Plant Leaf Recognition

Albert Liu, albertpl@stanford.edu Yangming Huang, yangming@standford

I. Introduction

Automatic plant species recognition with image processing is a fine-grained image recognition problem and generally considered a hard problem due to the subtle differences between different species in the same class. Sometime such fine differences can be even challenging to humans. Typically this requires large training data but it is not feasible due to the number of planets (over 220000 [2]).

1) The subtle differences between different species in the same class. Sometime such fine differences can be even challenging to humans.
2) Large number of categories, over 220000 [2].

The main application are weeds identification, species discovery, plant taxonomy, natural reserve park management and so on [3].

In this report, we describe our exploration with this problem, using traditional handcrafted features and features extracted from pretrained deep convolution neural network (ConvNets).

The rest of the report is organized as follows. In section III, we describe the data set. Section.

II. Related Work

Research on automatic leaf classification from image has been active since 2000. Lots of hand-crafted features have since proposed, ranging from shape based, to statistical texture and margin [2] [3] [1]. Also generic computer vision object recognition/detection features, such SIFT and HOG are studied for this problem. [TODO] Most of such manually engineered features achieve excellent accuracy on clean images taken in controlled conditions, which consist of one single well aligned leave on contrasting background, such as those images in data set [TODO]. Recently, with the huge success of ConvNets, particularly in annual ImageNet Large Scale Visual Recognition Challenge [11], researchers start to apply ConvNets to this problem [TODO]. [TODO] have suggested that generic features can be extracted from large ConvNet and yield very good results on fine-grained classification problems even without fine-tuning the pretrained model.

III. DataSet

We found two types of data set

1) Clean images, which consists of well aligned leaf on single contrasting background, with little or no variations of luminance or color.

Name	Species	Samples Per Species
Swedish [15]	15	75
Flavia [16]	33	$\sim$ 60
UCI [14]:	40	5 16
Kaggle [3]:	99	16

2) ImageCLEF [17], which is collected through crowd sourced application, has 250 species and 26077 images. This is a much more challenging datasets with variations on lighting conditions, viewpoints, background clutters and even occlusions. The data further split to different subset: uniform, which is relatively with less clutter, and natrual-background, which is taken in a total natural enviroment. To effectively evaluate our approaches, we use subsets of this dataset.

Name	Species	train samples	test samples
uniform	66	[TODO:]	1194
natural	57	2585	521

IV. Approach

A. Overview

Images captured in the field, such as those from ImageCLEF [17], are more challenging as there are much more intra class variations due to viewpoint, lighting condition and occlusion. We break down the problem into two pieces. Firstly, we want to locate the leaf in the image which is an single object localization problem. This step becomes necessary if the image have multiple objects and possibly leaf is not the most salient object. Then, we predict the species of the leaf, which is a fine-grained classification problem. This is the main task and for most data set, we only need to perform this task. Here we explore both traditional method, such as Bag of Features, and ConvNets based method.

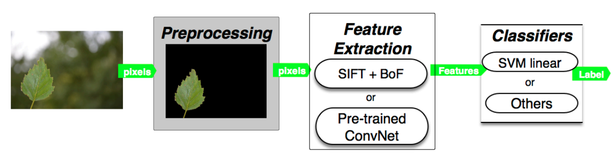

Fig. 1. Overview of the system

Firstly, during preprocessing, we apply Contrast Limited Adaptive Histogram Equalization to reduce lighting condition variation and raw images are resized to fit to the next layer. In traditional approach, we also use K-means to remove background color heuristically. For challenging dataset, we find the convex hull containing the largest N contours and then use GrabCut to segment leaf out of the background clutter. Next we extract features and we explore two options,

1) ConvNets. We take transfer learning approach, in order to make use of the power of ConvNets with the constrains of time and computations. Specifically, we take a couple of ConvNets that are pretrained with ImageNet for ILSVRC object image classification task, remove one or two top

layers and then treat the rest of the ConvNet as fixed feature extractor for the our data set.

2) SIFT + Bag of Features Key points are densely sampled and SIFT feature descriptors are retrieved from each raw images. Then at last, we train a classifier, using the feature vectors from the above step and predict labels for our test data.

