Image-based Plant Species Identification with Deep Convolutional Neural Networks

Mario Lasseck

Museum für Naturkunde Berlin, Germany
Mario.Lasseck@mfn-berlin.de

Abstract. This paper presents deep learning techniques for image-based plant identification at very large scale. State-of-the-art Deep Convolutional Neural Networks (DCNNs) are fine-tuned to classify 10,000 species. To improve identification performance several models trained on different datasets with multiple image dimensions and aspect ratios are ensembled. Various data augmentation techniques have been applied to prevent overfitting and to further improve model accuracy and generalization. The proposed approach is evaluated in the LifeCLEF 2017 campaign. It provides the best system among all participating teams by achieving a mean reciprocal rank (MRR) of 92 % and a top-5 accuracy of 96 % on the official PlantCLEF test set.

Keywords: Plant Species Identification, Biodiversity, Deep Learning, Convolutional Neural Networks, Fine-grained Image Classification, Data Augmentation

1 Introduction

Image-based plant identification is a promising tool supporting agriculture automation and environmental conservation efforts. It can be used via mobile applications like Pl@ntNet [2] or Naturblick [3] for education, biodiversity monitoring and the collection of plant observation records either by professionals or in a citizen science context. It helps to bridge the taxonomic knowledge gab and offers new interactive and efficient ways of browsing large image collections of flora.

Distinguishing between 10,000 individual plant species is a challenging fine-grained classification problem. One has to deal with categories that are very similar and often share a common part structure leading to low inter-class variations. On the other hand, plants are extremely diverse in size, shape, color and texture. Furthermore images of a particular species can contain different plant organs or content types. A single image can either show an entire plant or just a small part of it (e.g. flower, fruit, branch, stem or leaf) with significant changes of appearance throughout the year leading to high intra-class variations.

The LifeCLEF 2017 plant identification challenge aims to evaluate image-based plant identification systems close to conditions of real-world biodiversity monitoring scenarios at a very large scale. This year the LifeCLEF evaluation campaign provides two main datasets and participants are encouraged to evaluate and compare classifica-

tion results using either one or both of them for training. The "trusted" training set is based on the online collaborative Encyclopedia of Life [1] with ca. 260,000 images coming from several public databases (Wikimedia, iNaturalist, Flickr, etc.) and institutions or websites dedicated to botany. Additionally up to 100,000 labeled and "trusted" images from previous campaigns are also provided. The second much larger "noisy" training set is built by web crawlers (e.g. Google and Bing image search). It contains over 1.4 million images among them many with wrong content (wrong species, portrait of a botanist working on a species, drawings, herbarium sheet of a dry specimen, etc.). All in all over 1.7 million images of 10,000 species of wild, cultivated, ornamental and endangered plants mostly coming from Western Europe and North American flora with different types of views (branch, entire plant, flower, fruit, leaf, stem, bark, scans of leaf, etc.) are provided and can be used for training. For evaluation a test set of 25,170 images belonging to 17,868 plant observations is provided. The test images represent typical smartphone application queries from Pl@ntNet and need to be classified by identifying the correct species for each observation. More information on datasets and task can be found in the LifeCLEF 2017 Lab Overview [4] and the plant identification task summery [5].

2 Implementation Details and Model Training

To address the task of plant identification deep learning techniques are applied that already proved to be very successful in other image-based object classification scenarios. Several models are trained and prediction results are later bagged to increase identification accuracy on the test set. For late fusion, ensembles tend to yield better results if there is a significant diversity among the models [14], [15]. In order to generate a diverse set of models the following aspects are varied across them:

- network architecture
- batch size
- solver type
- learning rate schedule (base learning rate and decay factor)
- training dataset
- random partition of datasets for training and validation
- random seed (for weight initialization and model training)
- image dimension
- image aspect ratio
- crop size
- data augmentation (techniques and strengths of influence)

Three different state-of-the-art Deep Convolutional Neural Network architectures are used for training and classification:

- GoogLeNet [6]
- ResNet [7]
- ResNeXT [8]

GoogLeNet won the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) [9] in 2014 and was previously used for plant identification by [17], [20] and [21]. ResNet won ILSVRC and the Common Objects in Contexts (COCO) Detection Challenge [10] in 2015. For plant species recognition it was successfully used by [18] in the PlantCLEF 2016 challenge. ResNeXT ranked 2nd place in the ILSVRC 2016 classification task. It has not been used for plant identification before. The latter two residual network architectures allow to efficiently train very deep networks where each layer does not need to learn the whole feature space transformation but only a residual correction to the previous layer. The models trained for the current plant identification task are using the 152 layer version of ResNet (ResNet-152) and a 101 layer version with cardinality = 64 and bottleneck width = 4d of the ResNeXt architecture (ResNeXt-101-64x4d).

