sciendo

BULGARIAN ACADEMY OF SCIENCES

CYBERNETICS AND INFORMATION TECHNOLOGIES • Volume **19**, No 1
Sofia • 2019 Print ISSN: 1311-9702; Online ISSN: 1314-4081
DOI: 10.2478/cait-2019-0005

Deep Learning for Plant Classification and Content-Based Image Retrieval

Bálint Pál Gyires-Tóth, Márton Osváth, Dávid Papp, Gábor Szűcs

Budapest University of Technology and Economics, Department of Telecommunications and Media Informatics, 2nd Magyar Tudósok krt., H-1117, Budapest, Hungary
E-mails: toth.b@tmit.bme.hu osvathmarton@gmail.com pappd@tmit.bme.hu szucs@tmit.bme.hu

***Abstract*:** *The main goal of the present research is to classify images of plants to species with deep learning. We used convolutional neural network architectures for feature learning and fully connected layers with logsoftmax output for classification. Pretrained models on ImageNet were used, and transfer learning was applied. In the current research image sets published in the scope of the PlantCLEF 2015 challenge were used. The proposed system surpasses the results of all top competitors of the challenge by 8% and 7% at observation and image levels, respectively. Our secondary goal was to satisfy the users' needs in content-based image retrieval to give relevant hits during species search task. We optimized the length of the returned lists in order to maximize MAP (Mean Average Precision), which is critical to the performance of image retrieval. Thus, we achieved more than 50% improvement of MAP in the test set compared to the baseline.*

***Keywords*:** *deep learning, convolutional neural networks, Inception V3, MAP, image retrieval.*

1. Introduction

Being able to identify the different species of plants growing in agricultural areas and to automatically detect the presence of invasive species is crucial. Identifying plants is usually a difficult task, sometimes for professionals (such as farmers or lumberjacks) as well [1]. Using content-based image retrieval technologies is a promising possibility in this field (as a fine-grained object categorization problem [18]), and the aim of our work was to solve it automatically.

In this image-based plant identification work, we focused on tree, herb and fern species identification based on different types of images. We used PlantCLEF 2015 database [8], where the number of species was 1000, and the images showed either parts – branch, leaf, leafscan (scan or scan-like pictures of leaf), flower, fruit, stem – or the entire plant. The dataset was composed of 113,205 pictures belonging to 41,794 observations of 1000 species of plants living in Western European regions; and this was collected by 8,960 distinct users, the distribution of the different types of images by the part of the plant they show can be seen in Table 1.

Table 1. Distribution of images among different viewpoints

Method	Total	Branch	Entire	Flower	Fruit	Leaf	L. Scan	Stem
Trayning	91,759	8,130	16,235	28,225	7,720	13,367	5,476	12,605
Test	21,446	2,088	2,983	6,113	8,327	1,423	696	935
All	113,205	10,218	19,218	34,438	16,047	14,790	6,172	13,540

Our aim was twofold: (1) firstly, to classify the observed plants (so-called observations) into the known categories (species) with high accuracy (see extended classification below), (2) secondly, to construct a content-based image retrieval system. Besides the images of an observation, some contextual metadata (data, location, author and rating information) were available, but we had not used them (we focused on image contents only). Our aim was to elaborate fully automatic methods for these problems.

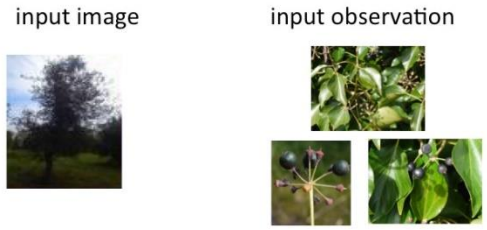
Given a set of N training examples of the form $\{(x_1, y_1), \ldots, (x_N, y_N)\}$ such that x_i is the feature vector of the i-th example, y_i is the corresponding label (class), and the number of the classes is denoted by C. Based on the learnt model the traditional classifier predicts a class for unknown example: $\hat{y}$, and in order to measure how well a function fits the training data, an error function $L : Y \times Y \to R^{\geq 0}$ is defined. The error of predicting the value of each example $L(y_i, \hat{y}_i)$ can be summarized for overall error.

In extended classification task not only one class is predicted, but series of them (vector), $\hat{y}_{i,1}, \hat{y}_{i,2}, \ldots, \hat{y}_{i,C}$, where these labels are in decreasing order according to their reliabilities (the first element is a most probable class for i-th example). So the error function $L(y_i, \hat{\hat{y}}_i)$ is based on the prediction vector and the real class.

