NLab-UTokyo at ImageCLEF 2013 Plant Identification Task

Hideki Nakayama

Grad. School of Information Science and Technology, The University of Tokyo,
7-3-1, Hongo, Bunkyo-ku, Tokyo, JAPAN
nakayama@ci.i.u-tokyo.ac.jp
http://www.nlab.ci.i.u-tokyo.ac.jp/

Abstract. We describe our system at the ImageCLEF 2013 plant identification task. Plant identification is extremely challenging because target classes are often visually quite similar. To distinguish them, we need to extract highly informative visual features. We believe that the key to achieving this is to enhance the discriminative power of local descriptors. We employed multiple local features with our polynomial embedding technique to boost the performance. Further, they were encoded into the sophisticated Fisher Vector representation which enables accurate classification with linear classifiers. Our system achieved promising performance, and got the first place in NaturalBackground and the third place in SheetAsBackground tasks, respectively.

Keywords: Fine-grained Visual Categorization, Multiple Local Descriptors, Polynomial Embedding, Fisher Vector

1 Introduction

In this report, we describe our contributions submitted to the ImageCLEF 2013 plant identification task [3, 8]. The system is based on our recently proposed method designed for fine-grained visual categorization (FGVC) [13]. The goal of FGVC is to categorize conceptually (and thus visually) similar classes such as plant and animal species [6, 18, 14, 10], and thus naturally includes the concept of this challenge. However, FGVC is regarded to be extremely difficult because of its high intra-class and low inter-class variations [6].

To distinguish very similar categories, we need to extract highly informative visual features. We believe that the key to achieving this is to enhance the discriminative power of local descriptors. Our method can efficiently improve the discriminative performance of arbitrary local descriptors for bag-of-words [5] based systems with a simple supervised dimensionality reduction method. Using polynomials of a descriptor and its neighbors, we can efficiently exploit local spatial co-occurrence patterns.

We implemented our method with standard object recognition pipelines using the state-of-the-art Fisher Vector coding [15]. Our submitted runs achieved the first place in NaturalBackground and the third place in SheetAsBackground

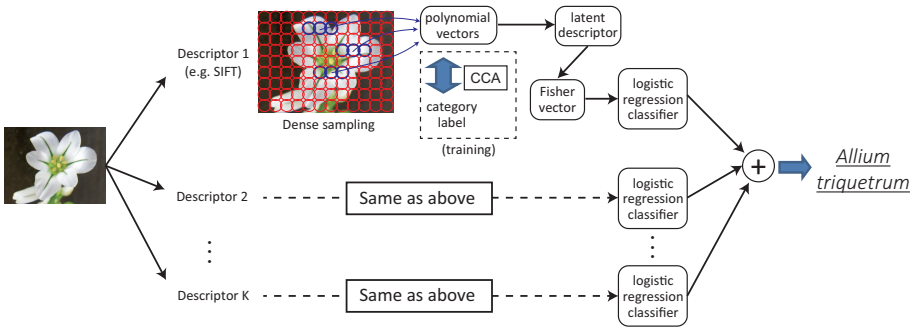

Fig. 1. Overview of our system.

tasks, respectively. Moreover, we achieved the first place in four of five sub-categories in NaturalBackground task.

2 Our Approach

2.1 Overview

Figure 1 illustrates the entire pipeline of our system. We first extract multiple local descriptors from images. For each type of descriptors, we first compute augmented latent descriptors in a supervised learning framework [13], which is our core contribution. Then we encode them into a global feature vector using the Fisher Vector [15] framework. As a classifier, we train a linear logistic regression model. Classifiers are trained independently for each descriptor and combined in late-fusion approach. The output of logistic regression is a probability and can be easily integrated when fusing multiple classifiers. Specifically, we simply take the average log-likelihood of posterior probability for each classifier.

