Flower Classification by Using Multiple Kernel Learning

Tomohiro SAKATA† and Koichi KISE††

† School of Engineering, Osaka Prefecture University
†† Graduate School of Engineering, Osaka Prefecture University
1-1 Gakuencho, Naka, Sakai, Osaka, 599-8531, Japan
E-mail: †sakata@m.cs.osakafu-u.ac.jp, ††kise@cs.osakafu-u.ac.jp

Abstract Object classification for categories with a significant visual similarity is a difficult problem. Because natural objects are slightly different for each individual, it is difficult to classify them with one feature. Therefore multiple features are needed to classify them. As a method of combining multiple features, MKL is focused recently. In this research, we employ color, shape, and texture features. We classify the flower images by using MKL and investigate the recognition rate. As a result, the best recognition rate is 75.66% in combining three features with flower 17 category dataset published by Visual Geometry Group of Oxford University.

Key words Multiple Kernel Learning, flower classification, SIFT, MR-8

1. Introduction

In recent years, more individuals have opportunities to handle image data because of the dissemination of digital cameras and camera phones. Accordingly, many researchers have studied to recognize objects in images by computer processing. As one of the studies, there is object classification for categories that have a significant visual similarity [1]. For example to classify flower images into flower categories such as the sunflower and the dandelion, or to classify dog images into dog categories such as the Shiba and the Golden Retriever.

Unlike industrial products, natural objects such as flowers and animals are slightly different for each individual. Therefore there are many instances with different shapes and colors in the same category. Conversely, there are many instances with similar shapes and colors in different categories. Although it is common to classify objects with one feature [2], it is difficult to classify these objects by only one feature. To solve this problem, we try to use multiple features to classify objects for categories that have a significant visual similarity.

As a method that classifies objects by using multiple features, MKL (Multiple Kernel Learning) [3], [4] has been focused recently. MKL is a method for classification by combining features in optimal weights. Because it determines the optimal weight automatically, we can obtain better results by using MKL than by using a method with a uniform weight combination.

In this research, we classify flower images by MKL and investigate the recognition rate. As features, we employ color, shape and texture features for classification by using MKL. Color features are described as histograms of the three color space, the shape feature is described with SIFT features [5], and the texture feature is described with the MR-8 filter bank [6]. As a result, the maximum recognition rate was 75.66% with flower 17 category dataset published by Visual Geometry Group of Oxford University.

This paper is organized as follows. In Section 2, we review the SVM(Support Vector Machine) [7], which is the basis of MKL, and MKL. Feature representations are explained in Section 3. In Section 4, we explain experimental conditions and results, and discuss experiments. Finally, the summary and future work are described in Section 5.

2. Multiple Kernel Learning

MKL is a method for classification by combining features in optimal weight. In addition to leading to good classification accuracies, MKL can also be useful for identifying relevant and meaningful features. To begin with, we explain the SVM which is the basis of MKL.

2.1 Support Vector Machine

The SVM is a linear learning machine. The SVM computes the decision function which describes a boundary between two classes. We explain the method as follows. Given training set of n pairs $\{(x_i, y_i)\}_{i=1}^{n}$, where $x_i \in \mathcal{X}$ is an input vector and $y_i \in \{+1, -1\}$ is its label, first, we introduce input vectors from the input space $\mathcal{X}$ to a reproducing kernel Hilbert space (RKHS) $\mathcal{H}$ with mapping ψ. Then linear classifiers in $\mathcal{H}$ of the form

$$f(\boldsymbol{x}) = \boldsymbol{w}^{\mathrm{T}}\psi(\boldsymbol{x}) + b \tag{1}$$

provides flexible classifiers in $\mathcal{X}$. The sign of this decision function determines the class (e.g. $f(\boldsymbol{x}) > 0$ means the class 1). The parameters $(\boldsymbol{w}, b)$ are determined by solving the optimization problem;

$$\begin{aligned} \min_{\boldsymbol{w},b,\xi} \quad & \frac{1}{2}\|\boldsymbol{w}\|^2 + C\sum_{i=1}^{n}\xi_i \ , \\ \text{s.t.} \quad & \forall i \ \ y_i(\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}_i + b) \geqq 1 - \xi_i; \ \ \forall i \ \ \xi_i \geqq 0 \ . \end{aligned} \tag{2}$$

where ξ_i is penalty of violating constraint and $C > 0$ is a regularization constant. Equation (2) is translated into the equivalent dual optimization problem;

$$\begin{aligned} \min_{\boldsymbol{\alpha}} \quad & \sum_{i=1}^{n}\alpha_i - \frac{1}{2}\sum_{i,l=1}^{n}\alpha_i\alpha_l y_i y_l \psi(\boldsymbol{x}_i)\cdot\psi(\boldsymbol{x}_l) \ , \\ \text{s.t.} \quad & \forall i \ \ 0 \leqq \alpha_i \leqq C; \ \ \sum_{i=1}^{n} y_i\alpha_i = 0 \ . \end{aligned} \tag{3}$$