An ‘extended classification’ task can occur at half automatic annotation, where a new instant should be categorized into large number classes; in this case a human annotator can see not only the most probable class but the second, third, etc. most probable classes as well (human annotator will accept the true class from the top of the predicted classes).

At retrieval task, the aim is to select examples from the available set for a class (as a query) and to rank them in decreasing order according to their predicted relevance (an example is relevant to the query if its class is the same as the class in the query). Let us denote the set of examples possessing class c by S_c, and the identities of k retrieved examples by $\vec{r}_k$ (as vector) in decreasing order, thus error function, $L(S_c, \vec{r}_k)$ can be defined to evaluate the retrieval. Finding a good k value is a subtask of retrieval.

At our first task (i.e., extended classification) there is an input image as can be seen at the left side of Fig. 1 (or observation with more images at the right side of the Fig. 1), and the aim is to predict series of species, as prediction vector. In every observation, the plants belong to the same species. The true species of the left image and all right images (observation) are Quercus ilex and Hedera helix, respectively; and the task was to find out the species (as single label classification).



output vector

1	Betula Pendula	0.623
2	Hedera helix	0.601
3	Quercus ilex	0.521
...		
1000	Papaver rhoeas	0.021

output vector

1	Hedera helix	0.713
2	Quercus ilex	0.591
3	Betula Pendula	0.551
...		
1000	Cercis silliq.	0.009

Fig. 1. Example inputs and outputs for extended classification

Our second task is the content-based image retrieval, where only visual information is available; the input is the name of species, and the outputs are predicted images.

On the contrary to a classical "query by image" system, the target here is "query by category": the user gives the name of a plant species and the system retrieve a ranked set of images corresponding to this species. The retrieval task, as an offline problem can only be executed on a large image set, so after the classification of all plant images. The goodness of the solution of offline problem can be measured by users' satisfaction; i.e., users, who search an interesting plant species, would like to get lots of images of this species, and none of the others. The "precision", as the relative ratio of the relevant images in the retrieved list, is a good indicator for measure the goodness. Even if the precisions are the same, the goodness can be different, e.g., the relevant images in the retrieved list being uniformly distributed is worse than all the relevant images being at the beginning of retrieved list. That is why we used a better indicator, Average Precision (AP), which is the average of precision values at each position in the retrieved list where the image is relevant. The difficulty is to determine the length of the retrieved list in order to meet users' needs. In the retrieval task, we focused on this length optimization.

2. Previous works

There have been a great number of previous works about plant classification based on image data [5] with methods Probabilistic Neural Networks (PNN) [31, 12] and Support Vector Machine (SVM) [14]. Plant identification task existed in the ImageCLEF challenge from 2011, and in 2014 a combined system of convolutional neural nets and SVM won the challenge [4]. The CNNs had five convolutional layers, however, their pure deep learning solution was outperformed by ensemble systems. Other approach used hand-crafted visual features for the different view types and trained SVM classifier [16]. SVM methods present good results in image categorization particularly when they are associated with a kernel function. In the challenge in 2014 we also used SVM with viewpoints combined method for image-

based plant identification [29]. In 2014 another group used a pretrained Overfeat [20] network for feature learning, and the output of the fully-connected layer (before the softmax layer) was fed into a tree-based ensemble classifier [25]. However, other groups with SVM based solutions resulted better. In 2015 an Inception Convolutional Neural Network (CNN) model-based network won the competition [26]. They have pretrained the model with ImageNet and fine-tuned with the PlantCLEF database. They used the combined output of five CNNs, that were fine-tuned with randomly selected parts of the database, however, hyperparameter optimization was not performed. Also in 2015 a pretrained AlexNet was fine-tuned, which resulted the 4th place [21].

Our goal was to introduce AlexNet augmented with new results of deep learning and Inception V3 architectures, apply hyperparameter optimization and investigate if further improvements could be achieved.

3. Content-based plant classification with deep learning

3.1. Convolutional neural network

Nowadays state-of-the-art image recognition and classification solutions generally use deep learning methodology. Deep convolutional neural networks are able to learn the descriptive features of the image database in many abstraction levels. In a baseline system the convolutional layers are followed by a classifier. Using feed-forward layers as classifier the whole end-to-end process of feature learning and classification is controlled by deep learning methods, thus higher accuracy can be achieved.

