Plant Seedling Classification for better Environment Stewardship

Pei-Fan Liu
NYCU
ID:109550146
ben900926.cs09@nctu.com

Pei-Chen Ho
NYCU
ID:109705004
jennyho0221.c@nycu.edu.tw

Abstract

Plant seedling classification is a competition on Kaggle, which requires to classify unseen seedlings' pictures into one of the twelve given species classes accurately. We intended to apply convolutional neural networks (CNN) and residual networks to perform the task.

1. Introduction

In recent years, image classification has became one of the the core problems in Computer Vision field, for its a large variety of practical applications. Our goal is to accurately differentiate 12 species of crop seedlings from weeds for better stewardship of the environment based on images in a reasonable time frame. There are some obstacles encountered such as the high similarity among each specie and imbalance of the dataset.

Convolutional neural networks (CNN) is one of the most popular models on solving image classification problems. They achieve great performance by the mechanism of local connectivity and weight sharing. However, CNN may not be able to classify correctly when the same object is in different positions, lightings or backgrounds. Thus, we use data augmentation during the training process of our CNN based models. The transfer learning approach is also adopted for more complex CNN-based models like ResNet. The effectiveness of our proposal is measured by the prediction score obtained in Kaggle.

2. Related Work

The first method that came to our mind, is one of the most established algorithm among various deep learning models - convolutional neural network (CNN), a class of artificial neural networks that has been a dominant method in computer vision tasks. CNN can detect the important features automatically without hand-craft feature extractions and develop an internal representation of a two-dimensional image, which allows the model to learn position and scale in variant frames in plant classification. [1] The CNN architecture includes several building blocks, such as convolution layers, pooling layers, and fully connected layers, followed by one or more fully connected layers. [7]. The model we implemented has 6 convolutional networks and 3 fully connected layers.

As for a improvement for former method, the ensemble CNN is introduced. Ensemble methods combine multiple classifiers and have been found to provide the possibility of higher accuracy results than a single classifier [8]. In this classification task, we use one of the most common ensemble method - average voting that generates posterior labels by calculating the average of the softmax class probabilities.

In 2012, a relatively large CNN, Alexnet, achieve the best score performance by far on the competition that involves in classifying 1.2 million high-resolution images. It contains some of the new features : local response normalization (Relu), overlapping pooling, to improve the performance and reduce training time [4]. We adjust to the smaller input size and stride numbers, for the dataset we specified. Using the 5 convolutional networks along with the pooling layer mentioned in the paper.

In 2014, VGG is introduced by Visual Geometry Group from University of Oxford, which reduces parameters by utilizing smaller covolutional layers for faster convergence and reduction in overfitting problem. [6]

In 2015, Deep Residual Learning for Image Recognition is introduced. The residual learning framework ease the training of networks that are substantially deeper than those used previously. [3]

3. Dataset

The database is recorded at Aarhus University Flakkebjerg Research station in a collaboration between University of Southern Denmark and Aarhus University. There are approximately 960 unique plants belong to 12 species at several growth stage. It comprised annotated RGB images with a physical resolutions of roughly 10 pixels of mm [2]. Figure 1 shows sample images belong to each species.

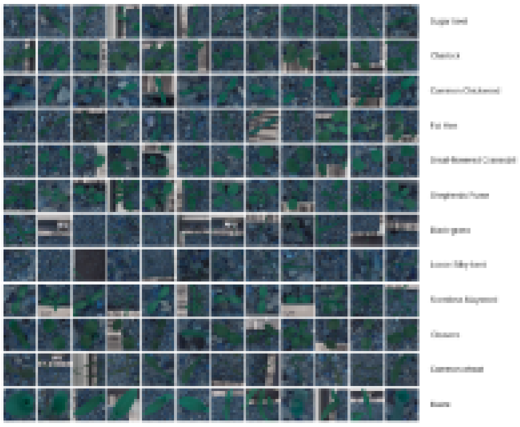

Figure 1. sample images and species

Observing the dataset, we found that there are two main limitations. First of all, the dataset is not large (4750 images), so the training set for each species are limited; Second, the dataset is imbalanced. The following figure shows the distribution of the dataset. There are about 655 training sample for the species "Loose Silky-bent", while only 222 of samples are for the species "Common Wheat".

