Residual Network with Delayed Max Pooling for Very Large Scale Plant Identification

Siang Thye Hang and Masaki Aono

Knowledge Data Engineering and Information Retrieval Laboratory,
Department of Computer Science and Engineering,
Toyohashi University of Technology, Japan
hang@kde.cs.tut.ac.jp, aono@tut.jp

Abstract. In our approach, we applied a few modifications to the 50-layered Residual Network. Our preliminary experiments with the PlantCLEF 2016 dataset showed that the modifications improved classification performance. We have trained three models based on the modified Residual Network configuration with different combinations of trusted and noisy PlantCLEF 2017 datasets. Using confidence scores extracted from the three models, we have submitted four runs and our methods showed competitive classification performance.

Keywords: Plant Identification, Deep Learning, Down-sampling

1 Residual Network with Delayed Max Pooling

We applied a few modifications to the 50-layered Residual Network by He et al. [3], which is also known as ResNet-50.

1.1 Max Pooling Based Down-sampling

In ResNet-50, a total of three convolution operations[1] with stride size 2 but filter size 1×1 is used for down-sampling. As the filter size is smaller than stride size, part of the activations may be ignored in the filtering i.e. convolution processes, as demonstrated in Figure 1.

Therefore, we reduced stride size of the three convolution operations in ResNet-50 from 2 to 1. In addition to this, max pooling of stride size 2 and filter size 2×2 are inserted before these convolution operations. We label this modified configuration as ResNet-50-MP.

1.2 Delayed Down-sampling

Down-sampling is an essential element in Convolutional Neural Network, which reduces number of activations (as well as computational complexity). However,

[1] Namely `res{3,4,5}a_branch2a` based on the ResNet-50 model definition in `github.com/KaimingHe/deep-residual-networks`

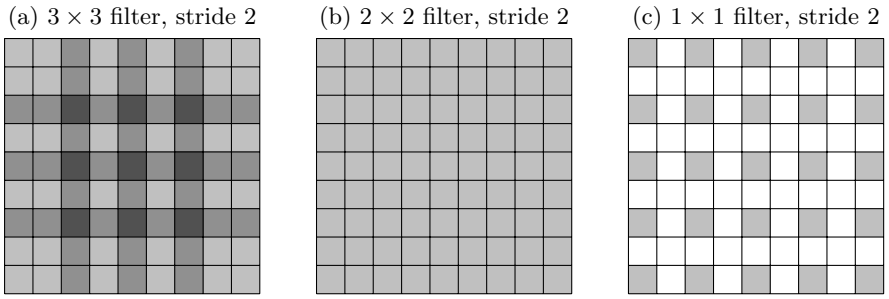

Fig. 1. Coverage of stride 2 based filtering with filter size $3 \times 3, 2 \times 2$, and 1×1. When filter size is larger than 2×2 e.g. (a), some of the regions are overlapped, as shown in the darker regions. When filter size is smaller than 2×2 e.g. (c), some of the regions (in white) are not covered at all.

applying down-sampling too early may leave too little activations for subsequent convolution operations, thus impacting classification performance.

In a paper by He et al. [2], delaying down-sampling shows improved classification performance. In their method, stride size of pooling operations are reduced from 2 to 1, and stride size of their subsequent convolution operations are increased from 1 to 2. In our method, we simply switched the position of the newly introduced max pooling operations (in Subsection 1.1) with their subsequent convolution operations. We label this configuration as ResNet-50-MPD.

2 Network Training and Testing

This section illustrates our implementation for the plant identification task i.e. PlantCLEF. We use the Caffe framework by Jia et al. [5] to implement all of the configurations. Note that **all configurations are trained from scratch** i.e. no pretrained weights are used.

2.1 Input Data

Data Augmentation. Training images are randomly scaled such that shorter sides are in the range of $[224, 336]$. Scaled images are randomly rotated for $\pm 45°$. Rotated images are randomly cropped into 224×224, and finally the cropped images are randomly horizontal flipped.

As for test images, they are scaled such that the shorter sides become 224. The scaled images are then horizontally flipped. Both flipped and non-flipped images are applied into a trained network for prediction. Class-wise outputs (before softmax normalization) of both instances are averaged and then softmax normalized.

