

Sabancı-Okan System in LifeCLEF 2015 Plant Identification Competition

Mostafa Mehdipour Ghazi¹, Berrin Yanikoglu¹, Erchan Aptoula²,
Ozlem Muslu¹, and Murat Can Ozdemir¹

¹ Faculty of Engineering and Natural Sciences, Sabanci University, Istanbul, Turkey

² Computer Engineering Department, Okan University, Istanbul, Turkey
{mehdipour,berrin,ozlemmuslu,ozdemirmcan}@sabanciuniv.edu
erchan.aptoula@okan.edu.tr

Abstract. We present our deep learning based plant identification system in the LifeCLEF 2015. The approach is based on a simple deep convolutional network called PCANet and does not require large amounts of data due to using principal component analysis to learn the weights. After learning multistage filter banks, a simple binary hashing is applied to the filtered data, and features are pooled from block histograms. A multiclass linear support vector machine is then trained and the system is evaluated using the plant task datasets of LifeCLEF 2014 and 2015. As announced by the organizers, our submission achieved an overall inverse rank score of 0.153 in the image-based and an inverse rank score of 0.162 in the observation-based task of LifeCLEF 2015, as well as an inverse rank score of 0.51 for the LeafScan dataset of LifeCLEF 2014.

Keywords: plant identification, deep learning, PCANet, support vector machine, inverse rank score

1 Overview

In recent years, research in the area of automatic plant identification from photographs has concentrated around annual plant identification competitions that are organized within the CLEF campaigns including ImageCLEF [1–3] and LifeCLEF [4]. CLEF is devoted to promoting and evaluating multilingual and multimodal information retrieval systems and the main goal of these competitions is to benchmark the challenging task of content-based identification and retrieval of plant species which is of immense importance in botany, agriculture, plant taxonomy, pharmacy, and pharmacology. The task is carried out using images of different types of plant parts, such as leaves, branches, stems, flowers, and fruits.

Since 2011, competitive submissions for the plant identification task have been made to ImageCLEF and LifeCLEF in which the participating systems have utilized widely different approaches; still, the problem is far from being solved due to several challenges including large color, illumination, background, size, and shape variations. Deep learning approaches are new and suitable for solving such problems with large amounts of intra-class variability [5].

There are two tasks within the LifeCLEF 2015 campaign: image-based and observation-based plant identification tasks. The image-based task requires identification given a single image, while the goal of the observation-based task is to identify plants based on multi-image query. The latter corresponds to the scenario in which a photographer uses the same camera to take snapshots from different views of various organs of a plant species under the similar lighting conditions and on the same day. The campaign started with the image-based task and the main task became the observation-based task later on. In 2015, the number of species has reached to about 1000, covering the entire flora of a given region.

In this work, we have utilized a system different from our previous submissions [6–9] to recognize plant species using a new deep convolutional network known as PCANet [5]. Although the PCANet method is suboptimal in comparison with common convolutional neural networks (CNN) [10,11], our experiments using PCANet resulted in good performances for aligned images such as scanned leaves.

2 PCANet

PCANet is a recently proposed convolutional network architecture that combines the strengths of principal component analysis (PCA) and deep learning [5]. In comparison with the CNN which attempts to find optimal filters for feature mapping, PCANet is suboptimal in that it learns the filter banks by applying PCA on the input data. On the other hand, its advantages lie in the facts that it does not require large amounts of data or long learning time while still using the core concept of the deep convolutional network architecture.

The general structure of PCANet and our proposed architecture for plant identification are presented in this section.

2.1 General PCANet Architecture

For the first layer in PCANet, overlapping patches of images are applied the Principal Component Analysis. The selected principal components form the first layer filters and the projections of the patches on to the principal components form the response of units in the first layer.

We then repeat this methodology to form a cascaded linear map in the next layers of the deep convolutional network architecture. Next, the method uses binary quantization and hashing for the multi-stage filtered image sets to concatenate them in the decimal form. Finally, local histograms are extracted from the blocks of the quantized images and spatial pyramid pooling method is applied to these histograms to extract features.