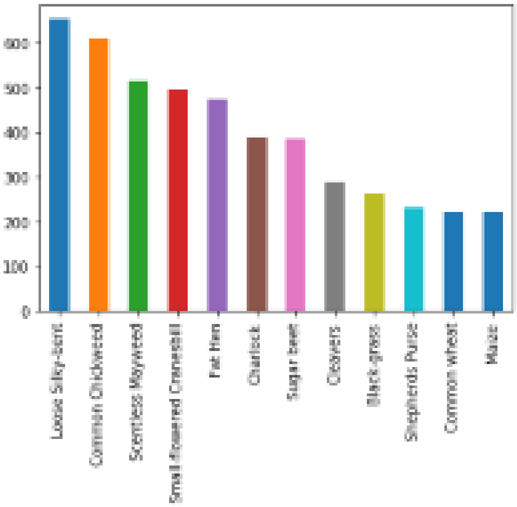

Figure 2. Distribution of data among the species

4. Methodology

4.1. Preprocessing

The training set is randomly divided into two parts, 90% is for training set and 10% is for validation. This step is to prevent the model from overfitting and simulate how the model should work on the unseen data. All images are cropped to the same size (100 X 100 for CNN inputs) and normalized to the range [-1,1].

4.1.1 Label Encoding

The given labels are of string data types, so it's required to transform them into a proper form for easier processing for training. Thus the labels are converted into binary class matrices, where 1 indicated that the species is detected.

4.1.2 Data Augmentation

To build an useful deep learning model, its validation error must decreases with the training error. With the aid of data augmentation, the augmented data minimizes the distance between training and validation set, as well as any testing sets [5]. The techniques we used including rotate, zoom, shift and flip randomly. Below shows a part of the augmented data.

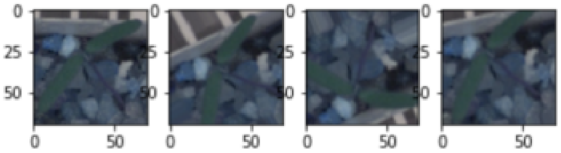

Figure 3. Example of Data Augmentation

4.1.3 Image Segmentation

Since the images were taken in the real world, we need to eliminate noise. Figure 4 shows the process of how we achieve the image segmentation. We first blur the image to remove the noise in the second image. Then, convert the RGB image into HSV, so that it can be easier to separate the color information from the luminance information, which is shown at the third image. Lastly, we create a mask to remove the background of the image and get the seed part for training and testing. The final result of the image segmentation process is shown in the last image.

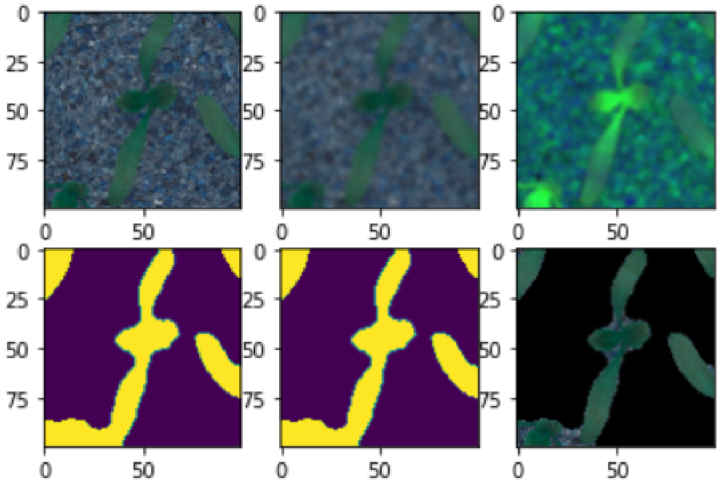

Figure 4. Process of Image Segmentation

4.2. Classification Models

4.2.1 Support-Vector Machine (SVM)

Support-vector machines are supervised learning models with associated learning algorithms that analyze data for classification and regression analysis. We chose it as our baseline.

