LifeClef 2017 Plant Identification Challenge: Classifying Plants Using Generic-Organ Correlation Features

Sue Han Lee, Yang Loong Chang, and Chee Seng Chan

Centre of Image & Signal Processing, Fac. Comp. Sci. & Info. Tech.,
University of Malaya, Malaysia
{leesuehan,yangloong}@siswa.um.edu.my
cs.chan@um.edu.my

Abstract. This paper describes our proposal in the multi-organ plant identification task (LifeClef2017 challenge [8]). The objective of the challenge is to evaluate to what extent machine learning and computer vision can learn from noisy data compared to trusted data. To address the challenge, we employ our recent proposed hybrid generic-organ convolutional neural network, abbreviated HGO-CNN [11] to train on different composition of plant datasets. Overall, all the submitted runs obtained comparable results in the LifeClef2017 plant classification task.

Keywords: Plant classification, deep learning, convolutional neural network

1 Introduction

Plant classification has received particular attention in the computer vision field [10] due to its important implications in agriculture automation and environmental conservation. Along with the recent advances in science and technology, automatic plant species recognition has been made possible to assist botanists in plant identification tasks [10]. For example: development of an efficient plant recognition system using the Local Binary Pattern [13] allows the classification of medical plants [12]. Robotic weed control system drives studies on automatic plant identification in agronomic research aimed at crop improvement by recognition of crop plants and elimination of weeds [5]. Despite these, automatic plant recognition, a foundational capability in this context, is nevertheless still in its early stages.

In 2013, LifeClef challenge [4] provided the first multi-organ plant dataset. This was the first multi-organ plant classification benchmark for the computer vision community. This year, LifeClef 2017 [8] offered a bigger amount of plant biodiversity data [3]. The objective is to identify 10000 species from images of plants collected based on two different channels: a "trusted" training set and a "noisy" training set. The trusted training set is collected from the online collaborative Encyclopedia Of Life (EoL) such as Wikipedia, iNaturalist and Flickr

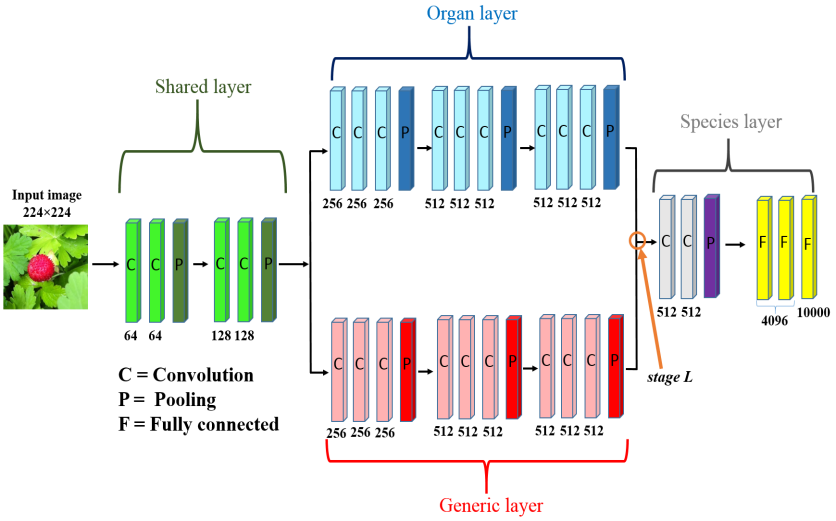

Fig. 1: Our proposed HGO-CNN architecture

while the noisy training set is collected based on the Google and Bing image search results. In this challenge, we employ our recently proposed convolutional neural network (CNN) architecture – namely the HGO-CNN [11] with small refinements. Specifically, it integrates both the generic and organ-specific information for the multi-organ plant classification task.

The rest of the working note is organized as follows. In Section 2, we present the methodology of our proposed architecture. Section 3 illustrates its training scheme. Section 4 shows the experiments and results for both the validation and testing set. Lastly, Section 5 presents conclusion and future work.

2 Method Description

Unlike previous approaches [1, 2] that trained CNN to capture solely generic representation from the plantation images, HGO-CNN [11] is able to encapsulate both the organ and generic information prior to the plant classification. We consider features from the organ because plant organs in general, are known prior to the exploration of its characteristics. For instant, when botanists study a **leaf**, they focus on the leaf characters such as *margin* or *venation*, and, when they study a **flower**, they focus on the characteristics of *petals*, *sepals* and *stamen* to identify the plant species. So, we believe that a better recognition method for plant species requires prior information of their respective organs.