Instead of starting from scratch all networks are trained via transfer learning by fine-tuning models pre-trained on the ImageNet dataset [11]. GoogLeNet and ResNeXt-101-64x4d had been pre-trained on the ILSVRC version of the dataset containing 1.2 million images of 1000 object categories while ResNet-152 had been trained on the 10 times larger complete ImageNet dataset covering 11k object classes. All pre-trained models mentioned here are publically available and can be downloaded for the corresponding frameworks [24], [25], [26]. Previous work [19] but also own experiments with the trusted training set suggested that starting training with pre-trained models leads to better results and faster convergence. Figure 1 shows progress of validation accuracy over 40 training epochs using a pre-trained GoogLeNet model compared to a model started from scratch with randomly initialized weights.

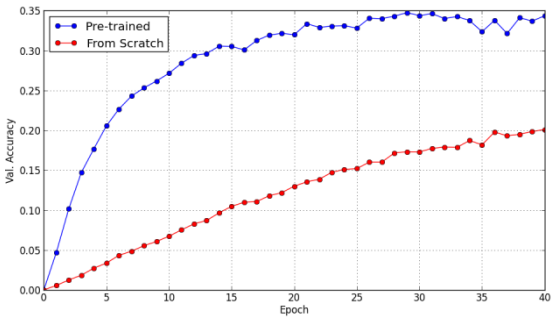

Fig. 1. Progress in validation accuracy using pre-trained networks vs. training from scratch.

To fine-tune a model the last fully-connected layer of the network is replaced and adapted to the 10k classes problem of the plant identification task. Before training – instead of random initialization – all weights except the ones of the exchanged layer are initialized with the pre-trained parameters.

In case of GoogLeNet, NVIDIA Digits [12] in combination with the Caffe framework [13] is used for training, data preparation and classification. The residual networks are trained via the MXNet framework [14]. All models are using either 2

NVIDIA GeForce GTX 1080 or 2 NVIDIA GeForce GTX 1080 Ti GPUs in parallel. Batch sizes are chosen mostly as large as possible to not run out of GPU memory. For some GoogLeNet models Caffe's *iter_size* parameter is applied to accumulate gradients over 2 batches. A fixed learning rate policy is used starting with a base learning rate around 0.01. It is decreased by a factor between 2 and 10 whenever validation loss or accuracy is not improving any longer. This is usually done twice within the entire training process. Stochastic Gradient Descent (SGD) and for some GoogLeNet models Nesterov's Accelerated Gradient (NAG) is used for model optimization. Since GoogLeNet is not providing batch normalization a mean image is computed from the training set and subtracted from all images.

Data Preparation

To evaluate to what extent DCNNs can learn from noisy data compared to trusted data, some models are trained using only images of the Encyclopedia of Life dataset (plus in some cases images of the PlantCLEF 2016 dataset) while others are trained with all available images. It was also tried to form a mixture of both main datasets (see section Submissions and Results). Besides that, random partitions (stratified folds) of the datasets are created, so each model can use a different fold for validation and the remaining folds for training. The bagging of models trained on different folds was previously successfully applied by [18] and the winning team of the PlantCLEF 2015 task [22]. Instead of applying advanced methods like Borda-fuse as in [22] or taking the species-wise maximum as in [18], in this work only simple averaging is performed for model ensembling.