The algorithm is explained in detail as follows. The training data contains $i = 1, 2, \dots, N$ images I_i of size $m \times n$. In the first stage, patches of size $k_1 \times k_2$ pixels are extracted around each pixel in the image I_i . Afterwards, all such

overlapping patches are collected, vectorized, and mean subtracted to obtain X_i . Repeating this operation for all images, we obtain a patch collection X , as:

$$X = [X_1, X_2, \dots, X_N] \in R^{k_1 k_2 \times Nmn} \quad (1)$$

Next, in order to calculate the desired filter banks of orthonormal filters, V , PCA minimizes the reconstruction error to compute their L_1 principal components. The constrained optimization is formulated as:

$$\min_{V \in R^{k_1 k_2 \times L_1}} \|X - VV^T X\|_F^2 \quad \text{subject to} \quad V^T V = I_{L_1} \quad (2)$$

where $\|\cdot\|_F$ is Frobenius norm, I_{L_1} is the identity matrix of size $L_1 \times L_1$ and the solution simply consists of finding L_1 principal eigenvectors of XX^T . Therefore, the PCA filters for the first layer form weights $W_{l_1}^1$ for $l_1 = 1, 2, \dots, L_1$, by converting the eigenvectors to matrices of size $k_1 \times k_2$. Hence, the l_1^{th} filtered image is calculated by convolving the l_1^{th} filter with the i^{th} patch-mean removed image, \bar{I}_i , as,

$$I_i^{l_1} = \bar{I}_i * W_{l_1}^1 \quad (3)$$

We can repeat the same approach to learn L_2 PCA filters for the second layer to create double filtered images. For this purpose, all the overlapping patches of each filtered image $I_i^{l_1}$ are collected, vectorized, and mean subtracted to obtain $Y_i^{l_1}$. Repeating this algorithm for all filtered images, we obtain,

$$Y = [Y_1^1, \dots, Y_N^1, Y_1^2, \dots, Y_N^2, \dots, Y_1^{L_1}, \dots, Y_N^{L_1}] \in R^{k_1 k_2 \times L_1 Nmn} \quad (4)$$

Similarly, PCA filters for the second layer, $W_{l_2}^2$ for $l_2 = 1, 2, \dots, L_2$, are obtained by finding L_2 principal eigenvectors of YY^T and rearranging them as matrices of size $k_1 \times k_2$. Therefore, the double filtered image, computed sequentially using the l_1^{th} and l_2^{th} filters, is obtained by convolving the l_2^{th} filter with the i^{th} patch-mean removed filtered image, $\bar{I}_i^{l_1}$, as,

$$O_i^{l_1, l_2} = \bar{I}_i^{l_1} * W_{l_2}^2 \quad (5)$$

As can be seen, in the output O for each image, we have $L_1 \times L_2$ double filtered images with real values. To decrease the number of images, it is proposed to first binarize them using Heaviside step function $H(\cdot)$. Next, for each pixel, we map L_2 quantized binary bits to a decimal number as

$$T_i^{l_1} = \sum_{l_2=1}^{L_2} 2^{l_2-1} H(\bar{I}_i^{l_1} * W_{l_2}^2) \quad (6)$$

In fact, this conversion maps each L_2 binary bits acquired from corresponding pixels of the double filtered binary images into a single graylevel image pixel in the range $[0, 2^{L_2} - 1]$.