The proposed HGO-CNN comprises of four layers or components: (i) a shared layer, (ii) an organ layer, (iii) generic layer, and (iv) a species layer. We introduce shared layer for both the generic and organ components. The reasons are

threefold. First, [17, 16] demonstrated that preceding layers in deep networks response to low-level features such as corners and edges. Since both the higher level generic and organ components require low-level features to build higher level features, we introduce shared preceding layers for these components. Second, according to [16], the shared layer may reduce floating point operations and memory footprint of the network execution, which are of importance for real world application. Lastly, using shared layer will help to reduce the number of training parameters, which is beneficial to the architecture's computational efficiency. Fig. 1 depicts the configuration of our proposed model. Input to our proposed model is a color image of 224 × 224 pixels. For the convolutional layer, we utilise 3 × 3 convolution filters with spatial resolution preserved using stride 1. Max pooling is performed using a 2 × 2 pixel window with stride 2. Three fully connected layers, which have 4096, 4096 and 10000 channels respectively, follow behind the stacks of convolutional layers. Finally, the HGO-CNN output is fed into a softmax layer to produce the softmax output. Note that for the q-th class, the softmax output is defined as $P_n^{(q)} = \frac{e^{s_q}}{\sum_{m=1}^{M} e^{s_m}}$ where M stands for the total number of classes and s stands for the class prediction score. After performing the softmax operation, softmax loss L is computed as follows:

$$L = \frac{1}{B} \sum_{n=1}^{B} -log(P_n^{(T)}) \tag{1}$$

where B is the batch size and T is the ground truth class label for the n-th input image.

In this challenge, we refine some of the configurations in the original HGO-CNN architecture: (1) the data layer normalization technique, called Batch Normalization (BN) [6] is included. We added BN from the last convolution layer of both the generic and organ components respectively until the fully connected layers. This is to enhance the correlation of representation learning between the two components, so that it is more robust to non-linearities. (2) During feature fusion, features summation is performed instead of concatenation to further amplify the correspondences of these features.

3 Training

Pre-Training Two-Path CNN We design a two-path CNN as shown in Fig. 2 for the purpose of training two different components: the generic and organ specific features. This two-path CNN is initially pre-trained using the ImageNet challenge dataset [14].

Organ layer After we obtained the pre-trained two-path CNN, one of the CNN path is repurposed for the organ task. This organ layer is trained together with the shared layer, using seven kinds of predefined organ labels. We obtain organ-based feature maps, $\mathbf{x}^{\mathbf{org}} \in \mathbb{R}^{H \times W \times Z}$ where H, W and Z are the height, width and number of channels of the respective feature maps. Since PlantClef2017

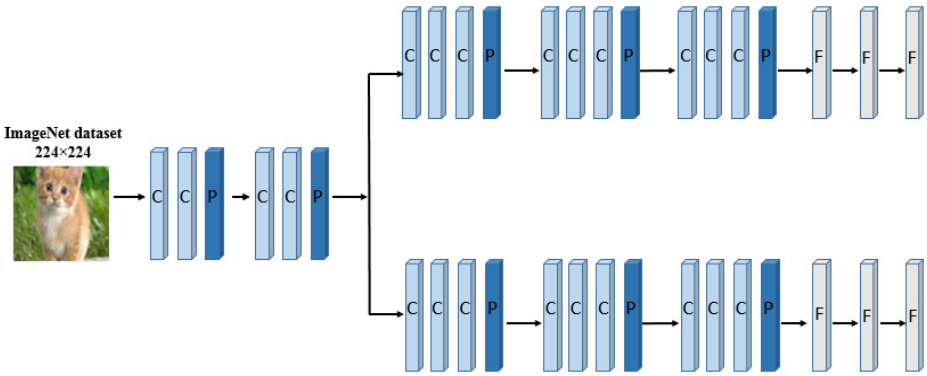

Fig. 2: A two-path CNN architecture

dataset does not provide organ information for every plant image, we train the organ layer based on the previous PlantClef2015 training set.