Original images for training and testing are of arbitrary dimensions and aspect ratios. Since the networks used in this work only accept fixed sized square images as input, images need to be preprocessed via rescaling and/or cropping. To gain diversity across models, training images are rescaled to various dimensions and sometimes aspect ratios are additionally changed. Ensembling models using different image scales already improved results in previous PlantCLEF tasks [23], [19]. Scaling images to different dimensions followed by random cropping helps to improve generalization by letting the network see patches of slightly different sections and resolutions of the original image. For GoogLeNet models, images are scaled to the following dimensions before random cropping is applied on-the-fly during training:

- 256x256 pixel
- 250x250 pixel
- 240x240 pixel

When using Digits, different resize transformations are chosen to handle non-square images. Figure 2 visualizes the four transformation options offered by Digits. To not lose too much information per image only option 3 *Half crop, half fill* and 4 *Squash* which includes a warping of images by changing its aspect ratio is used in submission models.

Fig. 2. Digits resize transformation options: 1. *Crop* 2. *Fill* 3. *Half crop, half fill* 4. *Squash*

For models trained via the MXNet framework all images are resized such that the shorter side becomes 256 pixel while preserving the original aspect ratio. This is also done for all images in the test set. For test images additionally 5 patches are extracted by cropping square patches – the same size as used for model training – from each corner plus the center of the image. Predictions of these patches are averaged to gain a more robust classification result per image.

Data Augmentation

During training square patches are cropped in real-time from each image at random positions to serve as network input. Most models use input patches of size 224x224 pixel except for one GoogLeNet model which uses input patches of 230x230 pixel. After cropping, horizontal flipping is applied to randomly chosen patches. For residual networks additional data augmentation techniques were explored. For submission models the following image manipulation methods are used on-the-fly during training:

- rotation by random angle
- random variation of saturation (S channel in HSL color space)
- random variation of lightness (L channel in HSL color space)

Image patches are rotated by an angle randomly chosen between ± 45°. Color variation is applied in the HSL color space by adding values randomly chosen between ± 32 for saturation and ± 20 for lightness to the S and L channel (originally ranging from 0 to 255). When learning rate is decreased during training the maximum values for rotation angle and color variation are also decreased, letting the network see patches closer to the original image at the end of each training procedure.

Figure 4 to 7 show augmentation examples combining random cropping, horizontal flipping, rotation and variations of saturation and lightness. Image sources with original aspect ratios are visualized in figure 3. The corresponding plant species and MediaIds are from left to right: Leucanthemum vulgare (MediaId: 254374), Streptanthus polygaloides (MediaId: 351199), Wikstroemia uva-ursi (MediaId: 378991) and Ipomoea sagittata (MediaId: 243459).

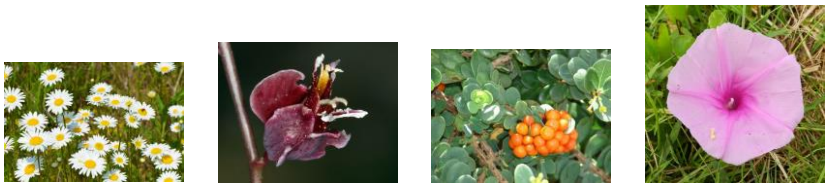

Fig. 3. Image examples with original aspect ratio

Fig. 4. Augmentation examples (MediaId 254374)

Fig. 5. Augmentation examples (MediaId 351199)

Fig. 6. Augmentation examples (MediaId 378991)

Fig. 7. Augmentation examples (MediaId 243459)

3 Submissions and Results

Identification performance of different model ensembles is evaluated on the official PlantCLEF 2017 test set. All submitted runs are created by aggregating predictions of

test images using models of all three network architectures trained with different datasets and configurations. For each observation predictions are averaged over:

- all images belonging to the same observation
- all 5 patches cropped from the resized image
- all models within the ensemble

Run 1
For the first run three models (one per network architecture) are ensembled using only images from the trusted datasets Encyclopedia of Life (E) and PlantCLEF 2016 (P) for training.

Run 2
For the second run an ensemble of six models containing four GoogLeNets, one ResNet-152 and one ResNeXt-101-64x4d is trained using images from datasets E and P plus all images from the noisy web dataset (W).

