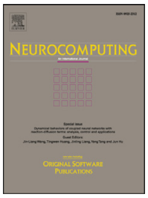
Neurocomputing 257 (2017) 16–23

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom



# A multi-scale kernel learning method and its application in image classification

Jian Bao[a], Yangyang Chen[a,*], Li Yu[b], Chunwei Chen[a]

[a] Institute of Intelligent and Software Technology, Hangzhou Dianzi University, Hangzhou, China
[b] Information and communication engineering institute, Wuyi University, Jiangmen, China

ARTICLE INFO

*Article history:*
Received 12 July 2016
Revised 18 November 2016
Accepted 19 November 2016
Available online 6 February 2017

*Keywords:*
Multi-scale kernel
Multiple kernel learning
Kernel polarization
Image classification

ABSTRACT

The success of support vector machine depends on the kernel function, which directly affects the performance of SVM. Therefore, to improve the generalization of SVM, we will study the selection of kernel function. The multi-scale kernel method is one particular type of multiple kernel method which combines multi-scale kernels through a multi-kernel learning framework. It has the capability of generalizing not only the scattered region of a training set very well but also generalizing the dense region of data sets very well. Inspired by the advantages of the multi-scale kernel learning method, we applied kernel centered polarization to construct an optimization problem which was used to learn the multi scale kernel function and select the optimal parameters. A thorough analysis and proofs are provided. Experimental results show that the proposed kernel learning method and algorithm are reasonable and effective and have very good generalization performance.

## 1. Introduction

The kernel method [1,2,6] is a learning method based on kernel functions; it is widely used in various fields of machine learning. Support vector machines (SVMs) are the most successful application of kernel methods. The kernel function can easily extend a linear SVM to nonlinear, because the kernel function of complex inner product computation of high-dimensional space is converted into low dimensional input space kernel function computation, eliminating the need to design feature space [3], and cleverly solving calculations in high dimensional space of problems such as "dimension disaster". Cortes and Vapnik [4] proposed the SVM method, and because of its inherent advantages, the SVM has become a hot spot in the machine learning field since that time. The kernel method maps the input space to the feature space. Most of the time, it leads to good generalization effects. But if the selected kernel function is improper, the generalization performance will not be as good. So, the success of kernel methods depends on the selection of kernels.

In general, the methods of cross validation or "leaving one out" are used to choose kernel functions. The algorithms of the two methods are simple, but their complexity is high, and they require a large amount of calculation. For the purpose of overcoming the disadvantages of these approaches, many methods have been proposed to minimize the upper boundary of errors. The RM (radius margin) [5] boundary is one of the most common error boundaries. For the sake of further improving calculation efficiency, many kernel measurement methods select a proper kernel function by measuring the distribution of samples in feature space. For example, Cristianini [7] proposed the kernel target alignment (KTA) for the first time. This method can be used to measure the quality of a kernel matrix. Additionally, it is easy to implement the algorithm with low complexity, and is widely used in the selection of kernel function. Subsequently, Baram [8] proposed kernel polarization which can be regarded as non-normalized kernel alignment; however, the method described above is single kernel learning method. In addition, each kernel function has a different characteristic, so in different scenarios, there is a large difference in performance of the kernel function. To ameliorate the above problems, multi-kernel learning methods [10–16] appeared. Although multi-kernel methods have been successfully applied, they are only generating a new kernel function according to the Mercer condition and using the linear combination of simple single-kernel functions. There is no perfect theory for the selection of kernel functions. These methods cannot solve the problem of the uneven distribution of samples, limiting the expression ability of the decision function. In this case, the multi-scale multiple kernel learning has arisen. For example Kingsbury [17] used several multi scale kernels to classify step by step. This method has the capability to seek out a suitable kernel scale for the input space for each local area. This kind of method is flexible and practical.

* Corresponding author
*E-mail address:* cyy3900@foxmail.com (Y. Chen).

http://dx.doi.org/10.1016/j.neucom.2016.11.069
0925-2312/© 2017 Elsevier B.V. All rights reserved.