This problem depends only on inner products of inputs. Inner products can be replaced with the kernel function k;

$$\psi(\boldsymbol{x})\cdot\psi(\bar{\boldsymbol{x}}) = k(\boldsymbol{x}, \bar{\boldsymbol{x}}) \ . \tag{4}$$

After optimal parameters are determined, the decision function (Eq.(1)) is translated into

$$f(\boldsymbol{x}) = \sum_{i=1}^{n}\alpha_i k(\boldsymbol{x}_i, \boldsymbol{x}) + b \ . \tag{5}$$

2.2 Multiple Kernel Learning

Let $K_1, ..., K_m$ be m kernel matrices with $K_t = [k_t(\boldsymbol{x}_i, \boldsymbol{x}_j)]_{i,j=1,...,n}$, calculated from different features. The MKL framework extends the regular SVM formulation by additionally learning a linear mixture of the kernels, i.e.

$$K = \sum_{i=1}^{m}\beta_i K_i \tag{6}$$

with $\beta_i \geqq 0$ and $\sum_i \beta_j = 1$. Thus, Equation (1) is extended to

$$f(\boldsymbol{x}) = \sum_{i=1}^{m}\beta_i \boldsymbol{w}_i^{\mathrm{T}}\psi_i(\boldsymbol{x}) + b \ . \tag{7}$$

Coefficients β_i are incorporated into the parameter vector $\boldsymbol{w}_\beta = (\sqrt{\beta_1}\boldsymbol{w}_1, ..., \sqrt{\beta_m}\boldsymbol{w}_m)^{\mathrm{T}}$ and the feature mapping $\psi_\beta(\boldsymbol{x}_i) = (\sqrt{\beta_1}\psi_1(\boldsymbol{x}_i), ..., \sqrt{\beta_m}\psi_m(\boldsymbol{x}_i))^{\mathrm{T}}$. Then parameter β_i is involved into the optimization problem, i.e., to optimize the parameters $\boldsymbol{w}, b, \xi_i$, and β_i simultaneously. Then Equation (2) can be written as

$$\begin{aligned} \min_{\boldsymbol{\beta},\boldsymbol{w},b,\xi} \quad & \frac{1}{2}\|\boldsymbol{w}_{\boldsymbol{\beta}}\|^2 + C\sum_{i=1}^{n}\xi_i \ , \\ \text{s.t.} \quad & \forall i \ \ y_i(\langle \boldsymbol{w}_{\boldsymbol{\beta}}, \psi_{\boldsymbol{\beta}}(\boldsymbol{x}_i)\rangle) + b) \geqq 1 - \xi_i; \\ & \forall i \ \ \xi_i \geqq 0; \ \ \forall i \ \ \beta_i \geqq 0 \ . \end{aligned} \tag{8}$$

In the case of $m = 1$, the above problem reduces to the original SVM. The above optimization problem can be translated into the following semi-infinite program [8];

$$\begin{aligned} \min_{\lambda,\boldsymbol{\beta}} \quad & \lambda \ , \\ \text{s.t.} \quad & \lambda \geqq \sum_{i=1}^{n}\alpha_i - \frac{1}{2}\sum_{i,l=1}^{n}\alpha_i\alpha_l y_i y_l \sum_{j=1}^{m}\beta_j k_j(\boldsymbol{x}_i, \boldsymbol{x}_l); \\ & \forall i \ \ \alpha_i \in \mathbb{R}^n; \ \ \forall i \ \ 0 \leqq \alpha_i \leqq C; \\ & \sum_{i=1}^{n} y_i\alpha_i = 0; \ \ \forall j \ \ \beta_j \geqq 0 \ . \end{aligned} \tag{9}$$

The optimized parameters are obtained by solving the above problem. Finally, the decision function can be written as

$$f(\boldsymbol{x}) = \sum_{i=1}^{n}\sum_{j=1}^{m}\alpha_i\beta_j k_j(\boldsymbol{x}_i, \boldsymbol{x}) + b \ . \tag{10}$$

We take a one-vs-all approach to account for multi-class problem.