2.2 Polynomial Embedding

Augmenting Descriptors
We densely extract local features $\boldsymbol{v} \in R^d$ from images. Each patch at position (x, y) is described by $\boldsymbol{v}_{(x,y)}$. We augment this by explicitly including the polynomials [1] of its elements. Let $\boldsymbol{p}^c_{(x,y)}$ denote the augmented descriptor, where c is the number of neighbors considered. When no neighbor is considered,

$$\boldsymbol{p}^0_{(x,y)} = \begin{pmatrix} \boldsymbol{v}_{(x,y)} \\ upperVec\left(\boldsymbol{v}_{(x,y)}\boldsymbol{v}^T_{(x,y)}\right) \end{pmatrix}, \tag{1}$$

[1] We use at most the second-order polynomials in this paper considering the computational cost, although our framework supports higher-order ones.

where, $upperVec()$ is the flattened vector of the components in the upper triangular part of a symmetric matrix.

Moreover, we can efficiently exploit local spatial information by taking the polynomials between neighboring descriptors. When considering two neighbors (left side and right side),

$$\boldsymbol{p}^2_{(x,y)} = \begin{pmatrix} \boldsymbol{v}_{(x,y)} \\ upperVec\left(\boldsymbol{v}_{(x,y)}\boldsymbol{v}^T_{(x,y)}\right) \\ Vec\left(\boldsymbol{v}_{(x,y)}\boldsymbol{v}^T_{(x-\delta,y)}\right) \\ Vec\left(\boldsymbol{v}_{(x,y)}\boldsymbol{v}^T_{(x+\delta,y)}\right) \end{pmatrix}, \tag{2}$$

where, $Vec()$ is the flattened vector of the components of a matrix, and δ is an offset parameter for defining neighbors.

In this work, we considered at most two neighbors, although our previous work suggests that using more neighbors could have resulted in better performance [13].

Supervised Dimensionality Reduction

We apply canonical correlation analysis (CCA) [9] to the pairs of the augmented descriptor $\boldsymbol{p}$ and corresponding label vector $\boldsymbol{l}$. In this work, we use the image-level label vector [2] for descriptor compression. That is, all $\boldsymbol{p}$ within an image are coupled with the same label vector for supervised dimensionality reduction [3].

CCA finds the linear projections $\boldsymbol{s} = A^T\boldsymbol{p}$ and $\boldsymbol{t} = B^T\boldsymbol{l}$ that maximize the correlation between the projected vectors $\boldsymbol{s}$ and $\boldsymbol{t}$. We randomly sample $\{\boldsymbol{p}^{(x,y)}, \boldsymbol{l}^{(x,y)}\}$ pairs from the entire training dataset, and let $C = \begin{pmatrix} C_{pp} & C_{pl} \\ C_{lp} & C_{ll} \end{pmatrix}$ denote their covariance matrices. Namely,

$$C_{pp} = \frac{1}{N}\sum(\boldsymbol{p}-\bar{\boldsymbol{p}})(\boldsymbol{p}-\bar{\boldsymbol{p}})^T, \tag{3}$$

$$C_{ll} = \frac{1}{N}\sum(\boldsymbol{l}-\bar{\boldsymbol{l}})(\boldsymbol{l}-\bar{\boldsymbol{l}})^T, \tag{4}$$

$$C_{pl} = \frac{1}{N}\sum(\boldsymbol{p}-\bar{\boldsymbol{p}})(\boldsymbol{l}-\bar{\boldsymbol{l}})^T, \tag{5}$$

$$C_{lp} = C^T_{pl}, \tag{6}$$

[2] The dimension of the vector is the number of categories. If the image belongs to the category w_i, the i-th element is one; otherwise, it is zero.

