Fine-tuning Deep Convolutional Networks for Plant Recognition

Angie K. Reyes[1], Juan C. Caicedo[2], and Jorge E. Camargo[1]

[1] Laboratory for Advanced Computational Science and Engineering Research, Universidad Antonio Nariño, Colombia
angreyes,jorgecamargo{@uan.edu.co},
[2] Fundación Universitaria Konrad Lorenz, Colombia
juanc.caicedor@konradlorenz.edu.co

Abstract. This paper describes the participation of the ECOUAN team in the LifeCLEF 2015 challenge. We used a deep learning approach in which the complete system was learned without hand-engineered components. We pre-trained a convolutional neural network using 1.8 million images and used a fine-tuning strategy to transfer learned recognition capabilities from general domains to the specific challenge of Plant Identification task. The classification accuracy obtained by our method outperformed the best result obtained in 2014. Our group obtained the 4th position among all teams and the 10th position among 18 runs.

Keywords: Plant recognition, deep learning, convolutional neural networks, image retrieval, imageCLEF

1 Introduction

Systems that assist professionals to recognize the species and categories of plants are important. The ImageCLEF challenge includes a task for plant identification that brings together researchers to study the problem and propose and exchange ideas and methods.

As in any other image recognition task, plant identification relies on computational methods to extract discriminative features from images. Features have been traditionally hand-crafted or hand-engineered. However, a recent trend in machine learning has demonstrated that learned representations are more effective and efficient. The main advantage of representation learning is that algorithms automatically analyze large collections of images and identify features that can categorize images with minimum error. How can we adapt these strategies for plant identification.

In this work we propose to use deep Convolutional Neural Networks (CNNs) to extract features and classify images at the same time. The proposed CNN is pre-trained in a large collection of generic web images and fine-tuned to recognize plants with the highest possible accuracy. Our approach is an end-to-end learning strategy, with minimum assumptions about the contents of images.

Using our strategy we achieved 0,487 precision in the main challenge of LifeCLEF 2015 improving the best result obtained by IBM Research Australia in LifeCLEF 2014 [10]. We submitted one entry that obtained the 10th best run and we obtained the 4th position among all the team participants.

Several groups participated in the LifeCLEF 2014 challenge, with a variety of techniques and results. These strategies used by the 3 best teams were: (1) The IBM Research Australia team [10] proposed a method based on the extraction of visual features (Fisher Vector) and linear classifiers as classification strategy. They also followed a Deep Learning approach in one of the runs; (2) The INRIA Pl@ntNet team [6] proposed a fusion strategy that combines a set of local descriptors to represent images and KNN as classification strategy; and (3) The BME TMIT team [4] proposed a strategy based on dense SIFT for feature detection and description, and a Gaussian Mixture Model based on Fisher Vector. They used the C-support vector classification algorithm as classification method.

The rest of this paper is organized as follows: section 2 presents the plant identification task; section 3 describes the details of our approach; section 4 presents conducted experiments and obtained results; and section 5, concludes the paper.

2 The Plant Identification Task

The Plant Identification task is oriented to recognize the species of a plant given an observation. An observation is a set of images (1 to 5) that capture the appearance of the plant from different perspectives or points of view. These points of view include entire plant, branch, leaf, fruit, flower, stem or leaf scan. It is important to note that images of an observation were acquired by the same person the same day. Images of an observation belong to one of 1,000 possible species (last year the total number of species were 500).

The plant identification task was based on the Pl@ntView dataset. It focuses on 1,000 herb, tree and fern species centered on France and neighboring countries, which contains 113,205 pictures. Contrary to previous plant identification challenges, queries are not defined as single images but as plant observations. That is to say, a query is composed of 1 to 5 images belonging to the same plant species.

Each participating team was allowed to submit up to 4 runs built from different methods. Semi-supervised and interactive approaches, particularly for segmenting leaves from the background, were allowed but were compared independently from fully automatic methods. Any human assistance in the processing of the test queries had therefore to be signaled in the submitted runs.

3 Proposed Approach

We designed a plant identification system using deep learning at its core. The proposed system is learned end-to-end, without hand-engineered components. This section presents the computational details of our approach.

3.1 Deep Convolutional Networks

A Convolutional Neural Network (CNN) is a stack of non-linear transformation functions that are learned from data. CNNs were originally proposed in the 1980's for digit recognition [9], and have been recently revisited for large scale recognition problems. The success of modern CNNs relies on several factors that include: availability of large datasets, more computing power and new ideas and algorithms. Among the most successful ideas that make CNNs a powerful tool for image recognition nowadays is the concept of deep architectures [8, 11].