4.2.2 Customized Convolutional Neural Network (CNN)

Our customized CNN uses 6 convolutional layers and 3 dense layers. Each layer has a batch normalization layer except for the last one. There are also a max pooling layer and a dropout layer follow every two convolutional layers, a dropout layer and a RELU activation layer follow each dense layer. The last dense layer is followed by a softmax layer.

We also designed an model having 3 different CNN ensembled. An ensemble model uses the input layer that is shared between all previous models. In the top layer, the ensemble computes the average of three models' outputs by using the "average" merge layer. The first CNN contains 9 convolutional layers with a RELU activation layer, the first and second three layers are followed by a max pooling layer with a stride size of 2. Except for the last convlotional layer, which is connected to the global average layer. This layer is required for taking the average of these CNNs. Finally followed by a softmax layer. The second CNN is similar to the first one, but with the max pooling layers removed ; and the third model is similar to but having the dropout layers added. There are many kinds of ensemble methods. The one I used is simply taking the average of the above models. Figure 5 shows this idea.

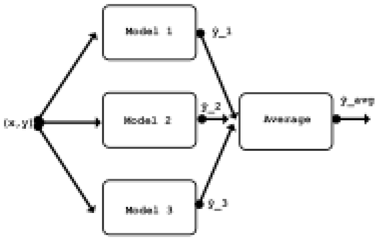

Figure 5. Structure of Ensemble CNN

4.2.3 Alexnet

Alexnet is originally used to complete the classification task of tons of large image (224 X 224) with high resolution. Due to its stunning performance on the Imagenet competition, I tried to modify the input shape and apply this structure on this task with relatively small dataset.

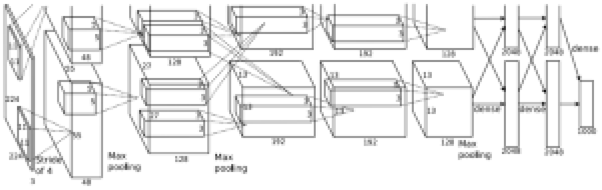

Figure 6. Structure of Alexnet

I chose the input shape of 120 X 120 to fit with the images, the other settings are the same as the original, since I found it result in the most accurate results.

4.2.4 Transfer Learning

Transfer Learning is a research problem in machine learning that focuses on storing knowledge gained while solving one problem and applying it to a different but related problem. In this paper, we use VGG11 and ResNet34 models pretrained on ImageNet dataset. The early convolutional layers of the network are frozen and only the last few layers which make the prediction are trained.

- VGG-11 : VGG-11 is attributes to 11 weighted layers, which takes RGB format 224 x 224 pixel image as input. It has a top-5 accuracy rate at 0.88628 in ImageNet. The number of parameters is 128.8 million.
- ResNet-34 : ResNet-34 is a residual network with 34 layers It has a top-5 accuracy rate at 0.91420 in ImageNet. Residual networks skip the training of few layers using skip connections to prevent vanishing gradient problem. The number of parameters is 21.7 million.

5. Experiments

5.1. Setup

In our project, there are 4750 images in the train set and 794 ones in the test set. We split the training set into 90 percent for training set and 10 percent as the validation set. Our experiment is divided into the following parts : customized CNN, Alexnet, VGG11 and ResNet34 .We trained these models on the Google Colab with the provided GPU supported. We use the accuracy as an assessment of the evaluation result. After that, we predict the test set and upload the output file to the Kaggle competition site for the final evaluation.