[3] Obviously, this is a rather rough approach, since not all local features within an image are actually related to the image-level labels. Nevertheless, we note that this assumption is justified somewhat for FGVC problems, since objects are often closely targeted by users.

where, N is the number of sampled pairs, and $\bar{\boldsymbol{p}}$ and $\bar{\boldsymbol{l}}$ are their means. The solution of CCA can be obtained by solving the following eigenvalue problem.

$$C_{pl}C_{ll}^{-1}C_{lp}A = C_{pp}A\Lambda^2 \quad (A^T C_{pp} A = I_m), \tag{7}$$
$$C_{lp}C_{pp}^{-1}C_{pl}B = C_{ll}B\Lambda^2 \quad (B^T C_{ll} B = I_m), \tag{8}$$

where Λ is the diagonal matrix of the first m canonical correlations, and m is the dimension of the canonical elements. The parameter m corresponds to the dimension of the embedded descriptor, and needs to be tuned manually. One problem is that m can be at most the dimension of the label vector because of the rank problem. If we need more features, we can project $\boldsymbol{p}$ into the orthogonal subspace and iteratively apply CCA to further extract discriminative components.

Using the projections obtained by CCA, we get a compact vector $\boldsymbol{s}$ that embeds a high-dimensional augmented vector, which we call the latent descriptor.

$$\boldsymbol{s} = A^T \boldsymbol{p}. \tag{9}$$

Once the latent descriptor is computed, it can be used in the exact same manner as widely-used raw descriptors such as SIFT.

Global Feature Vector
We encode the latent descriptors into a global feature vector using the Fisher Vector framework [15], which is a recently established state-of-the-art variant of bag-of-words encoding. Since the dimensionality of Fisher Vector is in proportional to that of local descriptors, compactness of the latent descriptor is essentially important to utilize this representation.

3 Plant Identification Task

The goal of the challenge is to identify 250 species of plants from their photos. There are two main subtasks: SheetAsBackground and NaturalBackground (Fig. 2). While the objective of the former is to recognize leaves spread on white background, the latter targets more organs and generic background. Therefore, NaturalBackground task has more generic nature like typical FGVC problems and thought to be challenging. The performance is evaluated in terms of the rank of the correct species in the list of retrieved species. The score is normalized by the numbers of content owners, individual plants, and pictures taken from the same plant. For more information, refer to [8].

3.1 Details of the system

We used several standard local descriptors in our system, such as SIFT [12], C-SIFT [2], Opponent-SIFT [16], HSV-SIFT [1], and the self-similarity (SSIM) [17] descriptors. The dimension of SSIM is 40 in our system (4 radial bins and 10

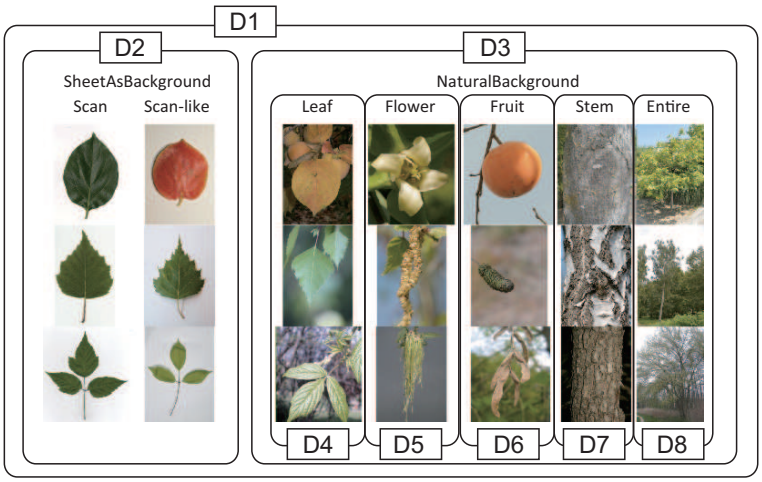

Fig. 2. Domains for classifiers in different runs. For each domain, we train a classification system independently (See text).

Table 1. Number of samples for each domain. 'trainval' corresponds to the originally provided training examples.