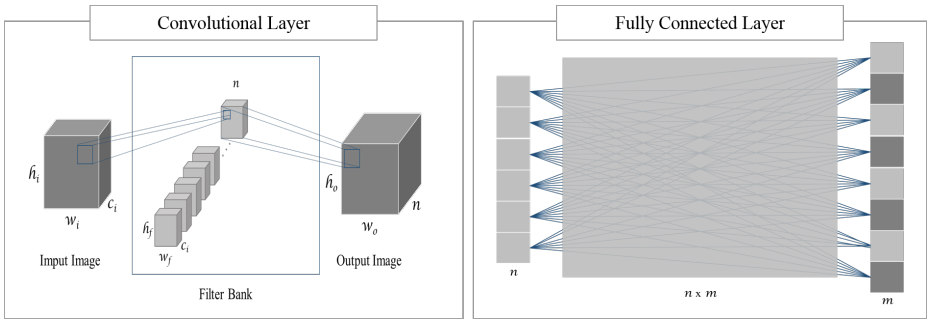

Fig. 1. Illustration of the types of layers in a CNN. A convolutional layer is a bank of n learned filters that are applied to an input image. The output is another image with the response of the n filters. A fully-connected layer takes a vector as input and transforms it with a non-linear function with parameters in the corresponding connection matrix.

A deep CNN has multiple layers that progressively compute features from input images. The deep architecture proposed by Krishevsky et al. [8] demonstrated the power of deep CNNs for the first time in a large scale image classification setting. There are mainly two types of layers in this network: convolutional layers and fully connected layers. Convolutional layers may be understood as banks of filters that transform an input image into another image, highlighting specific patterns. On the other hand, fully connected layers take a vector as input and produce another vector as output. Figure 1 illustrates the concept of convolutional and fully connected layers.

In our work, we use a CNN following the architecture proposed by Krishevsky et al. [8] which has 5 convolutional layers and 2 fully-connected layers. An additional prediction layer is added to the top of the network to obtain classification

scores given the learned image representation. This network is illustrated in Figure 2. The entire network is learned from data using the back-propagation algorithm. We pre-train this network on the ILSVRC 2012 dataset [2], which consists of 1.2 million images of 1,000 different categories. In this way, the network is initialized with a powerful and generic set of features that can recognize a variety of objects in images with low error rate.

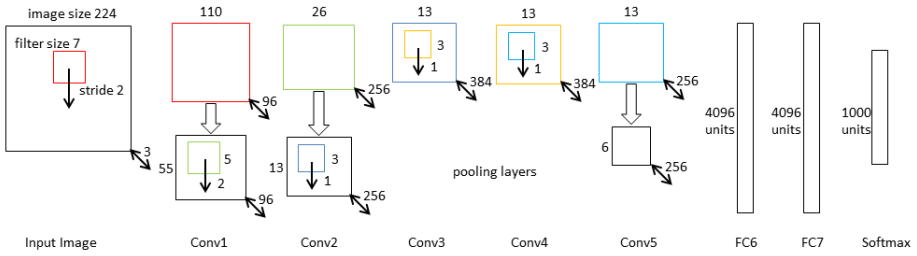

Fig. 2. Architecture of the CNN used in this work: 5 convolutional layers (conv), 2 fully-connected (fc) layers and 1 classification layer (softmax). Colored squares show the size of the output in the corresponding layer. Small squares with a downwards arrow indicate the size of filters. Diagonal arrows indicate the number of channels in the output of each convolutional layer.

3.2 Fine-tuning the CNN

We initialized the CNN to recognize 1,000 categories of generic objects that are part of the ImageNet hierarchy following the procedure described in the previous section. Then, we proceed to finetune the network for the Plant Identification task.

Fine-tuning a network is a procedure based on the concept of *transfer learning* [1, 3]. We start training a CNN to learn features for a broad domain with a classification function targeted at minimizing error in that domain. Then, we replace the classification function and optimize the network again to minimize error in another, more specific domain. Under this setting, we are transferring the features and the parameters of the network from the broad domain to the specific one.

The classification function in the original CNN is a softmax classifier that computes the probability of 1,000 classes of the ImageNet dataset. To start the fine-tuning procedure, we remove this softmax classifier and initialize a new one with random values. The new softmax classifier is trained from scratch using the back-propagation algorithm with data from the Plant Identification task, which also has 1,000 different categories.

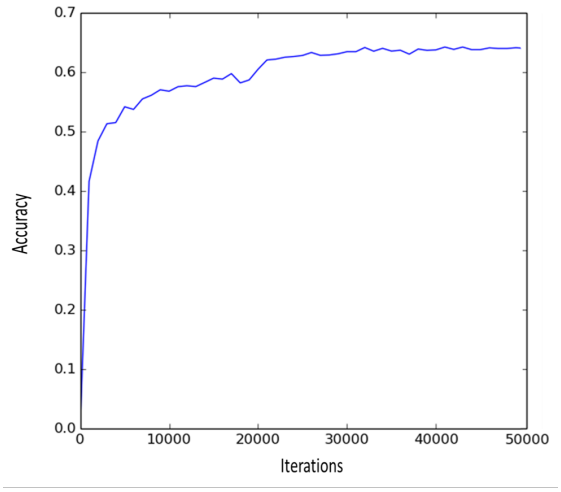

Fig. 3. Evolution of image classification accuracy in a validation set during the fine-tuning process. Accuracy improves quickly during the first iterations and stabilizes after 40,000 iterations.