- Customized CNN and Alexnet

I use the data augmentation technique to prevent overfitting during the fitting process. That is, randomly flips, rotates and shifts the image by 0.1 scale. When training the model, the learning rate will be reduced by 0.4 if the accuracy does not improve within six episodes, and has a minimum of 1e-5. The purpose here is to not let the model converges too quickly. I trained the model for 80 episodes with batch size of 32. The smaller batch size has the advantage of better generalization. The weights of the model will be stored for the future use if there’s improvement.

- Transfer Learning

In order to increase diversity of data available for training models, data augmentation was used. The augmentation including random rotation, random crop, random horizontal flip, random application of color jitter and Gaussian blur and random affine under a probability of 0.9. All the training and validation images are normalized using mean = [0.485, 0.456, 0.406] and std = [0.229, 0.224, 0.225]. Normalization helps get data within a range and reduces the skewness which helps learn faster and better. The loss function is cross entropy loss, optimizer is Adam, batch size is 64, learning rate is 1e-4 and the number of training epochs is 50.

5.2. Results

Below table shows the model and their training, testing accuracy and the score obtained on the Kaggle competition site.

Model	Train Accuracy	Test Accuracy	Score
SVM	0.51366	0.40756	0.41183
CNN	0.9453	0.9221	0.92821
ensemble	0.9488	0.9200	0.93198
Alexnet	0.9411	0.9326	0.94206
VGG11	0.9909	0.9545	0.94962
ResNet-34	0.9953	0.9627	0.95214

Table 1. Classification Result of Different Models

Below figure shows the confusion matrix of Resnet-34, which has the best performance in this task.

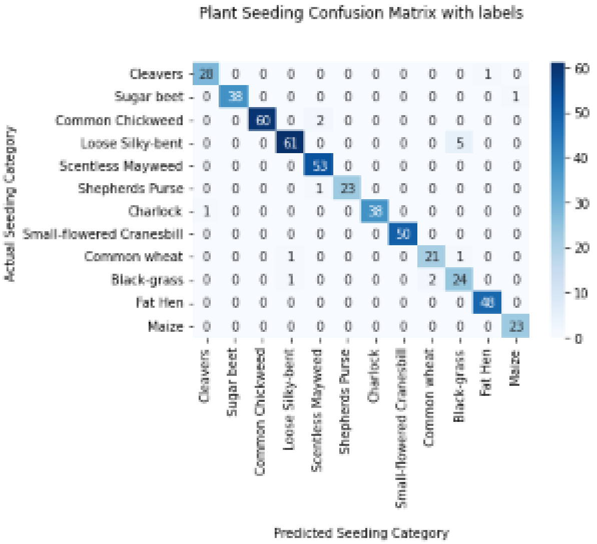

Figure 6. Confusion Matrix of ResNet-34

5.3. Discussion

The results of CNN and the ensemble model are quite similar. The initial motivation for applying the ensemble model is to fix the inaccuracy due to the misclassification between the ”Loose Silky-bent” and ”Black Grass” classes. I tried to implement three CNN as ”weak classifier”, see if they can classify what a single strong CNN fail to do. The result turns out to be negative to this prediction. The problem might be solved by choosing an specified classifier only for classifying between these two classes.

As for the Alexnet, it was originally used to classify large-scale images. We can still see the improvement compared to CNN, but the improvement scale is not quite significant. The reason behind this issue probably is that the overlapping pooling is able to keep the important feature and repetitively observe them while training.

ResNet-34 has more layer compared to VGG and CNN, but the performance is quite similar with each other. This may indicate that plant seedling classification task, which involves in classify the relatively smaller dataset, is essentially quite different from the general image classification task in requiring more detailed and subtle information from the input image. So models worked well for ImageNet dataset could not extract useful features for this task.

ResNet and VGG was trained for about an hour, but with only 50 episodes, it achieve the better result. CNN was trained for at least 15 minutes, which is a shorter time consumed, but the performance is not that well even with 80 episodes.