The LeNet was among the first neural networks that were directly fed with matrix representation of images instead of feature vectors [15]. This type of network is referred to as CNN. The LeNet was followed by many variants until the real breakthrough of convolutional neural networks was achieved in 2012, when a team led by Geoffrey Hinton and Alex Krizhevsky won the ImageNet Large Scale Visual Recognition Competition [23] by a large margin [13]. The main strength of their solution lay in the application of numerous fine-tuned convolutional and max-pooling layers, in the application of dropout method [24], in the usage of non-saturating neurons (Rectified Linear Units, ReLUs) [30], and in an efficient GPU implementation. This model is often referred to as AlexNet. AlexNet is followed by many improved convolutional neural network variants. These variants include, e.g., the Inception models [2, 27, 28] and the deep residual neural network [9]. Our proposed method is based on a simple data preparation and we used a modified version of AlexNet and the Inception *V*3 model as described below.

3.2. Data preparation

We kept the data preparation methods as low as possible. As state-of-the-art applications of CNN suggest – due to the deep convolutional network's feature learning behavior – we applied cropping, scaling and normalization. Hence, we only cropped the center of the images with the shorter dimension of the image and scaled it down to the network's input dimension, which is 299×299 pixels. Finally, we

normalized the red, green and blue color channels individually to zero mean and unit variance.

We could have improved the performance by generating "more" training data with simple transformations on the provided pictures, like random rotation, mirroring and cropping. Moreover, we could also have mixed in some random noise to make the recognition more robust.

3.3. The proposed convolutional network

For training the PlantCLEF 2015 database, first we used a modified version of AlexNet [13]. We changed the ReLU (Rectified Linear Unit) activation functions to parametric ReLUs (PReLUs) [10] in order to avoid zero gradients. In PReLU the activation function is defined as

$$f(y_i) = \begin{cases} y_i & \text{if } y_i > 0, \\ a_i y_i & \text{if } y_i \leq 0. \end{cases} \tag{1}$$

The PReLU is the generalization of ReLU and Leaky ReLU, because if we use parameter a_i=0, then we will get ReLU; and if the parameter a_i is equal to 0.01 (or other small fixed number), then this is so-called Leaky ReLU, which surpasses the accuracy of ReLU [19]. In case of PReLU a_i is a learnable parameter and adjusted to the training data, thus higher classification accuracy was achieved on the ImageNet 2012 (which is a post-competition result).

The other modification we applied was the introduction of batch normalization [11] before the max-pooling layers of AlexNet. It has been known for a long time [17] that standardizing the inputs to zero mean and unit variance the neural net converges faster. However, the distribution of each layer's inputs changes due to the change in network parameters during training, which can radically slow down convergence. Batch normalization addresses this problem by standardizing the inputs for each layer. Experiments in [11] show that with batch normalization significantly better accuracy can be achieved on MNIST and ImageNet datasets with faster convergence. The block diagram of the proposed convolutional network is shown in Fig. 2. However, AlexNet had a poor performance on the PlantCLEF 2015 dataset (the MAP value was below 0.1).

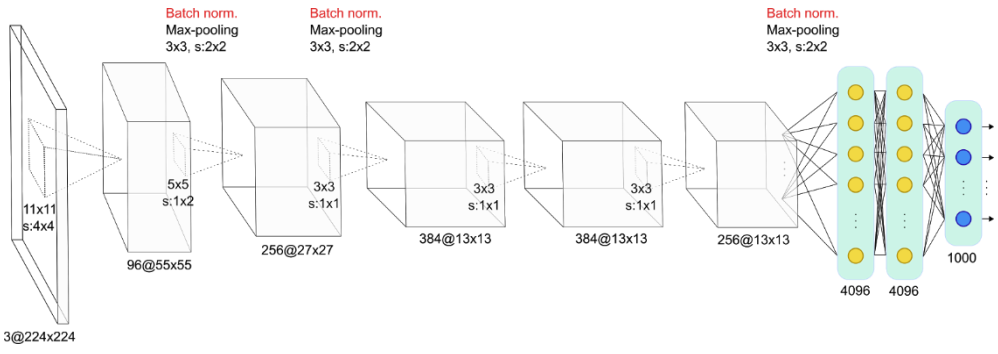


Fig. 2. The block-diagram of the proposed convolutional network, which is a modification of AlexNet [15]. A@B×B refers to A number of planes with size B×B. The C×C, s: D×E refers to C×C kernel size with D×E stride

Therefore, a more advanced model, namely the Inception V3 [1] was introduced. Inception V3 uses ReLU as activation function; it applies batch normalization and additional factorization ideas (e.g., 1×1 convolutions for dimensionality reduction). The Inception V3 model has great performance on ImageNet database, and we expect that it will also be suitable for plant recognition purposes.

