Plant Recognition by Inception Networks with Test-time Class Prior Estimation

CMP submission to ExpertLifeCLEF 2018

Milan Šulc[1][0000−0002−6321−0131], Lukáš Picek[2][0000−0002−6041−9722], and Jiří Matas[1][0000−0003−0863−4844]

[1] Center for Machine Perception, Dept. of Cybernetics, Faculty of Electrical Engineering, Czech Technical University in Prague
{sulcmila,matas}@cmp.felk.cvut.cz
[2] Dept. of Cybernetics, Faculty of Applied Sciences, University of West Bohemia, Pilsen, Czech Republic
picekl@kky.zcu.cz

Abstract. The paper describes an automatic system for recognition of 10,000 plant species from one or more images. The system finished 1st in the ExpertLifeCLEF 2018 plant identification challenge with 88.4% accuracy and performed better than 5 of the 9 participating plant identification experts. The system is based on the Inception-ResNet-v2 and Inception-v4 Convolutional Neural Network (CNN) architectures. Performance improvements were achieved by: adjusting the CNN predictions according to the estimated change of the class prior probabilities, replacing network parameters with their running averages, and test-time data augmentation.

Keywords: Plant Recognition, Plant Identification, Computer Vision, Convolutional Neural Networks, Machine Learning, Class Prior Estimation, Fine-grained, Classification

1 Introduction

The ExpertLifeCLEF [3] plant identification challenge is organized in connection with the LifeCLEF 2018 workshop [4] at the Conference and Labs of the Evaluation Forum. The goal of the challenge is assess the quality of automatic, machine-learned recognition systems and to compare their accuracy with human experts in plant sciences. For practical reasons, the experts are evaluated on a small subset of the test data.

The data provided for the challenge cover 10 000 species of plants – herbs, trees and ferns – and consist from:

– PlantCLEF 2017 EOL: 256K images from the Encyclopedia of Life[3] provided in the 2017 challenge [2] as the "trusted" training set.

[3] http://www.eol.org

– PlantCLEF 2017 web: 1.4M images automatically retrieved by web search engines, provided in the 2017 challenge [2] as the "noisy" training set.
– PlantCLEF 2017 test set: 25K test images from the 2017 challenge [2], now available with ground truth label annotations.
– PlantCLEF 2016 subset: 64K images from the PlantCLEF 2016 [1] challenge training- and test sets, covering only 717 of the 10k species. The remaining classes from the 2016 challenge do not exactly taxonomically match the 2017/2018 list of species.
– ExpertLifeCLEF 2018 test set: 6 892 unlabeled images used for evaluation of the submitted methods. Examples from the set are displayed in Figure 1.

Fig. 1. ExpertLifeCLEF 2018 test set - randomly selected samples.

The proposed classification system builds upon the state-of-the-art Convolutional Neural Network (CNN) architectures, described in Section 2.1. Section 2.3 discusses the use of running averages of the trained network parameters instead of values from the last training step which noticeably increased the accuracy of our models.

The class frequencies in the training data follow a long-tailed distribution. It is reasonable to expect that the training data, whose significant majority was downloaded from the web, have different class prior probabilities than the test set. In section 2.4 we consider the problem of different class prior probability distributions and implement an existing method [5,6] to improve the CNN predictions by estimating the test-time priors.

Section 3 describes the 5 submissions we made. Results of the challenge are presented in Section 4. One of the submitted plant recognition methods achieved the best accuracy among automated systems, and thus placed 1st in the challenge and it outperformed 5 of 9 human experts.

2 Methodology

2.1 Convolutional Neural Networks

Table 1. Optimizer hyper-parameters, common to all networks in the experiments:

Parameter	Value
Optimizer	rmsprop
RMSProp momentum	0.9
RMSProp decay	0.9
Initial learning rate	0.01
Learning rate decay type	Exponential
Learning rate decay factor	0.94

The proposed method is based on two architectures – Inception Resnet v2 and Inception v4 [7] – and their ensembles described in Section 3. The TensorFlow-Slim API was used to adjust and fine-tune the networks from the publicly available ImageNet-pretrained checkpoints[4]. All networks in our experiments shared the optimizer settings enumerated in Table 1. Batch size, input resolution and random crop area range were set differently for each network listed in Table 2. The following image pre-processing was used for training:

- Random crop, with aspect ratio range $(0.75, 1.33)$ and with different area ranges listed in Table 2,
- Random left-right flip,
- Brightness and Saturation distortion.