Input Normalization. Mean centering of input is already a common procedure to train or test a network. However, this alone may not be sufficient as variance is not considered in this process.

Therefore, we attempted to normalize at higher order by applying Batch Normalization [4] directly onto the (augmented) input. With Batch Normalization, an input is normalized into zero mean and unit variance (and then scaled and shifted accordingly). As the number of filters are quite limited i.e. 64 in the very first convolution layer of ResNet i.e. `conv1`, such normalization should facilitate this convolution layer to learn filters of more varying features.

2.2 Preliminary Experiments with PlantCLEF 2016 Dataset

Dataset Preparation. ResNet-50, ResNet-50-MP and ResNet-50-MPD configurations in Section 1 are trained for 100 epochs with PlantCLEF 2016 training dataset. After each training epoch, each configuration is validated with PlantCLEF 2016 test dataset. Omitting unseen images i.e. of ClassId 9999, a total of 113204 training images and 4510 test images are utilized. Note that data augmentation and normalization as detailed in Section 2.1 are applied to all configurations.

Batch Size. The hardware we use for preliminary experiments is a single NVIDIA's Tesla K40. We use largest possible batch sizes i.e. based on the hardware's memory limitation of 12 GiB.

Learning Schedule. Initial learning rate is 0.1 and is multiplied by 0.1 twice throughout the training process. Training iterations for each learning rate is divided with ratio 4:2:1 across 100 training epochs.

In summary, batch sizes for both original ResNet-50 and ResNet-50-MP are maximized at 31 and they are trained with learning rates 0.1, 0.01, 0.001 for 208671, 104336, 52168 iterations respectively. As for ResNet-50-MPD, the largest possible batch size is 21 and this configuration is trained with the same learning rates for 308038, 154019, 77010 iterations respectively. Validation accuracy of the whole training process for all three configurations is shown in Figure 2.

2.3 Experiments with PlantCLEF 2017 Dataset

Based on the results as shown in Figure 2, we selected the ResNet-50-MPD configuration for this year's plant identification task.

Dataset Preparation. Out of the 256287 trusted training images provided by the task organizers of PlantCLEF 2017 [6][1], we randomly selected around $\frac{1}{10}$ of the images for validation purpose. Specifically, after separating 25063 images for validation, 231224 images remain for 'trusted' training. As for noisy images, out of the provided 1442642 metadata, we managed to obtain 99.0% of them, specifically 1428395 images.

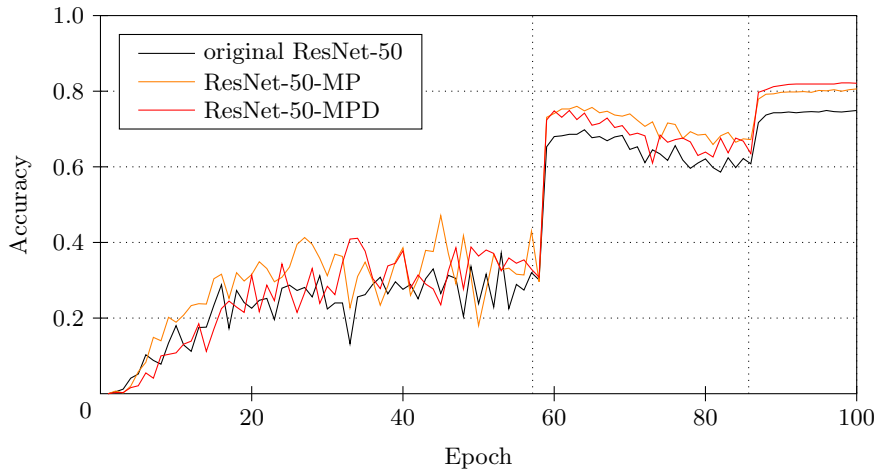

Fig. 2. Validation accuracy of ResNet-50, ResNet-50-MP and ResNet-50-MPD configurations using PlantCLEF 2016 dataset. Base learning rate 0.1 is dropped to 0.01 at around 57^{th} epoch, and is further dropped to 0.001 at around 86^{th} epoch, as indicated by the vertical dotted lines.