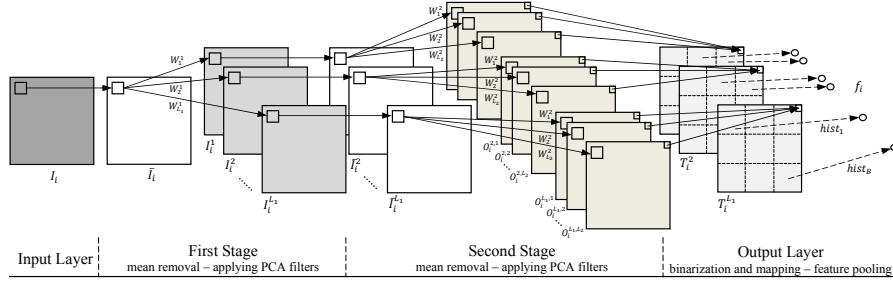


Fig. 1: Block diagram of a two-stage PCANet

Finally, we partition each of L_1 decimal images ($T_i^{L_1}$) into B blocks and compute block histograms (with 2^{L_2} bins) for all L_1 images as the features of i^{th} image,

$$f_i = [hist_1(T_i^1), \dots, hist_B(T_i^1), \dots, hist_1(T_i^{L_1}), \dots, hist_B(T_i^{L_1})] \in R^{1 \times 2^{L_2} L_1 B} \quad (7)$$

where $hist_j(\cdot)$ indicates the histogram of the j^{th} block of the partitioned image. Utilizing local histograms provides translation invariance in the extracted features. Figure 1 displays the block diagram of a two-stage PCANet.

2.2 PCANet Architecture for Plant Identification

In order to process each plant image, color images are converted from RGB to HSY color space [12] and scaled identically to 128×128 pixels. We apply PCANet on each color component of the scaled plant images assuming a 2-stage convolutional network with $L_1 = 10$ and $L_2 = 8$ as the number of filter banks in each stage with the overlapping image patches of size 7×7 and histogram block size of 20×10 . Because of the massive size of data obtained after feature pooling, we chose the multi-class linear Support Vector Machine (SVM) classifier for complexity and accuracy issues in the final stage. For the SVM implementation, we used the Liblinear toolbox [13] and applied the dual L2-regularized-L2-loss model and a misclassification penalty cost of one.

2.3 Score Fusion in the Observation-Based Task

For the observation-based task, we applied the Borda count fusion method [14] to the proposed system outputs to combine the scores obtained by different photographs of individual plants. In this method, each class that appears in the top classes list returned by the classifier receives a vote that is inversely proportional to its rank in that list. Note that for each observation with k images, there are k such class lists. We modified the Borda count in this problem such that votes are distributed not only to the class, but also to the members in the same genus.

3 Performance Analysis


In this section, we will explain the competition datasets and adjust optimal parameters of our proposed system by extracting validation sets from the training data. We then define the performance metrics and report the experimental results on the validation and official test sets.

3.1 Dataset

The plant identification task in LifeCLEF 2015 involves identifying 1,000 species of trees, herbs, and ferns from photographs of their different organs mostly taken within France by different users. The collected dataset contains 113,205 pictures, 91,759 images for training and 21,446 images for testing. Table 1 shows the details of the provided datasets and their sample images.

Table 1: Details of the datasets for plant identification task within the LifeCLEF 2015

	Branch	Entire	Flower	Fruit	Leaf	LeafScan	Stem
# Training Samples	8,130	16,235	28,225	7,720	13,367	12,605	5,476
# Test Samples	2,088	6,113	8,327	1,423	2,690	221	584
# Classes	891	993	997	755	899	351	649



To validate our results, we used the proportionate stratified random sampling, i.e. we first randomly split the training dataset into two subsets after placing images of each plant species into both the training and the validation sets. The proportion of validation sets to training sets is assumed to be around 1 to 3. In other words, for any dataset, we randomly selected one-fourth of available samples of each individual class, if possible, as samples of the validation set.

3.2 Results

We next applied our proposed plant identification system for different plant categories on the obtained training and validation sets. Table 2 shows the system performance in terms of the obtained first rank classification accuracies. As expected, the flower, fruit, and stem photographs are relatively easier to classify compared to branch, leaf, and entire categories.