At test-time, 14 predictions per image are generated by using 7 crops and their mirrored versions:

- 1x Full image,
- 1x Central crop covering 80% of the original image,
- 1x Central crop covering 60% of the original image,
- 4x corner crops covering 60% of the original image.

Table 2. Networks and hyper-parameters used in the experiments:

#	Net architecture	Batch size	Input Resolution	Random crop area
1	Inception-ResNet v2	32	299×299	5% - 100%
2	Inception-ResNet v2	16	498×498	25% - 100%
3	Inception-ResNet v2	16	498×498	5% - 100%
4	Inception v4	32	299×299	5% - 100%
5	Inception v4	32	598×598	5% - 100%
6	Inception v4	32	299×299	50% - 100%

[4] `https://github.com/tensorflow/models/tree/master/research/slim#Pretrained`

2.2 Fine-tuning and Data Splits

Networks #1,..,#6, initialized from the ImageNet pre-trained checkpoints, were first trained on PlantCLEF data from previous years (PlantCLEF 2017 EOL + PlantCLEF 2017 web + PlantCLEF 2016 subset). PlantCLEF 2017 test set was used for validation.

Another set of networks, denoted as $\#1^{\text{clean}}$,..,$\#6^{\text{clean}}$, was fine-tuned from models #1,..,#6 without using the noisy PlantCLEF 2017 web set. For this fine-tuning, we also added most of the PlantCLEF 2017 test set, keeping only 1 000 observations (1 403 images) as a min-val set.

2.3 Running Averages

Preliminary experiments, using the 2017 test set for validation, showed a significant improvement in accuracy when using running averages of the trained variables instead of the values from the last training step. Namely we used an exponential decay with decay rate of 0.999.

In this task where majority of the training data is noisy, we interpret this as keeping a stable version of the variables, since mini-batches with noisy samples may produce large gradients pointing outside of the local optima. Another possible interpretation is that the learning rate was still too high. Unfortunately, we did not have the computational time to experiment with different learning rate schedules.

2.4 Class Prior Estimation

In many computer vision tasks, the class prior probabilities are assumed to be the same for the training data and test data. In ExpertLifeCLEF, however, it is reasonable to assume that class priors change: The largest part of the training set comes from the web, where the class frequencies may not correspond with the test-time priors (dependening on the species incidence, the interest of users, etc.). The problem of adjusting CNN outputs to the change in class prior probabilities was discussed in [6], where it was proposed to recompute the posterior probabilities (predictions) $p(c_k|\mathbf{x}_i)$ by Equation 1.

$$p_e(c_k|\mathbf{x}_i) = p(c_k|\mathbf{x}_i)\frac{p_e(c_k)p(\mathbf{x}_i)}{p(c_k)p_e(\mathbf{x}_i)} = \frac{p(c_k|\mathbf{x}_i)\frac{p_e(c_k)}{p(c_k)}}{\sum_{j=1}^{K} p(c_j|\mathbf{x}_i)\frac{p_e(c_j)}{p(c_j)}} \propto p(c_k|\mathbf{x}_i)\frac{p_e(c_k)}{p(c_k)}, \quad (1)$$

The subscript e denotes probabilities on the evaluation/test set. The posterior probabilities $p(c_k|\mathbf{x}_i)$ are estimated by the Convolutional Neural Network outputs, since it was trained with the cross-entropy loss. For class priors $p(c_k)$ we have an empirical observation - the class frequency in the training set. The evaluation/test set priors $p_e(c_k)$ are, however, unknown.

We follow the proposition from [6] to use an existing EM algorithm [5] for estimation of test set priors by maximization of the likelihood of the test observations. The E and M step are described by Equation 2, where the super-scripts (s) or $(s+1)$ denote the step of the EM algorithm.

$$p_e^{(s)}(c_k|\mathbf{x}_i) = \frac{p(c_k|\mathbf{x}_i)\frac{p_e^{(s)}(c_k)}{p(c_k)}}{\sum_{j=1}^{K} p(c_j|\mathbf{x}_i)\frac{p_e^{(s)}(c_j)}{p(c_j)}}, \qquad (2)$$

$$p_e^{(s+1)}(c_k) = \frac{1}{N}\sum_{i=1}^{N} p_e^{(s)}(c_k|\mathbf{x}_i),$$

In our submissions, we estimated the class prior probabilities for the whole test set. However, one may also consider estimating different class priors for different locations, based on the GPS-coordinates of the observations. Moreover, as discussed in [6], one may use this procedure even in the cases where the new test samples come sequentially.

