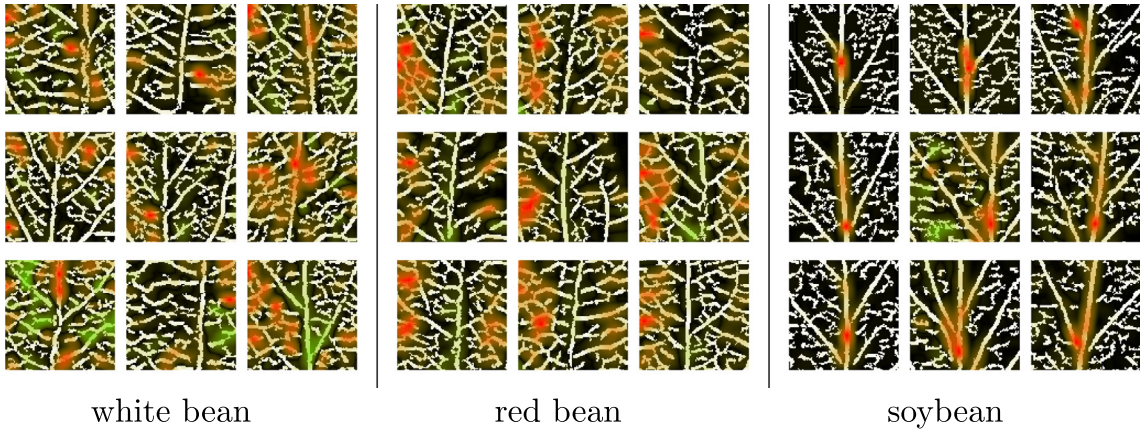

Fig. 7. Heatmaps indicating the input image region to which the correct-class output probability of the network is more sensitive to partial occlusion. Red colored regions correspond to a decrease of the output probability while green indicates an increase. Each panel shows 9 examples from each class. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