	D1 (All)	D2 (SAB)	D3 (NB)	D4 (Leaf)	D5 (Flower)	D6 (Fruit)	D7 (Stem)	D8 (Entire)
training	18731	8072	10015	3023	2952	1090	1076	1150
validation	2254	1709	1189	480	570	297	261	305
trainval	20985	9781	11204	3503	3522	1387	1337	1455
test	5092	1250	3842	790	1233	520	605	694

angle bins). All these local features are extracted in a dense sampling approach without rotation invariance [11]. We extract local features from 24x24 patches on regular grids spacing five pixels. They are compressed into 64 dimensions via PCA, except for SSIM. Finally, we apply our polynomial embedding (PE) method with CCA and obtain 64-dimensional latent descriptor ($m = 64$). We fix the offset parameter $\delta = 20$ for defining neighbors. For implementing Fisher Vectors, we use 64 Gaussians for estimating a Gaussian mixture model and concatenate feature vectors from an entire image and three horizontal regions.

We used the feature extraction software provided by the authors of [16] and [4] for computing SIFT (including its variants) and SSIM, respectively. Also, we used the LIBLINEAR [7] package for the implementation of our classifiers.

3.2 Our Runs

We submitted three runs under different configurations. Each run consists of some classifiers independently trained for a certain domain as in Fig. 2. Information for identifying the domain of testing samples can be drawn from the corresponding xml files [4].

- **Run 1 (D1)**: A multi-class classifier is trained on the whole dataset of 250 classes without distinguishing SheetAsBackground and NaturalBackground categories (and its sub-categories).
- **Run 2 (D2, D3)**: Two classifiers are trained independently for SheetAsBackground and NaturalBackground categories, respectively. We do not distinguish their sub-categories.
- **Run 3 (D2, D4-8)**: Classifiers are trained independently for SheetAsBackground and NaturalBackground categories. The former is the same one used in Run 2 (D2). For the latter, we train classifiers independently for each of five sub-categories (D4-8).

To tune our system, we take roughly 10% of the individual plants in the provided training dataset for validation. Table 1 summarizes the number of samples for each domain. For simplicity, we evaluate the classification accuracy on the validation dataset without distinguishing individual plants and owners. After optimizing the parameters, classifiers are trained again on the original training dataset and applied to testing data.

3.3 Validation Results

For various domain and feature combinations, we tuned our system and validated their effectiveness. Table 2 shows the results. "PCA64" denotes the Fisher Vector using 64-dimensional descriptors compressed via PCA [5]. This is a typical implementation of Fisher Vector coding and serves as the baseline. For most of the trials, PE reasonably improves the performance of the original descriptors. Also, the relative improvement seems more significant in NaturalBackground domains.

Based on the results, we selected the features for final submissions. In SheetAsBackground (D2) domain, we used only gray SIFT because we found color descriptors were not effective. As for NaturalBackground domain, we chose color SIFTs + SSIM combination considering its good performance in all subcategories.

3.4 Test Results

Based on the validation results, we submitted three runs. Figure 3 summarizes the performance of submitted runs from all participants. Not surprisingly, Run

[4] Note that this is not interpreted as a manual intervention in this challenge.
[5] We use the raw SSIM descriptor without applying PCA.

Table 2. Classification performance on the validation dataset (%). Checkmarks indicate that the classifiers based on the corresponding descriptors are integrated.