Different combinations of trusted and noisy images are used to train the ResNet-50-MPD configuration. Our first model is trained with trusted training images only[2], and our second model is trained with noisy training images only[3], while our third model is trained with both mixed together. We label the first as model T, the second as model N, and the third as model X. All three models are validated with the 25063 validation images.

Batch Size. We use a single NVIDIA's Quadro P6000 to train the three models. With 24 GiB memory, we were able to use batch size up to 47.

Learning Schedule. As the trusted and noisy datasets are a lot larger compared to last year's, we were not able to train for 100 epochs but a fixed amount of training iterations. All three models are trained for 350000 iterations: learning rate 0.1 for 200000 iterations, 0.01 for 100000 iterations and 0.001 for 50000 iterations. In other words, model T was trained for around 71 epochs, model N was trained for around 12 epochs, while model X was trained for around 10 epochs. Validation accuracy of the training processes is summarized in Figure 3.

Run Submission. A total of 25170 test images are provided by the task organizers. As detailed Subsection 2.1, each test image is scaled and then horizontally flipped. Both flipped and non-flipped test images are applied into all

[2] Corresponds to training set E in `imageclef.org/lifeclef/2017/plant`
[3] Corresponds to training set W

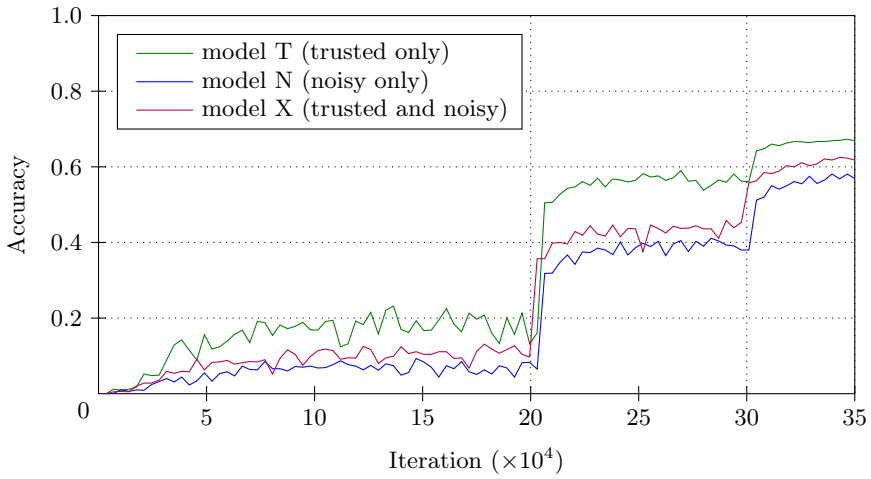

Fig. 3. Validation accuracy of ResNet-50-MPD trained with different combinations of trusted and noisy PlantCLEF 2017 datasets, namely model T, N and X. All of the models are trained for 350000 iterations (but different number of training epochs due to different number of image in each dataset).

three models i.e. model T, N and M. For each model, class-wise average of confidence scores (before softmax normalization) extracted from both flipped and non-flipped images is computed. Additionally, class-wise average of confidence scores with the same ObservationId is also computed. The averaged confidence scores are then softmax normalized. We have submitted four runs with team name KDETUT for this year's plant identification task, as summarized below.

- KDETUT Run 1: Based on model T (trained with trusted images only)
- KDETUT Run 2: Based on model N (trained with noisy images only)
- KDETUT Run 3: Based on model X (trained with trusted and noisy img.)
- KDETUT Run 4: Average of confidence scores based on model T and N

3 Evaluation Result

Mean Reciprocal Rank (MRR) is used as evaluation metric. Evaluation results released by the task organizers are shown in Table 1 and Figure 4. Among the submitted four runs, Run 4, which is average of confidence scores extracted from model T and N, shows the best classification performance.

Table 1. Evaluation results of runs submitted by PlantCLEF 2017 participants, sorted in descending order. Our four submitted runs are highlighted in color.