From the confusion matrix, we can see that almost every species can be classified well, except for the classes ”Loose Silky-bent” and ”Black Grass”. Figure 7 shows the similarity of these two classes, thus leads to relatively higher misprediction rate than other species.

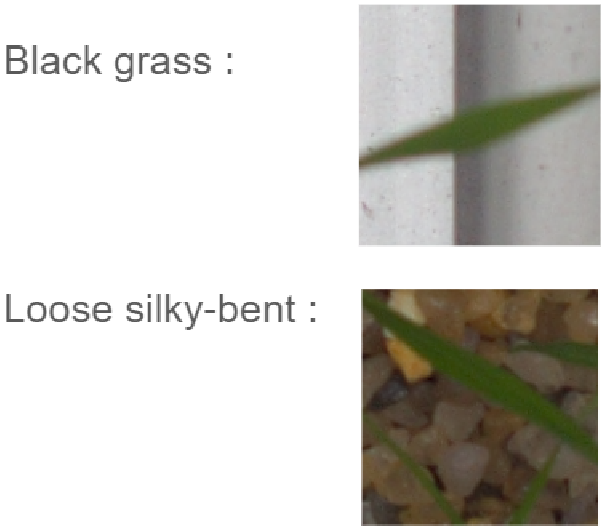

Figure 7. Comparison between ”Loose Silky-bent”

and ”Black Grass”

6. Future works

There are still no absolute instructions about how to improve the accuracy of the CNN model. The factors affecting the accuracy include various parameters (for example, the scale of data augmentation and the batch size), and the overall structure of the model, such as how many convolutional and max pooling layers should be used. I cannot guarantee that this is the best of what CNN can achieve without lots of trials and errors, so this could be one of the future works.

To balance the dataset, we can try the methods such as Synthetic Minority Over-sampling TEchnique (SMOTE). Comparing with random oversampling method, SMOTE method can effectively avoid the problem of overfitting of classifiers [9].

7. Contributions

Both of us contribute a lot to this project. Pei-fan Liu did the part of CNN and Alexnet, and Pei-Chen Ho was responsible for the implementation of the transfer models and SVM. We work together on the report and the literature researches.

Github link : https://github.com/ben900926/Plant-seedling-classification

References

[1] Jason Brownlee. When to use mlp, cnn, and rnn neural networks. machine learning mastery. *https://machinelearningmastery.com/when-to-use-mlp-cnn-and-rnn-neural-networks/*, 2018. 1

[2] Thomas Mosgaard Giselsson, Rasmus Nyholm Jørgensen, Peter Kryger Jensen, Mads Dyrmann, and Henrik Skov Midtiby. A public image database for benchmark of plant seedling classification algorithms. *https://arxiv.org/abs/1711.05458*, 2017. 1

[3] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. pages 770–778, 2016. 1

[4] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. *https://proceedings.neurips.cc/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf*, 2012. 1

[5] C. Shorten, Khoshgoftaar, and T.M.y. A survey on image data augmentation for deep learning. *J Big Data 6, 60 (2019). https://doi.org/10.1186/s40537-019-0197-0*, 2019. 2

[6] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 1

[7] Yamashita, R., Nishio, M., Do, and R.K.G. et al. Convolutional neural networks: an overview and application in radiology. *Insights Imaging 9, 611–629 (2018). https://doi.org/10.1007/s13244-018-0639-9*, 2018. 1

[8] Ali Yazdizadeh, Zachary Patterson, and Bilal Farooq. Ensemble convolutional neural networks for mode inference in smartphone travel survey. *https://arxiv.org/ftp/arxiv/papers/1904/1904.08933.pdf*, 2019. 1

[9] Zhuoyuan Zheng and Ye Li Yunpeng Cai. Oversampling method for imbalanced classification. *Computing and Informatics*, 2015. 5