Part of reason

[Explain the data set for our pretrained model and why this is this is applicable to our problem, i.e. same feature space.]

[Explains architecture of ResNet50]

The CNN codes (feature vector) we get from ResNet50 is 2048 dimensions. We have two settings.

– [explain preprocessing] The feature vector is normalized with unit variance and we learn a linear SVM classifier [(or others)] from the features.
– We further augment the training set by adding rotated and shifted copies of samples.

[Explain if and how this method is suitable for our task] - The CNN codes are trained for the object image classification task of ILSVRC (2013 for VGGNet and 2015 for ResNet), which is aligned with our task except that ours are more fine-grained classification. -Some of our data set [ImageCLEF2013] is more challenging in term of clutter and occlusion.

B. *ConvNets approach*

C. *exploration*

We started by training CNN classifier from scratch, following the guidelines below:

1) Convolutional layer learning features from general to specific, giving more layers helps with the transition.
2) Dropout reduces overfitting
3) Use aggressive pooling to reduce the dimensionality.

We designed our CNN with the architecture below:

$$(Conv - ReLU - MaxPool)x3 - FC - DropOut - SoftMax \quad (1)$$

It works well with [15], with an error rate of [], But when test against [17], the accuracy reduced significantly. There are a few things to consider:

- The fore-mentioned lighting variation and background clutter.
- More species with less samples per classes.

[More noise need more variations and more aggressive filtering thus deeper network is required]

Research shows number of samples need to be big enough to avoid excessive overfitting. See B section of the supplementary material of [21]. Note that the research is proven for AlexNet. While we don't have enough data to seek for the threshold which suffices to avoid overfitting for ResNet50, we assume 20 per samples is still not enough.

[explain why species with less sample will reduce the performance. Explain why sample per classes makes it harder to predict Learning Theory] [TODO]

[TODO: learning curve]

A deeper CNN with better architecture is needed for the complicity of our data. Thus we switched to pre-trained weight of proven CNN architecture.

Jason et al. found that even features transferred from distant tasks are better than random. weights[21]

Starting from VGG16, [TODO: reference to VGG16] [TODO: why ResNet50 is better than VGG16]

With ResNet50

D. *process*

generalization vs specialization,
remove top layer, predict with upper layer
extract features
final output

E. *Discussion*

How do you decide what type of transfer learning you should perform on a new dataset? This is a function of several factors, but the two most important ones are the size of the new dataset (small or big), and its similarity to the original dataset (e.g. ImageNet-like in terms of the content of images and the classes, or very different, such as microscope images). Keeping in mind that ConvNet features are more generic in early layers and more original-dataset-specific in later layers, here are some common rules of thumb for navigating the 4 major scenarios: [19]

1) similarity (transfer learning is good)
2) size for each class(fine tuning is not a good idea)

F. *Bag of Features (BoF) + local feature descriptors*

Due to the simplicity and performance, this well established approach was taken at first. Interest points are detected from the raw images and then local invariant feature descriptors are collected, which are clustered to form the visual vocabulary/codebook. Afterwards, each raw image can be represented with histograms of visual words, i.e. term vectors. We have prototyped a BoF system, based on the OpenCV package. Here is the illustration of the system 2.

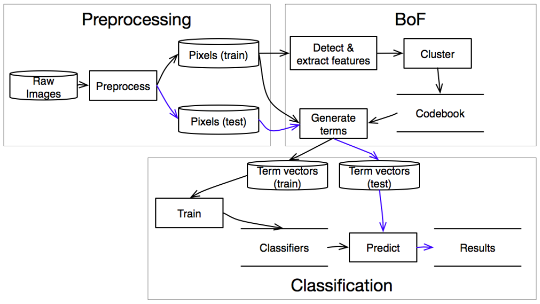

Fig. 2. System Design for BoF

During preprocessing, raw images are converted to gray-scaled images and resized to reduce computation complexity. We extract SIFT descriptors from the pixels after detecting the key points. Limiting the width of image to 128 pixels, we have roughly 200 SIFT descriptors per image. Then all the descriptors are clustered to build visual words via K-Means. Due to computation complexity, we pick randomly 100 training images to build the visual vocabulary for the initial run.