Run 3
For the third run the trusted datasets E and P are augmented by "good" (trustworthy) images from the noisy web dataset, as well as some images from the test set. To accomplish this, the web set is filtered by using models from the first run to identify plant species on web images. Correctly classified files (highest prediction equals ground truth) are selected to form a new filtered web dataset (FW) of 508,802 images. Furthermore some images from the test set are also added to the training set. Here the residual networks of run 1 and 2 are used to gather images with a prediction score greater than 0.98. The resulting set of 18,217 images is additionally filtered by choosing only images with at least 6 out of 20 predictions (5 predictions per image of 4 networks) suggesting the same species without confusion by any other species. This way, 9934 images from the test set are selected (T) and also added to the training set. Two GoogLeNet models pre-trained with data from the second run and one ResNeXt model from the first run are fine-tuned with the new formed trusted training set (E, P, FW, T) and used to aggregate predictions for the third run.

Run 4
For the final and best scoring fourth run, all 12 models from the previous runs are ensembled and their predictions bagged via averaging.

Table 1 gives an overview of number of models, datasets used for training and the official results for each submitted run. Identification performance is measured by evaluating mean reciprocal rank (MRR), top-1 and top-5 accuracy on the test set.

Table 1. Number of models, training datasets and performance results of submitted runs

Run	# Models	Datasets	MRR [%]	Top1 Acc. [%]	Top5 Acc. [%]
1	3	E,P	84.7	79.4	91.1
2	6	E,P,W	91.5	87.7	96.0
3	3	E,P,FW,T	89.4	85.7	94.0
4	12	E,P,W,FW,T	92.0	88.5	96.2

More information on models, their individual configurations, training details, validation scores and datasets used for each run can be found in a separate excel sheet [27]. It should be mentioned, that during training, scores on validation sets were in most cases significantly lower than scores on the PlantCLEF test set. Especially when using images from the noisy dataset for training and validation, accuracy dropped to around 52 %. When looking at validation scores in [27] one has to take into account that models use different datasets and folds for validation. Only scores of models using the exact same datasets are comparable. Table 2 illustrates three examples of models using identical subsets for training and validation.

Table 2. Implementation details of three models using all available images of the trusted and noisy datasets and identical subsets for training and validation

Model-ID	M5	M8	M9
Network architecture	GoogLeNet	ResNet-152	ResNeXt-101-64x4d
Framework	Digits/Caffe	MXNet	MXNet
Pre-trained via	ImageNet-1k	ImageNet-11k	ImageNet-1k
Datasets	E, P, W	E, P, W	E, P, W
Validation subset	2nd fold	2nd fold	2nd fold
Image size [px]	250x250	shorter side 256	shorter side 256
Resize transformation	half crop, half fill	-	-
Crop size [px]	224x224	224x224	224x224
Augmentation	crop, mirror	crop, mirror, rotation, saturation, lightness	crop, mirror, rotation, saturation, lightness
Learning rate schedule	0.015 40 epochs, 0.001 35 epochs, 0.0005 28 epochs, 0.0001 26 epochs	0.01 26 epochs, 0.001 9 epochs, 0.0001 10 epochs	0.01 15 epochs, 0.001 12 epochs
Batch size	128	44	48
Solver Type	SGD	SGD	SGD
Val. set accuracy [%]	52.74	52.53	52.99
Included in run	2 & 4	2 & 4	2 & 4

Figure 8 compares scores of all submissions to the LifeCLEF 2017 plant identification task. Run 2, 3 and 4 (MarioTsaBerlin) belong to the best scoring submissions among all participating teams. Especially run 2 and 4 show outstanding performances with a MRR of over 91 % and a top-5 accuracy of 96 % on the test set. More results and evaluation details can be accessed via the PlantCLEF 2017 homepage [28].