LifeCLEF lab itself employs a user-based metric called the average inverse rank score [15] instead of the total classification accuracy. The average inverse

Table 2: Top rank classification accuracies for the proposed plant identification system

Category	# Training Samples	# Validation Samples	Accuracy
Branch	6,447	1,683	23.53 %
Entire	12,567	3,668	25.08 %
Flower	21,531	6,694	34.02 %
Fruit	6,072	1,648	40.41 %
Leaf	10,367	3,000	33.80 %
LeafScan	9,576	3,029	90.49 %
Stem	4,344	1,132	37.19 %

Table 3: Official average inverse rank scores of our best run in the imaged-based task of LifeCLEF 2015

Branch	Entire	Flower	Fruit	Leaf	LeafScan	Stem	Overall
0.053	0.106	0.189	0.143	0.111	0.216	0.120	0.153

rank score S is defined as

$$S = \frac{1}{U} \sum_{u=1}^U \frac{1}{P_u} \sum_{p=1}^{P_u} \frac{1}{N_{u,p}} \sum_{n=1}^{N_{u,p}} s_{u,p,n} \quad (8)$$

where U is the number of users who have taken the query pictures; P_u is the number of individual plants observed by the u^{th} user; $N_{u,p}$ is the number of pictures taken from the p^{th} plant observed by the u^{th} user; and $s_{u,p,n}$ is the inverse of the rank of the correct species for the given image, ranging from 0 to 1. Considering this metric, we applied PCANet to the test sets using the learned parameters in the training step and submitted our prediction results to the organizers of LifeCLEF 2015 for official evaluation. Table 3 displays the inverse rank scores of our best run in the image-based LifeCLEF 2015 for different categories of plant identification task. In the observation-based task, our approach using Borda count achieved an inverse rank score of 0.162.

Bearing in mind that this large dataset consists of 1,000 classes of similar categories, our official overall rank score of 0.153 in the image-based task shows a fair performance for our submission. Comparing the official test results given in Table 3 with the higher accuracies reported in Table 2, we conclude that a considerable amount of overfitting existed during validation. This situation could have been improved if we had used more data, but the time complexity was an issue even with the simple architecture.

Furthermore, since a very small subset of the test data in LifeCLEF 2015 belonged to scanned leaves, we skipped the preprocessing and segmentation phase [16] which had been applied to the LeafScan category in our previous submissions [6–9]. Once we applied PCANet to the segmented and preprocessed LeafScan images of LifeCLEF 2014, we achieved the inverse rank score of 0.51 as measured by the campaign organizers.

3.3 Time Complexity

We measured the complexity of our system in terms of the running time for feature extraction and training the classifier. On average over all categories, PCANet took 1.37 seconds/image and 6.13 seconds/image for feature extraction and training, respectively. All codes were implemented in MATLAB (run in 80 GB RAM and 2.50 GHz CPU with two processors).

Although the campaign within the LifeCLEF 2015 had allowed using external training data, we restricted ourselves to the provided LifeCLEF datasets. That was due to the reasons that PCANet does not require large data to learn the weights and that using external datasets would increase the processing time for our system.

3.4 Effects of Parameter Selection

Parameters of the proposed PCANet-based plant identification system were selected experimentally through validation. Due to combinatorial increase, each time we adjusted only one of the parameters until finding the optimal point. For instance, we observed that by increasing or decreasing the image patch size from 7×7 , the performance rapidly decreased. The same conditions were held for the block size of histograms (20×10).

On the other hand, we observed that by increasing the normalization size of input images and/or the number of filter banks (especially within the first stage), the performance improved. However, there was a trade-off between complexity and accuracy, i.e. by increasing the input image size and/or the number of filters, the accuracy improved slightly while the size of the output feature vectors expanded as well. Consequently, the training time increased drastically. Therefore, we set the system parameters equal to values mentioned in the Section 2.2 to evaluate our system.