Generic layer After training the organ layer, another CNN path is repurposed for the generic task. This generic layer is trained using the species labels, regardless of organ information. We obtain generic-based feature maps, $\mathbf{x}^{\mathbf{gen}} \in \mathbb{R}^{H \times W \times Z}$. To allow both the organ and generic layers to share the common proceeding layer, we keep the shared layer's weights to be consistent. This is achieved by setting their learning rate to zero.

Species layer To introduce correlation between both the organ and generic components, a fusion function $\mathbf{y} = g(\mathbf{x}^{\mathbf{org}}, \mathbf{x}^{\mathbf{gen}})$ is employed at stage L (after the last convolutional layer for both components as shown in Fig. 1) to produce the organ and generic correlation feature maps, $\mathbf{y} \in \mathbb{R}^{H \times W \times Z}$. In our model, g performs summation of these two sets of features:

$$\mathbf{y}_{i,j,k} = \mathbf{x}^{\mathbf{org}}{}_{i,j,k} + \mathbf{x}^{\mathbf{gen}}{}_{i,j,k} \tag{2}$$

where $1 \leq i \leq H$, $1 \leq j \leq W$, $1 \leq k \leq Z$. The feature maps, $\mathbf{y}$ will then go through two convolution layers to learn the combined representation of generic and organ features. Since these two convolution layers are new randomly-initialised, we set their learning rate to be 10 times higher than the other layers during training.

4 Experiments and Results

Our architecture is trained using the *Caffe* [7] framework. The networks are trained with back-propagation, using stochastic gradient descent [9]. For the training parameter setting, we employed the fixed learning policy. We set the learning rate to 0.01, and then decreased it by a factor of 10 when the validation

set accuracy stops improving. The momentum was set to 0.9 and the weight decay to 0.0001. We run the experiments using a NVIDIA K40 graphics card.

4.1 Data Preparation

For the *trusted* training set, we first downloaded all 256287 images. We then randomly selected 208878 images for training and 47409 images for validation. To increase the robustness of the system in recognising multi-organ plant images, a multi-scale training was adopted. We isotropically rescaled the training images into three different sizes: 256, 385 and 512, then randomly cropped 224*224 pixels from the rescaled images to feed into the network for training. By doing this, the crop from the larger scaled images will correspond to a small part of the image or particularly subpart of the organ; while the crop from the smaller scaled images will correspond to the global structure of a plant. Besides that, we also increased the data size by mirroring the input image during training. After the data augmentation, we obtained 626634 training images and a validation set of 142227 images.

However, for the *noisy* dataset, we only managed to crawl up to 918216 number of images which is about 60% of the total number of images from the web due to resource limitations. We then separated it into 738716 images for training and 179500 images for validation. We performed the same data augmentation to produce another training set that contains 2216148 images and a validation set of 538500 images. For the testing set, all 25170 images are downloaded and similarly augmented.

4.2 Experimental results on validation set

For the evaluation of our validation set, the softmax output from our CNN model for each image was first collected. An averaging fusion was then used to combine the softmax scores of the augmented validation set. In this experiment, we computed the top-1 classification result (Acc) to infer the robustness of the system. We compared our method to the generic network, VGG-16 net [15].

Table 1: Performance comparison

Method	Trusted (Acc)	Noisy (Acc)
Finetuned VGG-16 top layer	0.44	**0.44**
HGO-CNN	**0.45**	0.42

Table 1 shows the comparison of the performance results. We can observe that the VGG-16 net performed better in the noisy dataset while our proposed HGO-CNN performed better in the trusted dataset. There are two possible reasons: (1) the organ layer in HGO-CNN that was trained on previous PlantClef2015 dataset might not be robust enough to model such a huge and diverse data, (2)

noisy dataset in this case is better modeled using generic features regardless of the organ information as the generic features might include many independent plant features that help in the classification performance.

4.3 Experimental Results on Test set

We submitted four runs to the LifeClef 2017 challenge. We finetuned all models using both the training and validation set to increase the robustness of the models. To obtain the observation level predictions, an averaging fusion method was employed to combine the results of the testing images that have the same observation id. Their performance were evaluated based on Mean Reciprocal Rank (MRR). The characteristics of each run are stated below:

- UM Run 1: Proposed HGO-CNN that trained with the trusted set only.
- UM Run 2: VGG-16 net that trained with the noisy set only.
- UM Run 3: Combined results of UM Run 1 and UM Run 2 based on averaging fusion at image level.
- UM Run 4: Combined results of UM Run 1 and UM Run 2 based on max voting at image level.