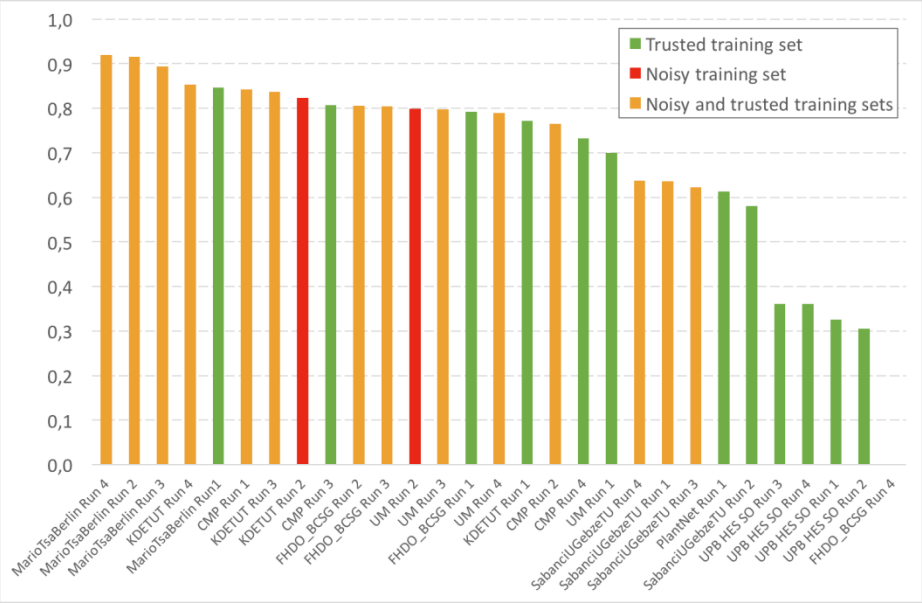

Fig. 8. Official scores [MRR] of the LifeCLEF 2017 plant identification task. The above described methods and submitted runs belong to MarioTsaBerlin.

4 Discussion

State-of-the-art DCNNs are powerful tools to identify objects in images. By fine-tuning pre-trained models (originally trained to classify ImageNet categories like cars, dogs, cats, etc.) they can be adapted to identify a large number of different plant species. By bagging several diverse models, identification performance can be significantly increased. To gain diversity, models are trained on different folds using various datasets, image dimensions, aspect ratios and augmentation techniques.

Submission results suggest that if the number of training examples is sufficiently high DCNNs are capable of handling quite a fairly large amount of noise in the training set. For the third run the trusted training set is augmented by adding “good” examples from the noisy web dataset. To sort out which images contain correct content of plant species, DCNNs previously trained with trusted datasets are used to filter the web set. Although increasing identification accuracy by almost 8 % for models using the augmented dataset, it still does not reach the performance of models using all

available data for training including many images with wrong content. A different filter approach, focused on discarding “bad” images rather than including “good” ones, might lead to better results and would be worth investigating in the future.

Due to time constraints it was not possible to systematically investigate the influence of certain settings on single network architectures. For this to be done, individual parameters need to be changed while keeping all other configurations the same. This was given up in favour of producing models with high diversity.

Unfortunately training from scratch was only carried out for a relatively small number of iterations using the trusted dataset exclusively. It was abandoned in favour of fine-tuning pre-trained models which seemed to be more promising at that time. Nevertheless it would be interesting to train networks from scratch for a longer period of time using all available images and to compare results with fine-tuned networks.

The models trained in this work will be further developed and later integrated into the mobile application Naturblick. The application provides information about urban nature in Berlin and offers different species identification tools including audio-based bird identification using algorithms developed and evaluated in previous LifeCLEF campaigns [29], [30].

Acknowledgments. I would like to thank Hervé Goëau, Alexis Joly, Pierre Bonnet and Henning Müller for organizing this task. I also want to thank the BMUB (Bundesministerium für Umwelt, Naturschutz, Bau und Reaktorsicherheit), the Museum für Naturkunde Berlin and especially the Naturblick project team for supporting my research.

References

1. Encyclopedia of Life Homepage, http://eol.org/, last accessed 2017/05/21
2. Goëau H, Bonnet P, Joly A et al. (2013) Pl@ntNet mobile app. In: Proceedings of the 21st ACM international conference on Multimedia, pp 423-424, 2013
3. Naturblick App Homepage, http://naturblick.naturkundemuseum.berlin/, last accessed 2017/05/21
4. Joly A, Goëau H, Glotin H, Spampinato C, Bonnet P, Vellinga WP, Lombardo JC, Planqué R, Palazzo S, Müller H (2017) LifeCLEF 2017 Lab Overview: multimedia species identification challenges. In: Proceedings of CLEF 2017
5. Goëau H, Bonnet P, Joly A (2017) Plant identification based on noisy web data: the amazing performance of deep learning (LifeCLEF 2017). In: CLEF working notes 2017
6. Szegedy C et al. (2014) Going Deeper with Convolutions, In: arXiv:1409.4842, 2014
7. He K, Zhang X, Ren S, Sun J (2016) Deep Residual Learning for Image Recognition. In: CVPR, 2016
8. Xie S, Girshick R, Dollár P, Tu Z, He K (2016) Aggregated Residual Transformations for Deep Neural Networks, In: arXiv:1611.05431, 2016
9. Russakovsky O et al. (2014) ImageNet Large Scale Visual Recognition Challenge. In: arXiv:1409.0575, 2014
10. Lin TY et al. (2014) Microsoft COCO: Common Objects in Context. In: Fleet D, Pajdla T, Schiele B, Tuytelaars T (eds) Computer Vision – ECCV 2014. ECCV 2014. Lecture Notes in Computer Science, vol 8693. Springer, Cham