Run Name	MRR
MarioTsaBerlin Run 4	0.920
MarioTsaBerlin Run 2	0.915
MarioTsaBerlin Run 3	0.894
KDETUT Run 4	0.853
MarioTsaBerlin Run 1	0.847
CMP Run 1	0.843
KDETUT Run 3	0.837
KDETUT Run 2	0.824
CMP Run 3	0.807
FHDO_BCSG Run 2	0.806
FHDO_BCSG Run 3	0.804
UM Run 2	0.799
UM Run 3	0.798
FHDO_BCSG Run 1	0.792
UM Run 4	0.789

Run Name	MRR
KDETUT Run 1	0.772
CMP Run 2	0.765
CMP Run 4	0.733
UM Run 1	0.700
SabanciUGebzeTU Run 4	0.638
SabanciUGebzeTU Run 1	0.636
SabanciUGebzeTU Run 3	0.622
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
FHDO_BCSG Run 4	0.000

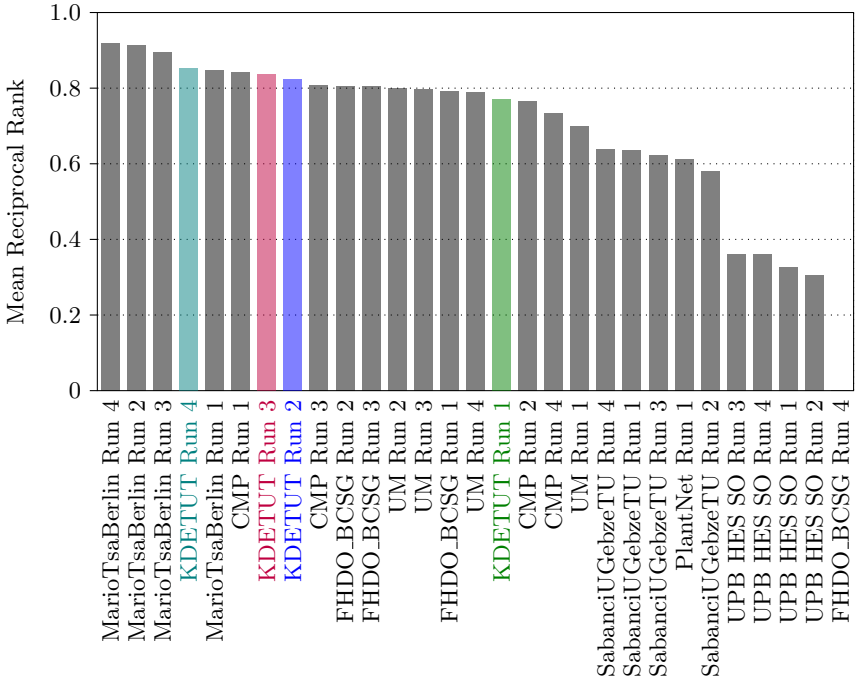

Fig. 4. Visualization of Table 1.

4 Conclusion

In this paper, we described our approach to PlantCLEF 2017, focusing on some modifications to the 50-layered Residual Network. Nevertheless, there are still rooms for improvements in our approaches, as itemized below.

- Especially models trained with very large datasets i.e. model N and X, the fixed amount of 350000 training iterations may not be sufficient. For example, as detailed in Subsection 2.3, in fact model X is only trained for around 10 epochs. However, this setup already requires 4 days to train each model. More training iterations i.e. longer training time may be required to achieve superior classification performance.
- Among the four runs we have submitted, Run 4 yields the highest classification performance. It is based on average of confidence scores computed from model T and N (one each). We believe classification performance can be further improved if the confidence scores are averaged from even more models, for example five model T and five model N. This is however at the cost of multiplied computation time.

References

1. 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)
2. He, K., Sun, J.: Convolutional neural networks at constrained time cost. In: Computer Vision and Pattern Recognition (CVPR), 2015 IEEE Conference on (2015)
3. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Computer Vision and Pattern Recognition (CVPR), 2016 IEEE Conference on (2016)
4. 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 2015, Lille, France, 6-11 July 2015. pp. 448–456 (2015)
5. Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., Guadarrama, S., Darrell, T.: Caffe: Convolutional architecture for fast feature embedding. In: Proceedings of the 22nd ACM international conference on Multimedia. pp. 675–678. ACM (2014)
6. 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