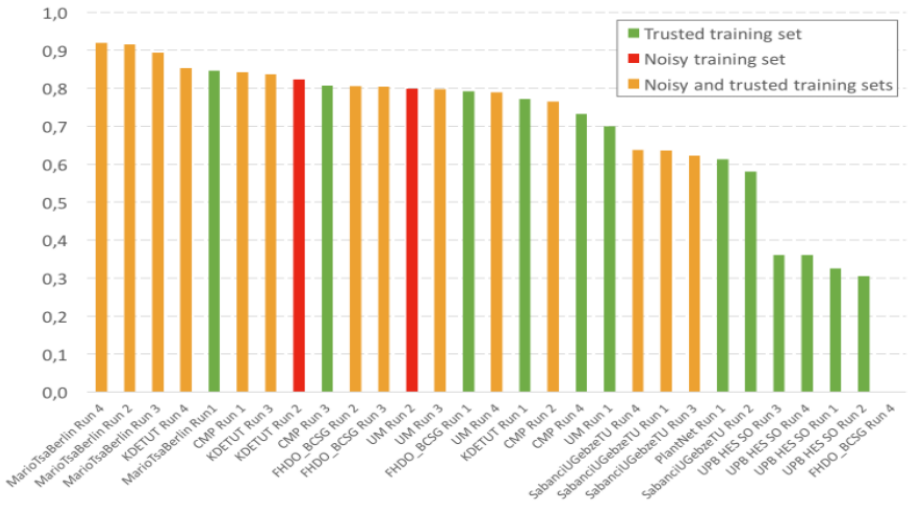

Fig. 3: Results of the LifeClef2017 multi-organ plant classification task

Fig. 3 shows the overall results of the LifeClef2017 multi-organ plant classification task. We observed that Run 2 which is ranked at 12th out of a total of 28 runs is the best among the submitted runs while Run 1 which is ranked at 19th shows the lowest result. Henceforth, we make a deduction that the fusion model HGO-CNN that we currently trained is not generalized enough to predict

Table 2: Performance comparison for trusted training set(EOL)

Method	MRR
CMP Run 3	0.807
FHDO_BCSG Run 1	0.792
KDETUT Run 1	0.772
CMP Run 4	0.733
UM Run 1	**0.700**
PlantNet Run 1	0.613
SabanciUGebzeTU Run 2	0.581
UPB HES SO Run 3	0.361
UPB HES SO Run 4	0.361
UPB HES SO Run 1	0.326
UPB HES SO Run 2	0.305

Table 3: Performance comparison for noisy training set(WEB)

Method	MRR
KDETUT Run 2	0.824
UM Run 2	**0.799**

Table 4: Performance comparison for noisy training set(WEB+EOL)

Method	MRR
MarioTsaBerlin Run 4	0.920
KDETUT Run 4	0.853
KDETUT Run 3	0.837
UM Run 3	**0.798**
UM Run 4	**0.789**
SabanciUGebzeTU Run 4	0.638
SabanciUGebzeTU Run 1	0.636
SabanciUGebzeTU Run 3	0.622

unseen testing images. Run 3 (ranked at 13th) and Run 4 (ranked at 15th), both are the combined results of Run 1 and Run 2 respectively, are ranked lower compared to Run 2. This is clearly due to the poor performance of the Run 1 model. Furthermore, Run 3 is ranked higher than Run 4, an indication that the averaging fusion method performs better than the max voting method.

Next, we compared the results of our submitted runs based on different composition of the dataset. Table 2 shows the results of the trusted training set(EOL). We observe that our UM Run 1 provides a comparable result, where it is ranked at 5th out of a total of 11 runs. We believe that the performance could have been better if the organ layer in the HGO-CNN is trained with the latest Plantclef2017 dataset. However, it was restricted as the organ information is not provided for most of the images. In Table 3, we have the lowest rank but we used only 60% of the noisy WEB dataset. Furthermore, there are only two

participants in this category which is hardly a thorough comparison. Lastly, in Table 4, we observed comparable results for our submitted runs. However, we believe that our performance can be improved. In the current experiments, we separately trained the CNN models using two different datasets and inferred the results using average fusion. It is possible that the performance could have been better if both of the datasets are trained in one single end-to-end CNN model without the prerequisite of external fusion to infer the species. Moreover, training on 100% of noisy images might be able to boost up the classification performance.