11. Deng J, Dong W, Socher R, Li LJ, Li K, Fei-Fei L (2009) Imagenet: A largescale hierarchical image database. In: IEEE Conference on Computer Vision and Pattern Recognition, 2009. pp. 248–255 (2009)
12. Digits Homepage, https://developer.nvidia.com/digits, last accessed 2017/05/21
13. Jia Y et al. (2014) Caffe: Convolutional Architecture for Fast Feature Embedding. In: arXiv:1408.5093, 2014
14. Chen T et al. (2016) MXNet: A Flexible and Efficient Machine Learning Library for Heterogeneous Distributed Systems. In: Neural Information Processing Systems, Workshop on Machine Learning Systems, 2016
15. Kuncheva L, Whitaker C (2003) Measures of diversity in classifier ensembles, Machine Learning, 51, pp. 181-207, 2003
16. Sollich P, Krogh A (1996) Learning with ensembles: How overfitting can be useful, Advances in Neural Information Processing Systems, volume 8, pp. 190-196, 1996
17. Ghazi MM, Yanikoglu B, Aptoula E (2016) Open-set Plant Identification Using an Ensemble of Deep Convolutional Neural Networks. In: CLEF2016 Working Notes
18. Šulc M, Mishkin D, Matas J (2016) Very Deep Residual Networks with MaxOut for Plant Identification in the Wild. In: CLEF2016 Working Notes
19. Lee SH, Chang YL, Chan CS, Remagnino P (2016) Plant Identification System based on a Convolutional Neural Network for the LifeClef 2016 Plant Classification Task. In: CLEF2016 Working Notes
20. McCool C, Ge Z, Corke P (2016) Feature Learning via Mixtures of DCNNs for Fine-Grained Plant Classification. In: CLEF2016 Working Notes
21. Champ J, Goeau H, Joly A (2016) Floristic participation at LifeCLEF 2016 Plant Identification Task. In: CLEF2016 Working Notes
22. Choi S (2015) Plant identification with deep convolutional neural network: Snumedinfo at lifeclef plant identification task 2015. In: Working notes of CLEF 2015 conference
23. Hang ST, Tatsuma A, Aono M (2016) Bluefield (KDE TUT) at LifeCLEF 2016 Plant Identification Task. In: CLEF2016 Working Notes
24. GoogLeNet Model files for the Caffe framework, https://github.com/BVLC/caffe/tree/master/models/bvlc_googlenet, last accessed 2017/05/21
25. ResNeXt-101-64x4d model files for the MXNet framework, http://data.mxnet.io/models/imagenet/resnext/101-layers/
26. ResNet-152 model files for the MXNet framework, http://data.mxnet.io/models/imagenet-11k/resnet-152/, last accessed 2017/05/21
27. Model and training details, http://www.animalsoundarchive.org/RefSys/PlantCLEF2017, last accessed 2017/05/21
28. PlantCLEF 2017 Homepage, http://www.imageclef.org/lifeclef/2017/plant, last accessed 2017/05/21
29. Lasseck M (2015) Towards Automatic Large-Scale Identification of Birds in Audio Recordings. In: Mothe J. et al. (eds) Experimental IR Meets Multilinguality, Multimodality, and Interaction. Lecture Notes in Computer Science, vol 9283. Springer, Cham
30. Lasseck M (2016) Improving Bird Identification using Multiresolution Template Matching and Feature Selection during Training. In: CLEF2016 Working Notes