3. Feature representation

We employ commonly used features of color, shape, and texture [1]. In the following subsections we explain the representation of each feature.

3.1 Color feature

There are several color spaces and we do not know which color space is the best to classify flower images. Therefore in order to examine which color space is better, we use three color space, i.e. RGB, HSV, and CIE L*a*b*. Each color space representation is defined as its frequency histogram of three parameters.

3.2 Shape feature

We compute SIFT features [5] to describe shape features. SIFT features of an object are less affected by scaling and rotation.

We employ the Bag-of-features method to describe shape features. We review how to obtain Bag-of-features as follows. SIFT feature is described as a 128 dimensional vector. From one image, several hundreds to thousands vectors are obtained. These vectors are clustered by the k-means method. Then the cluster centers are called Visual Words. An image is described as a frequency histogram of Visual Words. This frequency histogram is called Bag-of-features.

3.3 Texture feature

We describe the texture by convolving the images with the MR-8 filter bank [6]. This filter bank contains filters at multiple orientations. Thus, we can obtain rotation invariant features by choosing the maximum response over orientations. We obtain a 8 dimensional vector from one pixel. Like the shape feature, these vectors are clustered to obtain Visual Words. Finally, the texture feature is described as a frequency histogram of Visual Words.

Figure 1 flower dataset

Figure 2 unfavorable images

4. Experimental results

4.1 Dataset

We employed a flower dataset published by Visual Geometry Group of Oxford University [1]. This dataset contains 1360 images of 17 categories (80 images per category). Figure 1 shows samples of the dataset. Some images of unfavorable condition are shown in Figure 2. For example some images contain non flower objects such as a golf ball, insects, and characters, or flowers occupy small extent of image.

4.2 Condition

We took the 5-fold cross validation to investigate the MKL performance for the flower classification. The dataset was divided into five. Then, we appropriated one of them for test and the others for training and repeated this for all divisions. We obtained three features from the dataset, and described each feature as a 300 dimensional representation. Color features were 100 dimensions per parameter, and we made 300 Visual Words for shape and texture features. We combine these features and classify the images by using MKL, and investigate the recognition rate. In the case that we classified images with one feature, we employed the regular SVM. The baselines SVM and MKL were implemented using the Shogun library [8].

We employed as the kernel function of the MKL and SVM the Gaussian kernel;

$$k(\boldsymbol{x}_i, \boldsymbol{x}_j) = \exp(-\frac{\|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2}{\sigma^2}) \tag{11}$$

where the parameter σ was determined for each feature. The σ value of a feature was the value which achieved the best recognition rate by the SVM with the feature. This value is also utilized to combine features by using MKL.

4.3 Experiment

Table 1 shows the recognition rates, and Table 2 shows the average weights of all divisions.

In this result, combining HSV, Shape and Texture features achieved the recognition rate 75.66%. We can see that the recognition rate is improved by combining features. Thus, using more features will improve the recognition rate. The shape feature achieved the highest recognition rate in one feature, and the weight was also large. This shows that shape feature is important for the flower classification

We can see the definite difference of performance among the divisions. In the result where the difference is the most significant, there is the more than a 20% difference in the recognition rate. This indicates two matters. First, MKL is easily affected by the training data. Thus we need to select the training data carefully. Second, training data is insufficiency in this experiment. Thus, we need to examine the flower classification with a larger dataset. If we employ the images which are easy to classify, the recognition rate might not be improved so much.

To investigate which flower is easy or difficult for classification, the recognition rates of each flower at combining HSV, shape, and texture features are shown in Table 3. The sunflower achieved the highest performance. This is because, Sunflowers comparatively have less differences among each individual. Moreover, sunflower is mostly taken a picture from the front. Therefore, the features of sunflowers are similar for each image and useful for classification. Tulips are the most difficult for classification in this experiment. The tulip

Table 1 Recognition rate[%]