V. Experimental Results

[Explain cross validation approach we used]

We use prediction rank-1 identification (i.e. accuracy) as our performance metric, which is defined as $Accurary = \frac{N_c}{N_t} \times 100\%$, where N_c represents the number of correct match and N_t is the total number of test samples.

[Test results as table.]

VI. Discussions

VII. Conclusion and Future work

References

[1] S. Cho, D. Lee, and J. Jeong. Automation and emerging technologies: Weedplant discrimination by machine vision and artificial neural network. Biosystems Engineering, 83(3):275280, 2002.

[2] Charles Mallah, James Cope, James Orwell. Plant Leaf Classification Using Probabilistic Integration of Shape, Texture and Margin Features. Signal Processing, Pattern Recognition and Applications, in press. 2013

[3] Pedro F. B. Silva, Andre R.S. Marcal, Rubim M. Almeida da Silva. Evaluation of Features for Leaf Discrimination. 2013. Springer Lecture Notes in Computer Science, Vol. 7950, 197-204.

[4] Itheri Yahiaoui, Nicolas Herve, and Nozha Boujemaa. Shape-based image retrieval in botanical collections, Lecture Notes in Computer Science including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinfor matics, vol. 4261 LNCS, pp 357-364, 2006.

[5] Jassmann TJ, Tashakkori R, Parry RM (2015) Leaf classification utilizing a convolutional neural network. In: SoutheastCon

[6] Neeraj Kumar, Peter N Belhumeur, Arijit Biswas, David W Jacobs, W John Kress, Ida C Lopez, and Joao VB Soares, Leafsnap: A computer vision system for automatic plant species identification, in ECCV, pp. 502516. Springer, 2012.

[7] DavidHall,ChrisMcCool,FerasDayoub,NikoSunderhauf, and Ben Upcroft, Evaluation of features for leaf classification in challenging conditions, 2015.

[8] Monica G Larese, Ariel E Baya, Roque M Craviotto, Miriam R Arango, Carina Gallo, and Pablo M Granitto, Multiscale recognition of legume varieties based on leaf venation images, Expert Systems with Applications, vol. 41, no. 10, pp. 46384647, 2014.

[9] Hasim A, Herdiyeni Y, Douady S (2016) Leaf shape recognition using centroid contour distance. In: IOP conference series: earth and environmental science, p 012002

[10] Hall, David, McCool, Chris, Dayoub, Feras, Sunderhauf, Niko, & Upcroft, Ben (2015), Evaluation of features for leaf classification in challenging conditions. In IEEE Winter Conference on Applications of Computer Vision (WACV 2015), 6-9 January 2015, Big Island, Hawaii, USA.

[11] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C. Burges, L. Bottou, and K. Weinberger, editors, Advances in Neural Information Processing Systems 25, pages 10971105. Curran Associates, Inc., 2012.

[12] Y. Jia. Caffe: An open source convolutional architecture for fast feature embedding. http://caffe.berkeleyvision.org, 2013.

[13] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei. Imagenet large scale visual recognition challenge. arXiv preprint arXiv:1409.0575, 2014.

[14] https://archive.ics.uci.edu/ml/datasets/Leaf

[15] Oskar J. O. Sderkvist. Computer vision classifcation of leaves from swedish trees. Master's Thesis, Linkoping University, 2001.

[16] Stephen Gang Wu, Forrest Sheng Bao, Eric You Xu, Yu-Xuan Wang, Yi-Fan Chang and Chiao-Liang Shiang, A Leaf Recognition Algorithm for Plant classification Using Probabilistic Neural Network, IEEE 7th International Symposium on Signal Processing and Information Technology, Dec. 2007, Cario, Egypt

[17] http://www.imageclef.org/2013/plant

[18] Chollet, François, 2015, https://github.com/fchollet/keras,

[19] http://cs231n.github.io/transfer-learning/

[20] Ali Sharif Razavian Hossein Azizpour Josephine Sullivan Stefan Carlsson. CNN Features off-the-shelf: an Astounding Baseline for Recognition, CVAP, KTH (Royal Institute of Technology) Stockholm, Sweden

[21] Jason Yosinski,1 Jeff Clune,2 Yoshua Bengio,3 and Hod Lipson4. How transferable are features in deep neural networks? arXiv:1411.1792, 2014.