For optimizer, first we chose AdaDelta [32], which is a great tool for adaptively adjusting the learning rate. It originated from AdaGrad [6], which applies a continual decay of learning rates throughout training, while AdaDelta uses a restricted accumulation window instead of accumulating the sum of squared gradients over all time. This prevents accumulation to infinity, thus it ensures that the learning can continue even after many iterations. However, according to our preliminary tests, the error did not converge when we used AdaDelta in the Inception V3 model. Thus, we ended up using Stochastic Gradient Descent (SGD) on mini-batches [17] in case of Inception V3. AlexNet exploited the advantage of AdaDelta, however, the classification accuracy was much lower.

Our research originated from an Inception V3 model that was pretrained on Imagenet. We applied transfer learning by resetting the last fully connected classification layer's weights to ensure that the network learns the appropriate classes. Furthermore, we added a narrowing layer to restrict the 1008 outputs to 1000. We did not freeze any layers; all layers were trained equally. Finally, we removed the last SoftMax layer and appended a LogSoftMax layer to ensure that the sum of the output is 1. As the last layer's output contains log-probabilities, we used Negative Log Likelihood as loss function.

Hyperparameter optimization was performed with grid and random search methods in terms of learning rate and batch size. Throughout the experiments the following hyperparameters achieved the best accuracy with simple Stochastic Gradient Descent with Learning rate: 0.045; Weight decay 1e-5; Momentum 0.1; Learning rate decay 1e-7; Batch size 10.

3.4. Usage of the proposed method for classification and retrieval

The Inception V3 convolutional neural network gives a prediction for each unknown image which contains 1000 values, one possibility for each class. Generally, in case of traditional classification tasks a category from the output distribution is chosen (e.g., class label with the highest possibility), however, in test phase this approach gives only a binary result (e.g., the predicted class was correct or not). Therefore, we calculated the more sophisticated *S* metric (see Section 4.1 for details) to evaluate the results of the extended classification. For the image level evaluation of S we used the possibility values provided by the CNN.

As the training was executed in image level, an observation level classifier was not available, so we constructed one using the image level predictions. In case of many observations only one viewpoint was provided, and in other cases, the number of available viewpoints was 2-3 only. For observation level classification we calculated the average of predictions in every available viewpoint and took a weighted average of them as final prediction. We used logistic regression to optimize the weights of different viewpoints. This optimization was performed on a validation

set generated from the training data of the PlantCLEF 2015 competition by randomly selecting 9.5% of images from it. Additionally, we calculated the simple average of the different viewpoints, i.e. with unified weights. Furthermore, it is important to note that training several CNNs, one for each viewpoint was less efficient than training one CNN with all images [26, 3]. Therefore, we chose to train a single network for the total dataset.

At the retrieval task, we used the same possibility values as for the calculation of image level S metric, and we sorted them in descending order for each category. We call these descending order of possibility "sorted list" or "retrieved list". The difficulty was to determine the length of the retrieved list in order to meet users' need. Our solution to this problem was a length optimization algorithm, described in Section 4.2.

3.5. Training times

The hardware we used for training were an NVidia GTX 970 (4 GB) and an NVidia Titan X (12 GB) GPUs. For data preparation, training and evaluating deep neural networks the Torch7 deep learning framework was used.

Preprocessing the images and converting them to the suitable format took about half an hour including the calculation of the pictures' mean and standard deviation per channel. As for the Inception V3 execution times, 1 training epoch took approximately 52 min, and the validation took about 13 minutes on the Titan. After 53 epochs (cca. 2 days) the training reached the best result thus early stopping was triggered after 103 epochs (cca. 4 days). However, evaluation of an image only took about a second on the trained network.

Regarding the limitations of the model, hyperparameter optimization is very resource demanding due to the quite complex architecture of Inception V3. Furthermore, the wide variety of crowd-sourced images would require a massive preprocessing step that includes a huge amount of manual work.