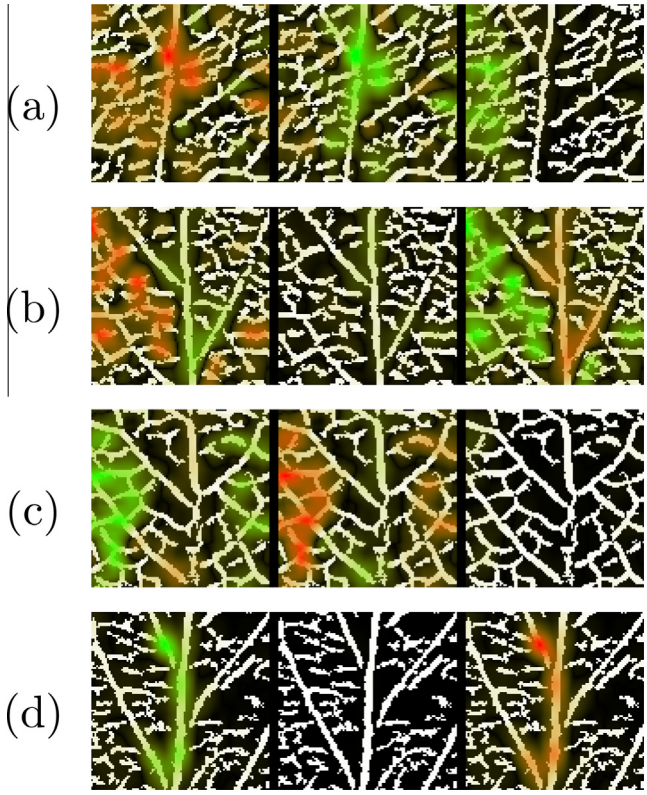

Fig. 8. Heatmaps indicating the locations within the input images to which each class output probability is more sensitive to partial occlusion. The columns correspond to each of the three outputs (white bean, red bean, and soybean). Red colored regions indicate a decrease in the corresponding output probability, while green denotes an increase. Rows (a) and (b) correspond to white bean images, (c) to red bean, and (d) to soybean. For these examples, the variation of the probability assigned to the correct class is also shown in Fig. 7. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Designing feature extractors is labor-intensive and requires expert skills. We lack data about the resources employed in Ref. Price et al. (2011) to design leaf vein features. Nevertheless, for most vision applications, suitable feature extractors are publicly available as in this case. The effort is transferred however to the search and selection of proper feature extractors for the task, within a very large corpus of machine vision literature. This learning process is the most time consuming task and can demand from weeks to months, depending on the problem at hand. This selection process includes the evaluation of classification models, which can be done with a predefined application agnostic pipeline. The comparison of different classification algorithms (SVM, PDA, RF) and selection of hyperparameters by cross validation can be automated and requires just a few hours of computation on a standard CPU.

On the other hand, for our proposal, we use a general purpose standard CNN configuration (convolutional layer + ReLU activation + dropout, LeCun et al., 2015). An implementation of such network can be found as a toy example in any deep learning library (Chollet, 2015; Jia et al., 2014; Goodfellow et al., 2013). We added no special features to adapt this standard model to our particular application. As a consequence, training our model on the leaf vein dataset to reach competitive validation accuracies took us a few hours. The rest of the time (around one week[4]) was spent designing and running cross validation experiments (as in any classification task) for parameter optimization and error estimation, which is only hardware intensive.

[4] The error estimation protocol (Section 4) required us around 1000 single training runs of 30 min. each on one of four available low-cost GPUs.

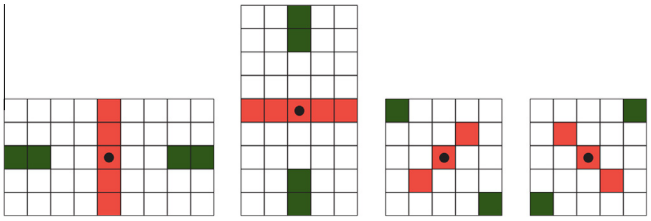

Fig. 9. The four pairs of flat composite structuring elements used in the UHMT computation to detect veins in four directions, from left to right: vertical, horizontal, +45°, and −45°. Foreground pixel configurations are depicted in red while background pixel configurations in green. The center of the composite structuring element is marked with a black dot. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

In summary, we cannot precisely state which pipeline is less time consuming. However, the deep learning approach eliminates the manual search for good feature extractors by automatically learning relevant features. This conveniently transfers human expert time to computational burden and, for this specific application, also yields improved accuracies.

8. Conclusions

In this work we report a successful application of deep learning to the area of agriculture, specifically to plant identification from leaf vein patterns. We replaced a task-specific module in a state of the art processing pipeline with a deep convolutional network. The main result is that we obtained an improved accuracy using a standard deep learning model. This implies that it is not necessary to handcraft a specific feature extraction method for this task. We also show that accuracy monotonically improves with model depth, which implies that depth is a key ingredient of the outperformance.

It is commonly argued that neural networks do not provide any new insight about the problem under study, as they fall into the category of black box models. However, thanks to a simple visualization technique, we were able to obtain and understand the relevant vein patterns used by the deep model for the classification task performed.

We are currently exploring other potential applications of deep learning in agriculture, in particular weeds detection and identification, and seeds viability tests.

Acknowledgment

The authors thank P. F. Verdes for valuable comments and acknowledge grant support from ANPCyT PICT-2012-0181.

Appendix A. Dataset

In this work we use the dataset introduced by Larese et al. (2014a). It is formed by 866 leaf images provided by INTA (Instituto Nacional de Tecnología Agropecuaria, Oliveros, Argentina). It is divided into three classes: 422 images correspond to soybean leaves, 272 to red bean leaves and 172 to white bean leaves. These are first foliage leaves after 12 days of seedling growth. The images were acquired using a standard flatbed scanner (Hewlett Packard Scanjet-G 3110) at a resolution of 200 pixels per inch. The images correspond to the abaxial surface of the leaves.

Appendix B. Processing pipeline details

In this appendix we describe in more detail the first two stages of the processing pipeline, i.e., the stages that we kept from the method proposed in Larese et al. (2014a).