4 Summary and Discussions

In this work, we used a simple deep convolutional approach called PCANet in order to identify the plant species within the LifeCLEF 2015 plant identification dataset. Our best run has shown a fair performance with overall inverse rank scores of 0.153 and 0.162 in the image-based and observation-based tasks, respectively. It seems that the proposed system is fast and efficient for aligned images such as preprocessed scanned leaves.

Acknowledgments. This project is supported by Turkish Scientific and Research Council of Turkey (TUBITAK) under project number 113E499.

References

1. Goëau, H., Bonnet, P., Joly, A., Boujemaa, N., Barthelemy, D., Molino, J.F., Birnbaum, P., Mouysset, E., Picard, M.: The CLEF 2011 plant images classification task. In: CLEF (Notebook Papers/Labs/Workshop), Amsterdam (2011)
2. Goëau, H., Bonnet, P., Joly, A., Yahiaoui, I., Barthelemy, D., Boujemaa, N., Molino, J.F.: The ImageCLEF 2012 plant identification task. In: CLEF (Online Working Notes/Labs/Workshop), Rome (2012)
3. Goëau, H., Bonnet, P., Joly, A., Bakic, V., Barthelemy, D., Boujemaa, N., Molino, J.F.: The ImageCLEF 2013 plant identification task. In: CLEF (Working Notes), Valencia (2013)
4. Goëau, H., Joly, A., Bonnet, P., Selmi, S., Molino, J.F., Barthelemy, D., Boujemaa, N.: LifeCLEF plant identification task 2014. In: CLEF (Working Notes), Sheffield (2014) 598–615
5. Chan, T.H., Jia, K., Gao, S., Lu, J., Zeng, Z., Ma, Y.: PCANet: A simple deep learning baseline for image classification? Computing Research Repository (CoRR - arXiv) (2014) arXiv:1404.3606v2.
6. Yanikoglu, B., Aptoula, E., Tirkaz, C.: Sabanci-Okan system at ImageClef 2011: Plant identification task. In: CLEF (Notebook Papers/Labs/Workshop). (2011)
7. Yanikoglu, B., Aptoula, E., Tirkaz, C.: Sabanci-Okan system at ImageClef 2012: Combining features and classifiers for plant identification. In: CLEF (Online Working Notes/Labs/Workshop). (2012)
8. Yanikoglu, B., Aptoula, E., Yildiran, S.T.: Sabanci-Okan system at ImageClef 2013 plant identification competition. In: CLEF (Working Notes). (2013)
9. Yanikoglu, B., Yildiran, S.T., Tirkaz, C., Aptoula, E.: Sabanci-Okan system at LifeCLEF 2014 plant identification competition. In: CLEF (Working Notes). (2014) 771–777
10. LeCun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. *Proceedings of the IEEE* **86**(11) (1998) 2278–2324
11. Krizhevsky, A., Sutskever, I., Hinton, G.E.: ImageNet classification with deep convolutional neural networks. In: *Neural Information Processing Systems*. (2012) 1106–1114
12. Hanbury, A.: A 3D-polar coordinate colour representation well adapted to image analysis. In: *Proceedings of the 13th Scandinavian conference on Image analysis*. (2003) 804–811
13. Fan, R.E., Chang, K.W., Hsieh, C.J., Wang, X.R., Lin, C.J.: LIBLINEAR: A library for large linear classification. *Journal of Machine Learning Research* **9** (2008) 1871–1874
14. Mladenov, V., Koprinkova-Hristova, P., Palm, G., Villa, A.E.P., Appollini, B., Kasabov, N.: Artificial neural networks and machine learning. In: *Proceedings of the 23rd International Conference on Artificial Neural Networks*. (2013) 8–9
15. Müller, H., Clough, P., Deselaters, T., Caputo, B.: ImageCLEF: Experimental evaluation in visual information retrieval. Volume 32 of *The Information Retrieval Series*. Springer (2010)
16. Yanikoglu, B., Aptoula, E., Tirkaz, C.: Automatic plant identification from photographs. *Machine Vision and Applications* **25**(6) (2014) 1369–1383