4. Evaluation

4.1. Results of classification

At extended classification task, the evaluation process takes into consideration not only one class but series of them (vector), $\hat{y}_{p,1}, \hat{y}_{p,2}, ..., \hat{y}_{p,C}$, where these labels are in decreasing order according to their reliabilities which were the response of the classifier (the first element is most probable class for *p*-th observation). For the evaluation we used different metrics from PlantCLEF [6]: The first metrics was a special score based on reciprocal rank, where the rank is the sequential number of the correct species in the list of retrieved species (sorted by decreasing order). The dataset was built in a collaborative manner, thus, simple average score may introduce some bias. Instead of "simple mean" the mean of the average score rate per author was defined for *S* at observation level as can be seen in the following equation:

$$S = \frac{1}{U} \sum_{u=1}^{U} \frac{1}{P_u} \sum_{p=1}^{P_u} S_{u,p}, \tag{2}$$

where U is the number of users (who have at least one image in the test data); P_u – the number of individual plants observed by the u-th user.

The error function $L(y_p, \hat{y}_p)$ is based on the prediction vector and the real class:

$$S_{u,p} = \frac{1}{j}\Big|\hat{y}_{p,j} = y_p\,, \tag{3}$$

where score $S_{u,p}$ is the p-th observation of the u-th user, so this equals to the inverse of the rank of the correct species.

The second method was used to evaluate the prediction of classification task at image level. Following the same motivations explained above, a simple mean on all test images would, however, introduce some bias as well (some authors sometimes provided many pictures of the same individual plant to enrich training data with less efforts). Because the final aim was to evaluate the ability of a system to provide the correct answer based on a single plant observation, we also had to average the classification rate on each individual plant. Finally, our secondary metric at image level is defined as the following average classification score S:

$$S = \frac{1}{U}\sum_{u=1}^{U}\frac{1}{P_u}\sum_{p=1}^{P_u}\frac{1}{N_{u,p}}\sum_{n=1}^{N_{u,p}} S_{u,p,n}, \tag{4}$$

where U is the number of users (who have at least one image in the test data); P_u – the number of individual plants observed by the u-th user; $N_{u,p}$ – the number of pictures taken from the p-th plant observed by the u-th user.

Similarly, $S_{u,p,n}$ is the n-th picture taken from the p-th plant observed by the u-th user:

$$S_{u,p,n} = \frac{1}{j}\Big|\hat{y}_{n,j} = y_n\,. \tag{5}$$

The final score S of extended classification in both of observation and image level will be between 0 and 1, and the goal is to attain larger S.

We tested our solution on the official test set of PlantCLEF 2015 competition (see Table 1) released by the competition organizers after the finish of the contest (the structure of the data is described in Section 1). We used the following weights – coming from the Logistic Regression (LogReg) optimization – for the observation level classification: 0.108, 0.123, 0.242, 0.051, 0.142, 0.080 and 0.222 for Flower, Fruit, Leaf, LeafScan, Entire, Stem and Branch, respectively. However, the S metric (observation level) with this configuration was slightly lower than with unified weights, namely 0.7051 compared to 0.7196. The results are shown in Table 2.

Table 2. Final test results of classification with the Inception V3 model (see Section 3.3 for details) on PlantCLEF 2015 dataset (as described in Section 1).

Solution	S (image level)	S (observation level)
Our solution (with unified fusion)	0.6956	0.7196
Our solution (with LogReg fusion)	0.6956	0.7051

4.2. MAP optimization in results of the image retrieval

The scores S of observation and image level are able to measure the goodness of classification, however, we need further metrics for measurement the goodness of retrieval task.

As we mentioned above, our second goal was to construct a content-based image retrieval system. A common issue with this is how to determine the length of the returned lists (the number of images in the list). It is a critical point of retrieval tasks because the users would like to get relevant pictures. Therefore, we focused on the optimization of these lists. We used AP (Average Precision) to measure the goodness of a retrieved list, i.e., the ratio of the relevant and all (non-relevant and relevant) images in that list. We calculated this AP for each category and averaged them to get the MAP (Mean Average Precision) metric [22], as can be seen in the following equation:

$$\text{MAP} = \frac{\sum_{k=1}^{C} \text{AP}_k}{C}, \qquad \text{AP}_k = \sum_{i=1}^{L} \text{precision}(i) \cdot \Delta\text{recall}(i), \tag{6}$$

where C is the number of classes, and L is the length of the retrieved list.

We estimated the optimal length of the returned lists by measuring MAP values for every different size; this optimization was performed on the same validation set that we used for the LogReg optimization.