For multi-scale kernel learning, it is critical to determine the multi-scale kernel coefficients. There are many ways to determine the coefficients of the kernel function. Some use the idea of averaging the effect of the kernel [17,31–33], so that different kernel functions have the same effect on the decision function. In addition, some use intelligent optimization methods [31] to obtain the objective function of the parameter values. However, the large numbers of iterative steps in this kind of optimization method greatly increase the learning time for SVMs. These methods have different kernel synthesis coefficients, but they are still empirical methods. With the increase of the number of kernel functions, the dimension of the optimization problem will increase greatly. However, kernel polarization makes use of the information from the complete training set and can be computed efficiently. Furthermore, it is independent of the actual learning machine used. We applied kernel polarization to construct an optimal problem which was used to learn the multi scale kernel function and select the optimal parameters. A thorough analysis and proofs are provided. In summary, the contributions of this paper are:

(1) A multi-scale kernel learning method is proposed.
(2) It is proven that this method can determine the optimal multi-scale kernel function with low algorithm complexity.
(3) Comprehensive experiments were conducted to empirically analyze our method and the algorithm on six image databases. The experimental results demonstrate that our algorithm outperforms other methods including SVM, TSVM [34], LapSVM [35], and k-nearest neighbor.

The following sections will be organized as follows: Section 2 introduce kernel evaluation measures and multi-scale kernel learning methods. We give a detailed description and analysis of multi-scale multiple kernel learning in Section 3.The detailed description of the proposed method and algorithm are presented in Section 4. Section 5 presents the experimental results. The paper concludes in Section 6.

## 2. Related works

### 2.1. Kernel evaluation measures

The model selection problem involves selecting the kernel and optimization. The kernel evaluation measure is a model selection method. It [18–22] is a good measure of model selection, which utilizes the distribution of the samples, and is an efficient method. In addition, it is independent of any particular learning method. Cristianini [7] first proposed kernel target alignment (KTA). It has been widely used in kernel function selection. Baram put forward kernel polarization [8], which can be seen as non-normalized KTA;

$$\mathrm{P}(K,Y)=\langle K,Y\rangle_F=\sum_{i=1}^{n}\sum_{j=1}^{n}y_iy_jk\left(x_i,x_j\right), \tag{1}$$

$K$ is the kernel matrix, $K=k(x_i,x_j)$, $y=(y_1,\ldots,y_n)^T$, $\langle K,\ yy^T\rangle_F$ is the Frobenius inner product of $K$ and $yy^T$, $Y{=}yy^T$ is the target matrix; the kernel polarization criterion represents the similarity of the kernel matrix $K$ and the target matrix. The greater the kernel polarization criterion value is achieved, the better the kernel function will be. So, when selecting a kernel function, it will be better to choose a kernel that allows reaching the highest kernel polarization criterion value.

### 2.2. Multiple kernel learning

Multiple kernels learning [12–16] is a flexible learning based on the kernel function. This kind of method is better than single kernel learning. The simplest and the most commonly used multiple kernel learning method is to linearly combine the basic kernel functions together, which can be described as follows:

$$\begin{cases} K=\sum_{i=1}^{m}\alpha_iK_i \\ \sum_{i=1}^{m}\alpha_i=1 \\ \alpha_i\geq 0, \end{cases}$$

$K_i$ represents the basic kernel, m stands for the sum of the basic kernels, $\alpha_i$ represents the weighted coefficient of $K_i$. Multiple kernel learning can be converted into selecting the basic kernel function and selecting the appropriate weight coefficient. The feature space of the samples is a combination of several feature spaces. Due to the use of a combination of various basic kernel feature mapping capabilities, it solves the problem of selecting the kernel function and related variables very well. Multiple kernel learning greatly improves the recognition rate and generalization ability. The most important issue is how to learn to obtain the weights. To solve this problem, more effective multiple kernel learning theories and methods have been proposed in recent years. In the early stage, the multiple kernel was learned by boosting methods [39], semi-definite programming [12], quadratically constrained quadratic program [13], semi-infinite linear program [14,25], and by Hyper kernels [11]. Subsequently, Simple MKL [28,29] was proposed. By combining the multiple kernel learning and SVM method, multiple kernel learning has been applied in many fields.