	1 feature					2 features							3 features		
	RGB	HSV	CL	S	T	RGB+S	HSV+S	CL+S	RGB+T	HSV+T	CL+T	S+T	3(RGB)	3(HSV)	3(CL)
1	27.6	36.4	34.9	54.0	39.3	58.1	61.4	61.4	45.6	49.3	48.2	59.2	62.1	65.1	62.9
2	42.6	50.4	46.3	64.7	50.7	71.0	72.8	73.2	52.9	54.8	54.4	67.3	72.4	73.9	74.3
3	47.1	53.3	47.4	73.2	57.7	78.3	79.8	77.3	64.0	69.9	60.7	76.5	78.7	79.8	77.9
4	50.0	56.3	52.9	73.2	55.1	78.4	79.0	77.6	61.8	67.3	61.8	75.7	80.1	81.2	80.5
5	48.1	55.1	49.6	71.0	46.0	73.2	78.4	74.6	53.7	62.9	62.1	70.2	73.9	78.3	76.1
AVG	43.08	50.30	46.22	67.22	49.76	71.80	74.28	72.82	55.60	60.84	57.44	69.78	73.44	75.66	74.34

S:Shape, T:Texture, CL:CIE L*a*b*, 3(⋄):⋄+S+T

Table 2 Kernel weights

Combination	Weight		
RGB : S	0.145	0.855	—
RGB : T	0.412	0.588	—
HSV : S	0.216	0.784	—
HSV : T	0.526	0.474	—
CL : S	0.171	0.828	—
CL : T	0.466	0.534	—
S : T	0.895	0.105	—
RGB : S : T	0.125	0.788	0.086
HSV : S : T	0.157	0.756	0.103
CL : S : T	0.198	0.718	0.084

S:Shape, T:Texture, CL:CIE L*a*b*

Table 3 recognition rate[%] of each flower at combining HSV, shape, and texture features

Flower	Rate	Flower	Rate
Daffodil	72.5	Sunflower	90.0
Snowdrop	71.3	Daisy	86.3
LilyValley	72.5	Coltsfoot	82.5
Bluebell	70.0	Dandelion	82.5
Crocus	63.8	Cowslip	60.0
Iris	82.5	Buttercup	77.5
Tigerlily	83.8	Windflower	80.0
Tulip	45.0	Pansy	81.3
Fritillary	85.0		

was often recognized as a Daffodil, and sometimes the other flower. This is because tulips and daffodils have the similar color and shape. However, dandelions and coltsfoots, which have the similar color and shape, achieves a decent recognition rate. The other reason might also relate to the bad result of tulips. Another reason is that the images of tulip are shot from various orientations. From this, the features have differences among each image, and we cannot obtain significant features.

5. Conclusion

In this paper, we investigate the MKL performance on the flower classification. As a result, combining HSV, shape, and texture features achieved the recognition rate 75.66%. Combining three features achieve the better recognition rate than one feature or two features. Therefore, if we employ other feature descriptors, the recognition rate might be further improved. We also confirm that the data for training is insufficient for precise classification. Thus we need to examine the flower classification with more flower images.

Future work is as follows. First, we examine the flower classification using more features. Secondly, we utilize more flower images and increasing flower categories. Last, we investigate how the backgrounds affect the classification.

Acknowledgements

This research was supported in part by the Grant-in-Aid for Scientific Research (B)(22300062) from Japan Society for the Promotion of Science (JSPS).

References

[1] M.-E. Nilsback and A. Zisserman, "A visual vocabulary for flower classification," IEEE Conference on Computer Vision and Pattern Recognition (2006), 2006. http://www.robots.ox.ac.uk/ vgg/data/flowers/

[2] S. Belongie, J. Malik, and J. Puzicha, "Shape matching and object recognition using shape contexts," IEEE Transactions on Pattern Analysis and Machine Intelligence, pp.509–522, IEEE Computer Society, 2002.

[3] S. Nakajima, "Multiple kernel learning for object classification," Technical Report on Information-Based Induction Sciences 2009 (IBIS2009), 2009.

[4] G. Lanckriet, N. Cristianini, P. Bartlett, L.E. Ghaoui, and M.I. Jordan, "Learning the kernel matrix with semi-definite programming," Journal of Machine Learning Research, pp.27–72, 2004.

[5] D.G. Lowe, "Distinctive image features from scale-invariant keypoints," IJCV, pp.91–110, 2004.

[6] M. Varma and A. Zisserman, "Classifying images of materials: Achieving viewpoint and illumination independence," ECCV, voluime 3, pp.255–271, May 2002.

[7] B. Scholkopf and A.J. Smola, Learning with kernels: Support vector machines, regularization, optimization, and beyond, MIT Press, 2001.

[8] S. Sonnenburg, G. Ratsch, C. Schafer, and B. Scholkopf, "Large scale multiple kernel learning," Journal of Machine Learning Research 7, pp.1531–1565, 2006.