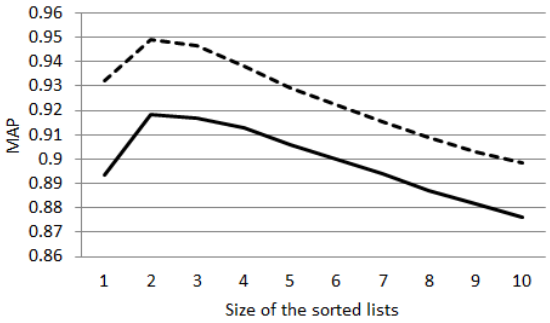


Fig. 3. MAP optimization for the validation and test sets (dashed – validation; solid – test)

We defined the same length for each category, thus every AP value was used uniformly with the same weight at calculating the MAP. As can be seen on the Fig. 3, the results of the validation set got the highest MAP (=0.9490) when the length of the returned lists was 2. After this point, the longer the lists were, the lower the MAP was. With full lists the MAP was 0.7515, which means that the optimization resulted in nearly 26% growth. So, based on this we chose the size of the returned lists to be also 2 for the test set (official test set of PlantCLEF 2015 competition). We evaluated the results of the test set for the 2-long lists, and it reached 0.9185 MAP score. To decide if it was a good estimation, we calculated MAP values for the test set (solid line on Fig. 3), similarly to the validation set. It can be seen that the decision was right because the 2-long lists were also optimal for evaluating MAP on the result of the test set. In this case, the MAP was 0.6086 without cutting the returned lists.

Based on this we can conclude that the preliminary prediction of the size of the returned lists had a positive effect, because it increased the MAP value by more than 50%. In retrieval tasks, it is highly recommended not to return too many non-relevant hits (especially in image retrieval). Supposing that we chose 10-long lists to return, Fig. 3 shows that the proposed system would give back approximately 9 relevant images out of 10.

4.3. Comparative assessment

Image recognition systems use common CNN architectures nowadays, like AlexNet, VGG Net, GoogLeNet and Inception. These networks for feature learning are typically trained on the ImageNet database and fine-tuned with data of the target scenario and generally apply techniques that are proven to be effective empirically. In this research, we concentrated on AlexNet and Inception V3, and with hyperparameter fine-tuning, we could surpass the performance of previous methods in the plant identification task. In the following, the similar approaches are briefly investigated, and Table 3 compares the main features of our and of the previous solutions.

SNUMED INFO [26] used 1 and 5 random CNNs, but did not use CNNs for each viewpoint. Their best run was based on Majority voting method (better than Borda Count). QUT RV [4] and INRIA ZENITH [3] tried combining several CNNs (one for each view), but 1 CNN was better in case of both researches. At the best run of INRIA ZENITH observations composed of several images, are combined using a Maximum function, although they tried Borda Count. (At another run of INRIA they applied SVM with Fisher Vectors.) At the best run of EcoUAN [21] the sum of all predicted image vectors was used for observations, which is equivalent to average function. At the work of MICA [16] Kernel DES (KDES) and SVM were used for classification, and BC (Borda count), IRP (Inverse Rank Position) and WP (Weighted Probability) for observation fusion.

We compared our results with other available results coming from state-of-the-art solutions as can be seen in Table 3. According to the results our model surpasses the S metrics of all other previous solutions at observation and at image level as well. Compared to the best performing previous system we have reached 8% and 7% better result at observation and image levels, respectively.

Table 3. Comparison of test results and approaches of different solutions

Our solution	Goodness of classification: *S* metrics		Fusion method	Ordered list optimization
	S (image level)	*S* (observation level)		
	0.6956	**0.7196**	Unified, Log Regr.	Yes
Best run of SNUMED INFO [26]	0.652	0.667	BC, Majority voting	No
Best run of QUT RV [4]	0.590	0.633	n.a.	Only first 1 & 5 was used
Best run of INRIA ZENITH [3]	0.581	0.609	Maximum function	No
Best run of EcoUAN [21]	0.486	0.487	Average	No
Best run of MICA [16]	0.194	0.209	BC, IRP, WP	No

5. Conclusion

We elaborated a classification method for image-based plant identification task and content-based image retrieval problem. We used pretrained convolutional neural networks with transfer learning. At the preprocessing stage the images were normalized to zero mean and unit variance, they were also cropped and scaled down. AlexNet type neural network had significantly lower performance on the plant database compared to the Inception V3 model.

To evaluate the efficiency of the solution, the *S* and MAP metrics for classification and retrieval tasks were measured, respectively. The image sets published under the PlantCLEF 2015 challenge in the LifeCLEF campaign were used. The results showed that the proposed system surpasses the S metrics of all top competitors of this challenge. On the other hand, we optimized the length of the returned lists to maximize the MAP score. Comparing the baseline (i.e., without MAP optimization) and our solution we gained more than 50% improvement of MAP in the test set.