In order to start the back-propagation algorithm for fine-tuning, it is key to set the learning rates of each layer appropriately. The classification layer, i.e., the new softmax classifier, needs a large learning rate because it has been just initialized with random values. The rest of the layers need a relatively small learning rate because we want to preserve the parameters of the previous network to transfer that knowledge into the new network. However, notice that the learning rate is not set to zero in the rest of the layers: they will be optimized again at a slower pace.

In our experiments we set the learning rate of the top classification layer to 10, while leaving the learning rate of all the other seven layers to 0.1. We run the back-propagation algorithm for 50,000 iterations, which optimizes the network parameters using stochastic gradient descent (SGD). Figure 3 shows how the precision of classifying single images improves with more iterations. Our implementation is based on the open source Deep Learning library Caffe [7], and we run the experiments using a NVIDIA Titan Z GPU (5,760 cores and 12 GB of RAM).

3.3 Prediction on Observations

An observation in the Plant Identification task is a set of 3 to 5 images on average in the training set. The system is required to produce a single prediction given these images, and the fine-tuned CNN discussed before is able to make predictions for a single input image. Therefore, we need to aggregate predictions from multiple images to produce a single output for the whole observation.

We used a simple combination strategy. The output of the fine-tuned network is a 1,000-dimensional vector with probability values of all categories, and the input is a single image. To combine the predictions of all images in an observation we compute the sum of all predicted vectors. This can be understood as a voting scheme, as we aggregate evidence from multiple images in a single vector without any further processing step. Finally, we assign the category to the input observation by identifying the class with maximum score in the aggregated vector.

3.4 Training and Test Datasets

We used the ILSVRC 2012 dataset to pre-train the proposed CNN. This dataset is composed of 1.2 million images of 1,000 different object categories. To fine-tune the network, we used the released training set of LifeCLEF 2015 [5], which has 91,759 images distributed in 13,887 plant-observation-queries with examples of 1,000 species that include trees, herbs, and ferns, among others.

The test set is composed of 21,446 images organized in 13,887 observations: an average of 1.5 images per observation.

Table 1. Distribution of images in the Plant dataset for training and test.

View	**Training**	**Test**
Branch	8,130	2,088
Entire	16,235	6,113
Flower	28,225	8,327
Fruit	7,720	1,423
Leaf	13,367	2,690
Stem	5,476	584
Scans	12,605	221
Total	**91,758**	**21,446**

Images in the training and test sets may have been taken from 7 different views. These 7 views are: entire plant, branch, flower, fruit, leaf, stem and scans. An observation can have one of many of these views taken by the same person. Table 1 presents the distribution of images in the training and test sets according to viewpoint, and Figure 4 shows some example pictures.

4 Experiments and Results

We submit one run with the results of the proposed system. We only trained and fine-tuned one network with the architecture described in previous sections. To adjust the hyper-parameters of the network, we separated a validation set with 1% of the training images. After identifying the correct parameters in the validation set, we fine-tuned the network again with all images.

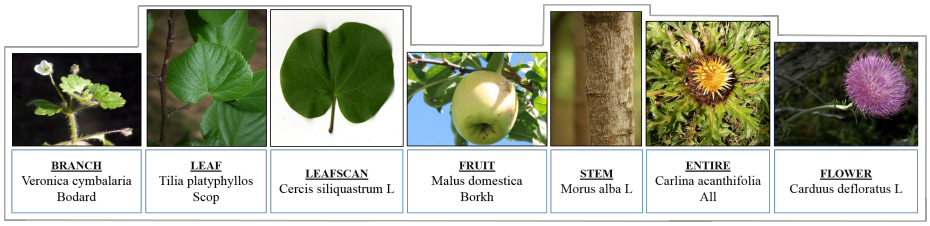

Fig. 4. Example images of the seven views taken from the training set.