	SIFT	C-SIFT	Opp.-SIFT	HSV-SIFT	SSIM	PCA64	Ours	Diff.	Submit
D1	✓					35.8	36.8	0.9	
(All)		✓				33.9	36.4	2.5	
			✓			32.1	34.0	1.9	
				✓		33.4	34.1	0.7	
	✓	✓	✓	✓		38.2	**38.8**	0.6	✓
D2 (Sheet As Background)	✓					50.8	**52.5**	1.7	✓
D3	✓					9.0	11.0	2.0	
(Natural		✓				15.6	15.3	-0.2	
Background)			✓			13.0	13.3	0.3	
				✓		9.8	14.1	4.3	
					✓	9.7	10.5	0.8	
		✓	✓	✓	✓	15.9	**17.8**	1.9	✓
	✓	✓	✓	✓	✓	15.0	**17.9**	2.9	
D4	✓					12.9	14.6	1.7	
(Leaf)		✓				12.1	13.1	1.0	
			✓			15.0	15.2	0.2	
				✓		12.5	13.1	0.6	
					✓	9.2	13.3	4.2	
		✓	✓	✓	✓	15.2	**17.3**	2.1	✓
	✓	✓	✓	✓	✓	16.0	**17.5**	1.5	
D5	✓					7.0	8.1	1.1	
(Flower)		✓				10.0	14.6	4.6	
			✓			12.6	14.2	1.6	
				✓		8.6	10.0	1.4	
					✓	7.2	9.1	1.9	
		✓	✓	✓	✓	21.2	**24.7**	3.5	✓
	✓	✓	✓	✓	✓	18.9	22.6	3.7	
D6	✓					6.7	7.7	1.0	
(Fruit)		✓				8.8	10.1	1.3	
			✓			6.7	9.4	2.7	
				✓		6.1	9.4	3.4	
					✓	3.7	6.7	3.0	
		✓	✓	✓	✓	7.4	**11.1**	3.7	✓
	✓	✓	✓	✓	✓	8.4	10.4	2.0	
D7	✓					7.3	6.9	-0.4	
(Stem)		✓				11.9	14.6	2.7	
			✓			10.7	14.2	3.4	
				✓		12.6	13.4	0.8	
					✓	9.2	11.9	2.7	
		✓	✓	✓	✓	13.8	**16.5**	2.7	✓
	✓	✓	✓	✓	✓	12.6	14.6	1.9	
D8	✓					5.6	5.9	0.3	
(Entire)		✓				4.3	5.2	1.0	
			✓			6.6	9.8	3.2	
				✓		9.8	7.9	-1.9	
					✓	6.6	6.6	0.0	
		✓	✓	✓	✓	8.2	**8.5**	0.3	✓
	✓	✓	✓	✓	✓	6.9	**8.5**	1.6	

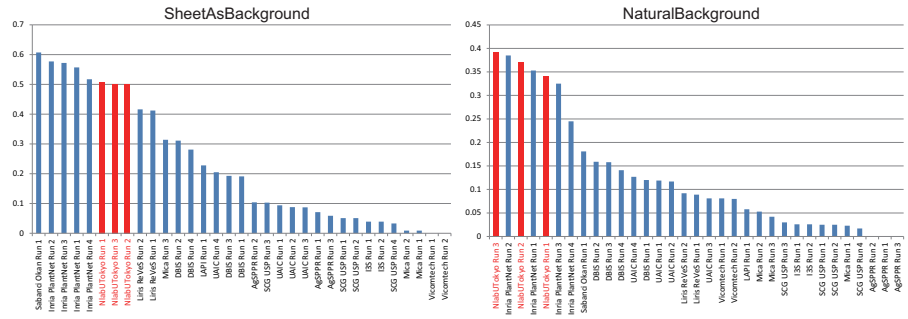

Fig. 3. Scores of all submitted runs. See [8] for details.

3 achieved the best in three runs on NaturalBackground task since it consists of multiple classifiers tuned for each sub-category. However, interestingly, the difference in performance is not large compared to the result of Run 2. Moreover, D1 classifier got better performance than D2 classifier on SheetAsBackground task. We noticed that some plants share similar appearance in different sub-categories (e.g. 'Flower' and 'Entire'). In such a case, universal classifier might result in better performance than specific ones for each sub-category.