Our contribution is applying and fine-tuning pretrained AlexNet and Inception V3 CNN models for plant classification purposes, furthermore the optimization of the training and investigation of the returned lists' length in the content-based image retrieval. These results show not only the importance of a well-trained state-of-the-art convolutional neural network but the significant impact of metrics optimization as well.

***Acknowledgments*:** The research presented in this paper has been supported by the European Union, co-financed by the European Social Fund (EFOP-3.6.2-16-2017-00013), by the BME-Artificial Intelligence FIKP grant of Ministry of Human Resources (BME FIKP-MI/SC), by Doctoral Research Scholarship of Ministry of Human Resources (ÚNKP-18-4-BME-394) in the scope of New National Excellence Program, by János Bolyai Research Scholarship of the Hungarian Academy of Sciences. We gratefully acknowledge the support of NVIDIA Corporation with the donation of the Titan Xp GPU used for this research.

References

1. Bonnet, P., A. Joly, H. Goëau, J. Champ, C. Vignau, J. F. Molino, D. Barthélémy, N. Boujemaa. Plant Identification: Man vs. Machine. – Multimedia Tools and Applications, Vol. **75**, 2016, No 3, pp. 1647-1665.
2. Szegedy, C., V. Vanhoucke, S. Ioffe, J. Shlens, Z. Wojna. Rethinking the Inception Architecture for Computer Vision. – arXiv preprint arXiv:1512.00567, 2015.
3. Champ, J., T. Lorieul, M. Servajean, A. Joly. A Comparative Study of Fine-Grained Classification Methods in the Context of the Lifeclef Plant Identification Challenge 2015. – In: Working Notes of CLEF 2015 Conference, 2015.
4. Chen, Q., M. Abedini, R. Garnavi, X. Liang. IBM Research Australia at LifeCLEF2014: Plant Identification Task. – In: Working Notes of CLEF 2014 Conference, 2014., pp. 693-704.
5. Cope, J. S., D. Corney, J. Y. Clark, P. Remagnino, P. Wilkin. Plant Species Identification Using Digital Morphometrics: A Review. – Expert Systems with Applications, Vol. **39**, 2012, No 8, pp. 7562-7573.
6. Duchi, J., E. Hazan, Y. Singer. Adaptive Subgradient Methods for Online Learning and Stochastic Optimization. – The Journal of Machine Learning Research, Vol. **12**, 2011, pp. 2121-2159.