(i) The color information was removed by converting the RGB images to grayscale.
(ii) A binary mask was obtained for each leaf, using the automatic iterative threshold selection algorithm (Sonka et al., 2014). Its holes were filled using morphological reconstruction (Soille, 1999) and finally all connected components except the largest one were removed.
(iii) An unconstrained version of the Hit or Miss Transform (UHMT) (Soille, 1999) on five differently sized (100%, 90%, 80%, 70% and 60%) versions of the images were computed. For this purpose, four composite structuring elements were used aimed at detecting leaf veins in four directions: vertical, horizontal, +45° and −45° (shown in Fig. 9).
(iv) The resulting UHMTs were resized back to their original sizes and added to obtain the combined UHMT. The resized UHMTs at 100%, 80% and 60% were also preserved.
(v) An adaptive histogram equalization and adaptive thresholding was performed, and all connected components with less than 20 pixels were removed.
(vi) The UHMTs were masked using the result of step (ii).
(vii) Finally, a central patch (100×100 pixels) of the binary image is cropped and the rest of the image is discarded. Since the images are taken with a resolution of 200 dpi, this corresponds to a square of 0.5×0.5 in. approximately located at the center of the leaf.

References

Agarwal, G., Ling, H., Jacobs, D., Shirdhonkar, S., Kress, W., Russell, R., Belhumeur, P., Dixit, N., Feiner, S., Mahajan, D., Sunkavalli, K., White, S., 2006. First steps toward an electronic field guide for plants. Taxon J. Int. Assoc. Plant Taxon. 55, 597–610.

Camargo Neto, J., Meyer, G.E., Jones, D.D., Samal, A.K., 2006. Plant species identification using Elliptic Fourier leaf shape analysis. Comput. Electron. Agric. 50, 121–134.

Chaki, Jyotismita, Parekh, Ranjan, 2012. Designing an automated system for plant leaf recognition. Int. J. Adv. Eng. Technol. 2 (1), 149–158.

Chollet, François, 2015. Keras. <https://github.com/fchollet/keras>.

Cireşan, Dan C., Giusti, Alessandro, Gambardella, Luca M., Schmidhuber, Jürgen, 2013. Mitosis detection in breast cancer histology images with deep neural networks. In: Medical Image Computing and Computer-Assisted Intervention–MICCAI 2013. Springer, pp. 411–418.

Dahl, George E., Sainath, Tara N., Hinton, Geoffrey E., 2013. Improving deep neural networks for LVCSR using rectified linear units and dropout. In: 2013 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, pp. 8609–8613.

Du, Ji-Xiang, Wang, Xiao-Feng, Zhang, Guo-Jun, 2007. Leaf shape based plant species recognition. Appl. Math. Comput. 185 (2), 883–893, Special Issue on Intelligent Computing Theory and Methodology.

Goodfellow, Ian J., Warde-Farley, David, Lamblin, Pascal, Dumoulin, Vincent, Mirza, Mehdi, Pascanu, Razvan, Bergstra, James, Bastien, Frédéric, Bengio, Yoshua, 2013. Pylearn2: A Machine Learning Research Library. Available from: arXiv preprint arXiv:1308.4214.

Gwo, Chih-Ying, Wei, Chia-Hung, Li, Yue, 2013. Rotary matching of edge features for leaf recognition. Comput. Electron. Agric. 91, 124–134.

Hastie, T., Tibshirani, R., Friedman, J., 2009. The Elements of Statistical Learning, second ed. Springer.

Husin, Z., Shakaff, A.Y.M., Aziz, A.H.A., Farook, R.S.M., Jaafar, M.N., Hashim, U., Harun, A., 2012. Embedded portable device for herb leaves recognition using image processing techniques and neural network algorithm. Comput. Electron. Agric. 89, 18–29.

Im, Cholhong, Nishida, Hirobumi, Kunii, Tosiyasu L., 1998. Recognizing plant species by leaf shapes-a case study of the Acer family. Int. Conf. Pattern Recogn. 2, 1171.