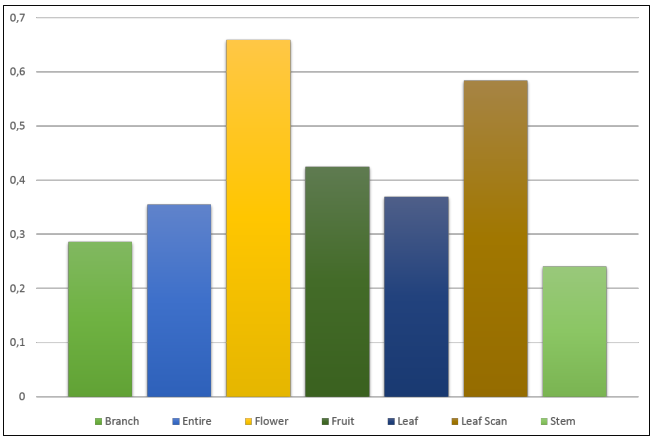

Fig. 5. Accuracy of our result in the test set by each of the seven views.

The results of our experiment are presented in Figure 5, which shows the accuracy of our method in each of the 7 possible views of a plant. The proposed system can categorize images of flowers and leaf scans with higher accuracy than the rest of the views. Our system obtained an average precision of 0.486 when classifying single images in the test set.

Table 2 compares our result with results of other participating groups in the challenge. This table presents the best experiment of each group with the scores obtained for single image classification and full observation identification. Our run was placed 10 among 18 total experiments, obtaining the 4th place for our group in the competition.

Notice that our experiment involved no special prior knowledge about plants or hand-engineered methods tailored for plant identification. We learned an image classification system end-to-end, using only publicly available training data. Our result is significantly better than the next result in the table, obtaining 42% better accuracy. Also, our result is 20% worse than the result above, explained mainly by the absence of plant specific strategies in our pipeline.

Table 2. Best result of each participating group in the 2015 challenge (accuracy). Our experiment was number 10 out of 18 submitted runs, leaving our group in the 4th place in the competition.

Research group	**Run name**	**Single Image**	**Observation**
SNUMED INFO	Run 4	0,652	0,667
QUT RV	Run 3	0,590	0,624
INRIA ZENTH	Run 1	0,581	0,609
ECOUAN (ours)	Run 1	0,486	0,487
MICA	Run 2	0,194	0,209
SABANCI	Run 1	0,153	0,160
UAIC	Run 1	0,013	0,013

5 Conclusions

This paper presented a system for plant identification based on Deep Convolutional Neural Networks. The proposed training strategy allows the system to learn all layers end-to-end from data, without involving techniques that are specific to plants. We believe that involving more domain knowledge in the design of the system can be beneficial to improve accuracy. x The fine-tuning strategy demonstrated to be a good solution to transfer learned recognition capabilities from general domains to the specific challenge of Plant Identification task. This is useful to take advantage of big visual data available on the Internet, and then transfer general recognition abilities to specific domains. In our future work we plan to evaluate deeper architectures of the CNN and adaptations of domain specific techniques to improve performance even further.

6 Acknowledgements

We gratefully acknowledge the support of NVIDIA Corporation with the donation of the Titan Z GPU used in this paper. This research was partially funded by BIGDATA SOLUTIONS S.A.S.

References

1. Bengio, Y.: Deep learning of representations for unsupervised and transfer learning. Unsupervised and Transfer Learning Challenges in Machine Learning, Volume 7 p. 19 (2012)
2. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on. pp. 248–255. IEEE (2009)
3. Donahue, J., Jia, Y., Vinyals, O., Hoffman, J., Zhang, N., Tzeng, E., Darrell, T.: Decaf: A deep convolutional activation feature for generic visual recognition. arXiv preprint arXiv:1310.1531 (2013)
4. Gbor Szcs, Dvid Papp, D.L.: Viepoints combined classification method in image-based plant identification task. In: Working notes ImageCLEF 2014. pp. 763–770. ImageCLEF (2014)
5. Herve Goau, A.J., Bonnet, P.: Lifeclef plant identication task 2015. In: CLEF working notes, 2015. (2015)
6. Herv Goau, Alexis Joly, I.Y.: Plantnet participation at lifeclef2014 plant identificatin task. In: Working notes ImageCLEF 2014. pp. 724–737. ImageCLEF (2014)
7. Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., Guadarrama, S., Darrell, T.: Caffe: Convolutional architecture for fast feature embedding. In: Proceedings of the ACM International Conference on Multimedia. pp. 675–678. ACM (2014)
8. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Advances in neural information processing systems. pp. 1097–1105 (2012)
9. LeCun, Y., Boser, B., Denker, J.S., Henderson, D., Howard, R.E., Hubbard, W., Jackel, L.D.: Backpropagation applied to handwritten zip code recognition. Neural computation 1(4), 541–551 (1989)
10. Qiang Chen, Mani Abedini, R.G.X.L.: Ibm research australia at lifeclef2014: Plant identification task. In: Working notes ImageCLEF 2014. pp. 693–704. ImageCLEF (2014)
11. Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A.: Going deeper with convolutions. arXiv preprint arXiv:1409.4842 (2014)