4 Discussion

During this challenge, we bet on implementing powerful image features, rather than classification algorithms and systems. We employed multiple local features with our polynomial embedding technique to boost the performance. They are further encoded into the powerful Fisher Vector representation. Our system achieved promising performance, and got the first place in NaturalBackground and the third place in SheetAsBackground tasks. On the other hand, our learning and classification algorithms are very simple and could be improved. Although some individual plants have multiple images of different organs, our system treats them independently and loses co-occurrence information. It would be interesting to develop classification methods utilizing them in an integrated manner.

Acknowledgement

This work is partially supported by the Nakajima Foundation.

References

1. Bosch, A., Zisserman, A., Muñoz, X.: Scene classification using a hybrid generative/discriminative approach. IEEE Trans. PAMI 30(4), 712–727 (2008)

2. Burghouts, G.J., Geusebroek, J.M.: Performance evaluation of local colour invariants. Computer Vision and Image Understanding 113(1), 48–62 (2009)
3. Caputo, B., Muller, H., Thomee, B., Villegas, M., Paredes, R., Zellhofer, D., Goeau, H., Joly, A., Bonnet, P., Martinez Gomez, J., Garcia Varea, I., Cazorla, M.: ImageCLEF 2013: the vision, the data and the open challenges. In: Proc. CLEF (2013)
4. Chatfield, K., Philbin, J., Zisserman, A.: Efficient retrieval of deformable shape classes using local self-similarities. In: IEEE ICCV Workshop on Non-rigid Shape Analysis and Deformable Image Alignment (2009)
5. Csurka, G., Dance, C.R., Fan, L., Willamowski, J., Bray, C.: Visual categorization with bags of keypoints. In: Proc. ECCV Workshop on Statistical Learning in Computer Vision (2004)
6. Deng, J., Berg, A., Li, K., Fei-Fei, L.: What does classifying more than 10,000 image categories tell us? In: Proc. ECCV. pp. 71–84 (2010)
7. Fan, R.E., Chang, K.W., Hsieh, C.J., Wang, X.R., Lin, C.J.: LIBLINEAR: A library for large linear classification. Journal of Machine Learning Research 9, 1871–1874 (2008)
8. Goëau, H., Bonnet, P., Joly, A., Bakić, V., Barthelemy, D., Boujemaa, N., Molino, J.F.: The ImageCLEF 2013 Plant Identification Task. In: CLEF 2013 Working Notes (2013)
9. Hotelling, H.: Relations between two sets of variants. Biometrika 28, 321–377 (1936)
10. Khosla, A., Jayadevaprakash, N., Yao, B., Li, F.F.: Novel dataset for fine-grained image categorization: Stanford dogs. In: Proc. CVPR Workshop on Fine-Grained Visual Categorization (FGVC) (2011)
11. Lazebnik, S., Schmid, C., Ponce, J.: Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In: Proc. IEEE CVPR. vol. 2, pp. 2169–2178 (2006)
12. Lowe, D.: Object recognition from local scale-invariant features. In: Proc. IEEE ICCV (1999)
13. Nakayama, H.: Augmenting descriptors for fine-grained visual categorization using polynomial embedding. In: Proc. IEEE ICME (2013)
14. Nilsback, M.E., Zisserman, A.: Automated flower classification over a large number of classes. In: Proc. Indian Conference on Computer Vision, Graphics & Image Processing. pp. 722–729 (2008)
15. Perronnin, F., Sánchez, J., Mensink, T.: Improving the Fisher kernel for large-scale image classification. In: Proc. ECCV (2010)
16. van de Sande, K.E.A., Gevers, T., Snoek, C.G.M.: Evaluating color descriptors for object and scene recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence 32(9), 1582–96 (2010)
17. Shechtman, E., Irani, M.: Matching local self-similarities across images and videos. In: Proc. IEEE CVPR (2007)
18. Welinder, P., Branson, S., Mita, T., Wah, C., Schroff, F., Belongie, S., Perona, P.: Caltech-UCSD birds 200. Tech. rep., California Institute of Technology (2010)