Jia, Yangqing, Shelhamer, Evan, Donahue, Jeff, Karayev, Sergey, Long, Jonathan, Girshick, Ross, Guadarrama, Sergio, Darrell, Trevor, 2014. Caffe: Convolutional Architecture for Fast Feature Embedding. Available from: arXiv preprint arXiv:1408.5093.

Krizhevsky, Alex, Sutskever, Ilya, Hinton, Geoffrey E., 2012. Imagenet classification with deep convolutional neural networks. In: Advances in Neural Information Processing Systems, pp. 1097–1105.

Larese, Mónica G., Namías, Rafael, Craviotto, Roque M., Arango, Miriam R., Gallo, Carina, Granitto, Pablo M., 2014a. Automatic classification of legumes using leaf vein image features. Pattern Recogn. 47 (1), 158–168.

Larese, Mónica G., Bayá, Ariel E., Craviotto, Roque M., Arango, Miriam R., Gallo, Carina, Granitto, Pablo M., 2014b. Multiscale recognition of legume varieties based on leaf venation images. Exp. Syst. Appl. 41 (10), 4638–4647.

LeCun, Yann, Boser, Bernhard E., Denker, John S., Henderson, Donnie, Howard, R.E., Hubbard, Wayne E., Jackel, Lawrence D., 1990. Handwritten Digit Recognition with a Back-Propagation Network. In: Touretzky, D.S. (Ed.), Advances in Neural Information Processing Systems 2. Morgan-Kaufmann, pp. 396–404.

LeCun, Yann, Kavukcuoglu, Koray, Farabet, Clément, 2010. Convolutional networks and applications in vision. In: Proceedings of 2010 IEEE International Symposium on Circuits and Systems (ISCAS). IEEE, pp. 253–256.

LeCun, Yann, Bengio, Yoshua, Hinton, Geoffrey, 2015. Deep learning. Nature 521 (7553), 436–444.

Price, Charles A., Symonova, Olga, Mileyko, Yuriy, Hilley, Troy, Weitz, Joshua S., 2011. Leaf extraction and analysis framework graphical user interface: segmenting and analyzing the structure of leaf veins and areoles. Plant Physiol. 155 (1), 236–245.

Pydipati, R., Burks, T.F., Lee, W.S., 2006. Identification of citrus disease using color texture features and discriminant analysis. Comput. Electron. Agric. 52, 49–59.

Sack, L., Dietrich, E.M., Streeter, C.M., Sanchez-Gomez, D., Holbrook, N.M., 2008. Leaf palmate venation and vascular redundancy confer tolerance of hydraulic disruption. PNAS USA 105, 1567–1572.

Scoffoni, C., Rawls, M., McKown, A.D., Cochard, H., Sack, L., 2011. Decline of leaf hydraulic conductance with dehydration: relationship to leaf size and venation architecture. Plant Physiol. 156, 832–843.

Soille, P., 1999. Morphological Image Analysis: Principles and Applications. Springer-Verlag.

Solé-Casals, Jordi, Travieso, Carlos M., Alonso, Jesús B., Ferrer, Miguel A., 2008. Improving a leaves automatic recognition process using PCA. In: IWPACBB, pp. 243–251.

Sonka, Milan, Hlavac, Vaclav, Boyle, Roger, 2014. Image Processing, Analysis, and Machine Vision. Cengage Learning.

Srivastava, Nitish, Hinton, Geoffrey, Krizhevsky, Alex, Sutskever, Ilya, Salakhutdinov, Ruslan, 2014. Dropout: a simple way to prevent neural networks from overfitting. J. Mach. Learn. Res. 15 (1), 1929–1958.

Taigman, Yaniv, Yang, Ming, Ranzato, Marc'Aurelio, Wolf, Lars, 2014. Deepface: closing the gap to human-level performance in face verification. In: 2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, pp. 1701–1708.

Zeiler, Matthew D., Fergus, Rob, 2014. Visualizing and understanding convolutional networks. In: Computer Vision–ECCV 2014. Springer, pp. 818–833.