3 Submissions

In the challenge, each team was allowed to submit up to 5 different run-files with predictions. We used this opportunity to evaluate the following 5 submissions:

CMP Run 1 is an ensemble of 6 CNNs: #1$^{\text{clean}}$,..,#6$^{\text{clean}}$ described in Section 2.2. This submission used the automatic test set class-prior estimation from the CNN outputs, discussed in Section 2.4.

CMP Run 2 was predicted by the ensemble from Run 1 without class prior estimation on the test data.

CMP Run 3 is an ensemble of 12 CNNs: #1,..,#6 described in Section 2.1 and #1$^{\text{clean}}$,..,#6$^{\text{clean}}$ described in Section 2.2. This submission used the automatic test set class-prior estimation.

CMP Run 4 is an ensemble of 6 CNNs: #1,..,#6 described in Section 2.1. This submission used the automatic test set class-prior estimation.

CMP Run 5 is a single Inception-v4 model, denoted as CNN #4$^{\text{clean}}$, using the automatic test set class-prior estimation.

In all runs, the predictions (optionally improved by the class prior estimation) for all crops of the test image are averaged to compute the final image prediction. Moreover, for observations with several images (connected by the ObservationID values in the provided data), the final classification decision is taken based on the average of all corresponding image predictions.

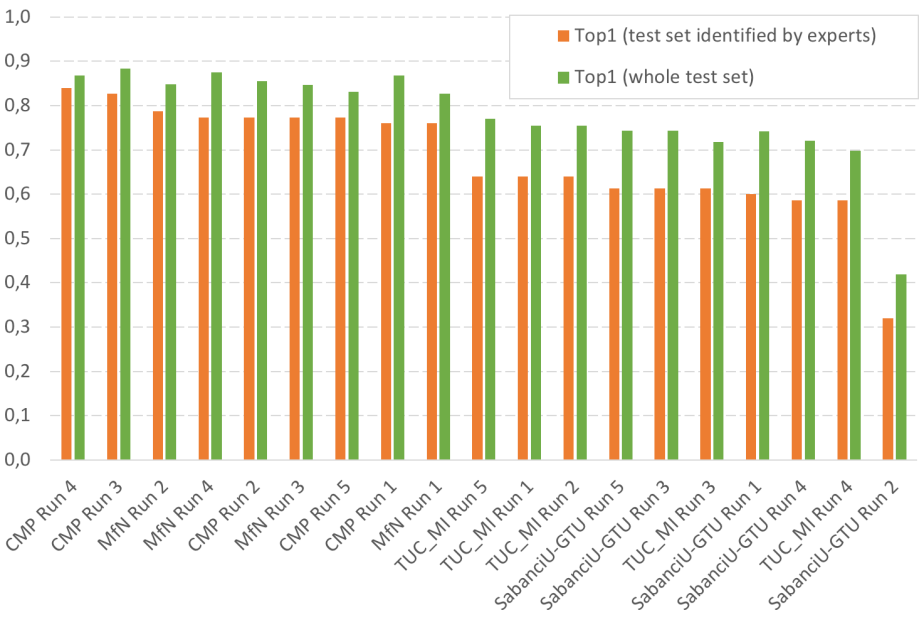

Fig. 2. Results of runs submitted by the challenge participants.

4 Results

The official results of the challenge are displayed in Figure 2. Our system achieved the best results among automatic methods: 88.4% accuracy on the full test set. The best scoring submission was the largest ensemble - CMP Run 3 - using all 12 models. Results of all CMP submissions are listed in Table 3.

When evaluated against human experts in plant sciences, the system (both CMP Run 3 and CMP Run 4) outperformed 5 of 9 tested human experts. That means that in the task of plant recognition from images, machine learning systems reached human expert performance - achieving better accuracy than the median of human experts. The detailed results are displayed in Figure 3.

Interestingly, while fine-tuning on "clean" data slightly improved the recognition accuracy on the full test set, it significantly decreased the accuracy on the test subset for human experts. Similarly, test-time prior estimation on the full test set noticeably improved the accuracy, but had an opposite effect on the subset. We assume that the test subset selected for human experts was too

Table 3. Results of CMP submissions on the full test set and its subset for human experts.