7. G e, Z., C. M c c o o l, P. C o r k e. Content Specific Feature Learning for Fine-Grained Plant Classification. – In: Working Notes of CLEF 2015 Conference, 2015.
8. G ö e a u, H., A. J o l y, B. P i e r r e. LifeCLEF Plant Identification Task 2015. – CLEF Working Notes, 2015.
9. H e, K., X. Z h a n g, S. R e n, J. S u n. Deep Residual Learning for Image Recognition. – In Proc. of IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 770-778.
10. H e, K., X. Z h a n g, S. R e n, J. S u n. Delving Deep into Rectifiers: Surpassing Human-Level Performance on Imagenet Classification. – In: Proc. of IEEE International Conference on Computer Vision, 2015, pp. 1026-1034.
11. I o f f e, S., C. S z e g e d y. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. – arXiv preprint arXiv:1502.03167, 2015.
12. K a d i r, A., L. E. N u g r o h o, A. S u s a n t o, P. I. S a n t o s a. Leaf Classification Using Shape, Color, and Texture Features. – International Journal of Computer Trends and Technology, Vol. **1**, July-August 2011, No 3, pp. 306-311.
13. K r i z h e v s k y, A., I. S u t s k e v e r, G. E. H i n t o n. Imagenet Classification with Deep Convolutional Neural Networks. – In: Advances in Neural Information Processing Systems, 2012, pp. 1097-1105.
14. K u m a r, N., P. N. B e l h u m e u r, A. B i s w a s, D. W. J a c o b s, W. J. K r e s s, I. C. L o p e z, J. V. S o a r e s. Leafsnap: A Computer Vision System for Automatic Plant Species Identification. – In: Computer Vision ECCV'2012. Berlin, Heidelberg, Springer, 2012, pp. 502-516.
15. L e C u n, B. B., J. S. D e n k e r, D. H e n d e r s o n, R. E. H o w a r d, W. H u b b a r d, L. D. J a c k e l. Handwritten Digit Recognition with a Back-Propagation Network. – In: Advances in Neural Information Processing Systems, Vol. **2**, 1990, pp. 396-404.
16. L e, T. L., D. N. D n g, H. V u, T. N N g u y e n. Mica at Lifeclef 2015: Multi-Organ Plant Identification. – In: Working Notes of CLEF 2015 Conference, 2015.
17. L e C u n, Y., L. B o t t o u, G. O r r, K. M u l l e r. Efficient Backprop. – In: G. Orr, K. Muller, Eds. Neural Networks: Tricks of the Trade. Springer, 1998, pp. 9-48.
18. L o n g, X., H. L u, Y. P e n g, X. W a n g, S. F e n g. Image Classification Based on Improved VLAD. – Multimedia Tools and Applications, Vol. **75**, 2016, Issue 10, pp. 5533-5555.
19. M a a s, A. L., Y. H. A w n i, A. Y. N g. Rectifier Nonlinearities Improve Neural Network Acoustic Models. – In: Proc. ICML, Vol. **30**, 2013, p. 1.
20. S e r m a n e t, P., D. E i g e n, X. Z h a n g, M. M a t h i e u, R. F e r g u s, Y. L e C u n. Overfeat: Integrated Recognition, Localization and Detection Using Convolutional Networks. – arXiv preprint arXiv:1312.6229, 2013.
21. R e y e s, A. K., J. C. C a i c e d o, J. E. C a m a r g o. Fine-Tuning Deep Convolutional Networks for Plant Recognition. – In: Working Notes of CLEF 2015 Conference, 2015.
22. R o b e r t s o n, S. A New Interpretation of Average Precision. – In: Proc. of 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, ACM, 2008, pp. 689-690.
23. R u s s a k o v s k y, O., J. D e n g, H. S u, J. K r a u s e, S. S a t h e e s h, S. M a, Z. H u n a g, A. K a r p a t h y, A. K h o s l a, M. B e r n s t e i n, A. C. B e r g, L. F e i-F e i. Imagenet Large Scale Visual Recognition Challenge. – International Journal of Computer Vision, Vol. **115**, 2015, No 3, pp. 211-252.
24. S r i v a s t a v a, N., G. H i n t o n, A. K r i z h e v s k y, I. S u t s k e v e r, R. S a l a k h u t d i n o v. Dropout: A Simple Way to Prevent Neural Networks from Overfitting. – The Journal of Machine Learning Research, Vol. **15**, 2014, No 1, pp. 1929-1958.
25. S ü n d e r h a u f, N., C. M c C o o l, B. U p c r o f t, T. P e r e z. Fine-Grained Plant Classification Using Convolutional Neural Networks for Feature Extraction. – In: Working Notes of CLEF 2014 Conference, 2014., pp. 756-762.
26. S u n g b i n, C. Plant Identification with Deep Convolutional Neural Network: SNUMedinfo at LifeCLEF Plant Identification Task 2015. – In: Working Notes of CLEF 2015 Conference, 2015.
27. S z e g e d y, C., W. L i u, Y. J i a, P. S e r m a n e t, S. R e e d, D. A n g u e l o v, D. E r h a n, V. V a n h o u c k e, A. R a b i n o v i c h. Going Deeper with Convolutions. – In: Proc. of IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 1-9.

28. S z e g e d y, C., S. I o f f e, V. V a n h o u c k e. Inception-v4, Inception-Resnet and the Impact of Residual Connections on Learning. – arXiv Preprint arXiv:1602.07261, 2016.
29. S z ű c s, G., D. P a p p, D. L o v a s. Viewpoints Combined Classification Method in Image-Based Plant Identification Task. – In: L. Cappellato, N. Ferro, M. Halvey, W. Kraaij, Eds. Working Notes for CLEF 2014 Conference, Sheffield, Great Britain, Vol. **1180**, 15-18 September 2014, pp. 763-770.
30. N a i r, V., G. E. H i n t o n. Rectified Linear Units Improve Restricted Boltzmann Machines. – In: Proc. of 27th International Conference on Machine Learning, 2010, pp. 807-814.
31. W u, S. G., F. S. B a o, E. Y. X u, Y. X. W a n g, Y. F. C h a n g, Q. L. X i a n g. A Leaf Recognition Algorithm for Plant Classification Using Probabilistic Neural Network. – In: IEEE International Symposium on Signal Processing and Information Technology, 2007, pp. 11-16.
32. Z e i l e r, M. D. ADADELTA: An Adaptive Learning Rate Method. – arXiv preprint arXiv:1212.5701, 2012.

Received: 25.09.2018; Second Version: 30.11.2018; Accepted: 20.12.2018