5 Conclusions and Future work

This working note explains the implementation of the HGO-CNN for the Plantclef2017 challenge. We described the methodology of our proposed architecture and analyzed the results based on both validation and testing set. We observed that our current HGO-CNN model is not generalized enough to predict unseen testing images. This might due to the lack of robustness at the organ layer trained using the previous PlantClef2015 dataset. In the future, we will revise our proposed model to increase its robustness by re-training all layers using the latest datasets that incorporate both trusted and noisy images.

Acknowledgement

This research is supported by the Postgraduate (PPP) Grant PG007-2016A, from University of Malaya; and the used K40 GPU was donated by NVIDIA Corporation.

References

1. Champ, J., Lorieul, T., Servajean, M., Joly, A.: A comparative study of fine-grained classification methods in the context of the lifeclef plant identification challenge 2015. In: CLEF 2015. vol. 1391 (2015)
2. Choi, S.: Plant identification with deep convolutional neural network: Snumedinfo at lifeclef plant identification task 2015. In: Working notes of CLEF 2015 conference (2015)
3. Goëau, H., Bonnet, P., Joly, A.: Plant identification based on noisy web data: the amazing performance of deep learning (lifeclef 2017). In: CLEF working notes 2017 (2017)
4. Goëau, H., Bonnet, P., Joly, A., Bakic, V., Barthélémy, D., Boujemaa, N., Molino, J.F.: The imageclef 2013 plant identification task. In: CLEF (2013)
5. Haug, S., Michaels, A., Biber, P., Ostermann, J.: Plant classification system for crop/weed discrimination without segmentation. In: Applications of Computer Vision (WACV), 2014 IEEE Winter Conference on. pp. 1142–1149. IEEE (2014)
6. Ioffe, S., Szegedy, C.: Batch normalization: Accelerating deep network training by reducing internal covariate shift. In: Proceedings of the 32nd International Conference on Machine Learning (ICML-15). pp. 448–456. JMLR Workshop and Conference Proceedings (2015)

7. Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., Guadarrama, S., Darrell, T.: Caffe: Convolutional architecture for fast feature embedding. In: Proc. of the ACM International Conference on Multimedia. pp. 675–678 (2014)
8. Joly, A., Goëau, H., Glotin, H., Spampinato, C., Bonnet, P., Vellinga, W.P., Lombardo, J.C., Planqué, R., Palazzo, S., Müller, H.: Lifeclef 2017 lab overview: multimedia species identification challenges. In: CLEF 2017 Proceedings, Springer Lecture Notes in Computer Science (LNCS) (2017)
9. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Advances in neural information processing systems. pp. 1097–1105 (2012)
10. Lee, S.H., Chan, C.S., Mayo, S.J., Remagnino, P.: How deep learning extracts and learns leaf features for plant classification. Pattern Recognition 71, 1–13 (2017)
11. Lee, S.H., Chan, C.S., Remagnino, P.: Hgo-cnn: Hybrid generic-organ convolutional neural network for multi-organ plant classification. In: ICIP (2017)
12. Naresh, Y., Nagendraswamy, H.: Classification of medicinal plants: an approach using modified lbp with symbolic representation. Neurocomputing 173, 1789–1797 (2016)
13. Ojala, T., Pietikainen, M., Maenpaa, T.: Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. TPAMI 24(7), 971–987 (2002)
14. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., et al.: Imagenet large scale visual recognition challenge. International Journal of Computer Vision 115(3), 211–252 (2015)
15. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. CoRR, abs/1409.1556 (2014)
16. Yan, Z., Zhang, H., Piramuthu, R., Jagadeesh, V., DeCoste, D., Di, W., Yu, Y.: Hd-cnn: Hierarchical deep convolutional neural networks for large scale visual recognition. In: Proc. of the IEEE International Conference on Computer Vision. pp. 2740–2748 (2015)
17. Zeiler, M.D., Fergus, R.: Visualizing and understanding convolutional networks. In: Computer vision–ECCV 2014, pp. 818–833. Springer (2014)