CMP Run	1	2	3	4	5
Accuracy (full test set)	86.8%	85.6%	**88.4%**	86.7%	83.2%
Accuracy (smaller test set)	76.0%	77.3%	82.7%	**84.0%**	77.3%

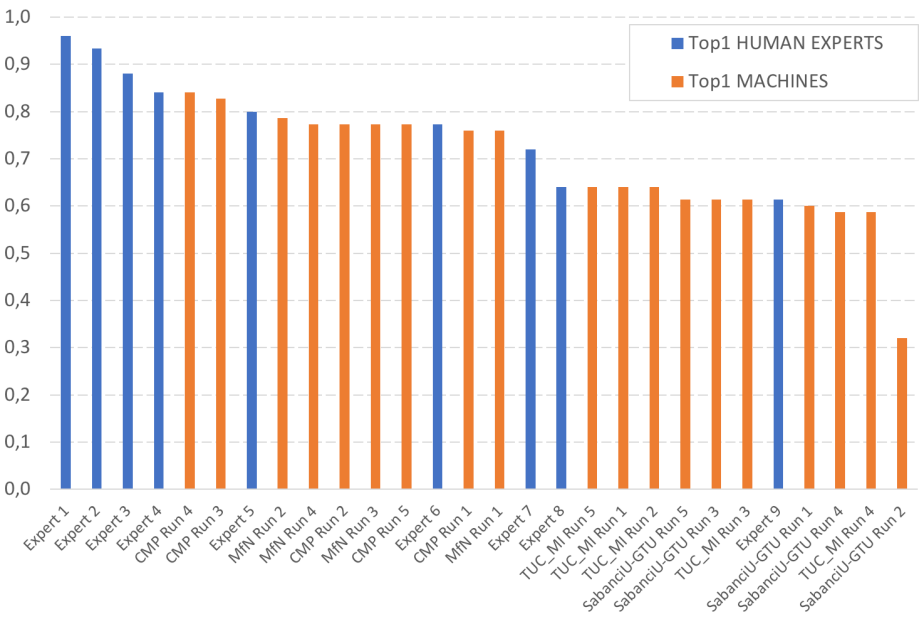

Fig. 3. Results of the "Experts vs Machines" experiment.

small to provide a representative, identically distributed, sample of the full test set. Therefore the results on the test subset for human experts may be biased towards a small number of species contained in it.

5 Conclusions

The proposed machine-learning system for recognition of 10 000 plant species achieved an excellent accuracy of 88.4% in the ExpertLifeCLEF 2018 challenge, scoring 1st among automated systems.

The ensemble of Convolutional Neural Networks benefited from the following improvements:

1. Adjusting the CNN predictions according to the estimated change of the class prior probabilities.
2. Replacing network parameters by their running averages with exponential decay.
3. Test-time data augmentation.

The experiment with human experts shows that machine learning reached the expert knowledge in plant recognition: our system scored better than an average (median) human expert in plant recognition, achieving better recognition rate than 5 of the 9 evaluated human experts.

Acknowledgements

MS was supported by the CTU student grant SGS17/185/OHK3/3T/13 and by the Electrolux Student Support Programme. JM was supported by The Czech Science Foundation Project GACR P103/12/G084 P103/12/G084. LP was supported by the UWB project No. SGS-2016-039.

References

1. Goëau, H., Bonnet, P., Joly, A.: Plant identification in an open-world (lifeclef 2016). In: CLEF working notes 2016 (2016)
2. Goëau, H., Bonnet, P., Joly, A.: Plant identification based on noisy web data: the amazing performance of deep learning (lifeclef 2017). CEUR Workshop Proceedings (2017)
3. Goëau, H., Bonnet, P., Joly, A.: Overview of expertlifeclef 2018: how far automated identification systems are from the best experts ? In: CLEF working notes 2018 (2018)
4. Joly, A., Goëau, H., Botella, C., Glotin, H., Bonnet, P., Planqué, R., Vellinga, W.P., Müller, H.: Overview of lifeclef 2018: a large-scale evaluation of species identification and recommendation algorithms in the era of ai. In: Proceedings of CLEF 2018 (2018)
5. Saerens, M., Latinne, P., Decaestecker, C.: Adjusting the outputs of a classifier to new a priori probabilities: a simple procedure. Neural computation **14**(1), 21–41 (2002)
6. Sulc, M., Matas, J.: Improving cnn classifiers by estimating test-time priors. arXiv preprint arXiv:1805.08235 (2018)
7. Szegedy, C., Ioffe, S., Vanhoucke, V.: Inception-v4, inception-resnet and the impact of residual connections on learning. CoRR **abs/1602.07261** (2016), `http://arxiv.org/abs/1602.07261`