### 2.3. Multi-scale kernel learning

The multi-scale kernel learning method is one kind of specialized kernel method. This method fuses several different scale kernels together, and is very flexible and effective. This method is currently performing well in application. For example, Kingsbury [17] used two different scale kernels to perform step by step classification. Zheng [30] and Yang [31] proposed multi-scale support vector regression which was used to estimate functions and forecast the time series. In addition, multi-scale kernel and SVMs can be combined together and applied to image compression [32], hot spot detection and modeling [33] and measuring time-series similarity [38].Our paper uses polarization as the objective function to construct an optimal problem which can be used to learn the proposed multi-scale kernel. Our algorithm is simple and effective.

## 3. Multi-scale multiple kernel method

The multi-scale kernel learning method is one kind of specialized kernel method which is flexible and effective. This method fuses kernels together. A series of multi-scale kernel functions should be found as the base kernel in the first step. Then the multi-scale multiple kernel function should be constructed on this foundation. A Gaussian kernel can be described as follows:

$$k(x,z)=\exp\left(-\frac{\|x-z\|^2}{2\sigma^2}\right). \tag{3}$$

A Gaussian kernel is one kind of multi-scale kernel function which is capable of universal expression. We will use it in our paper and each function will be assigned a different bandwidth.

$$\exp\left(-\frac{\|x-z\|^2}{2{\sigma_1}^2}\right),\ldots,\exp\left(-\frac{\|x-z\|^2}{2{\sigma_m}^2}\right), \tag{4}$$

$\sigma_1<\cdots<\sigma_{\mathrm{m}}$. For the Gaussian kernel, the higher bandwidth we use, the flatter the function will be. Based on this property, we know that the functions with low bandwidths will be a better measure for samples with large changes, and the functions with higher bandwidths will be a good measure for samples with small changes. They all have good performance.

If you want to use multi-scale multiple kernel learning, the most obvious approach is the direct method, multi-scale kernel sequence learning. Kingsbury's [17] paper uses the combination of two multi-scale kernel functions for the purpose of learning. One of the large scale kernel functions is used to fit the smooth change of the samples, another small scale kernel function is used to fit the larger changes, and then the results are applied to the rear, and the optimal classification results are gradually obtained.

We researched a two scale $k_1$ and $k_2$ kernel synthetic classification problem. In order to obtain the Decision function *f*($x$):

$$f(x) = f_1(x) + f_2(x), \tag{5}$$

where

$$f_1(x) = \sum_{i=1}^{n} \alpha_i k_1(x_i, x) + b_1, \tag{6}$$

and

$$f_2(x) = \sum_{i=1}^{n} \beta_i k_2(x_i, x) + b_2, \tag{7}$$

$k_1$ is a large-scale kernel function such as a high bandwidth Gaussian function. $\alpha_i$ is the related coefficient of the kernel $k_1$ and is chosen from support vectors corresponding to the smooth areas of the *f*($x$). $k_2$ is similar to $k_1$. $k_2$ is a small scale kernel function. $\beta_i$ is chosen from support vectors corresponding to the rough areas of the *f*($x$). The specific method first creates $f_1(x)$ using a single large scale kernel. The large scale kernel fits the smooth areas of the training set but it cannot fit the remaining areas. Then, $f_2(x)$ is created by using a small scale (finer scale) kernel $k_2$. It is obvious that $f_1(x) + f_2(x)$ will suit the samples better than $f_1(x)$ alone.

This method uses quadratic programming to obtain the relevant parameters, so the algorithm complexity is very high, and requires a great amount of calculation. When there is large amount of data, the support vector will greatly increase, and the algorithm will not be able to handle it.

We propose a multi-scale combined kernel determined with the help of a centered kernel polarization criterion that solves this problem. This method is simple and has a low computational complexity. At the same time, combined kernel learning is better than single kernel learning in classification and generalization performance.

## 4. Proposed multi-scale kernel learning

In this section, we present an optimal method, which is used to learn a multi-scale kernel function. $K$ is a multi-scale kernel function.

$$K = \alpha K_1 + (1 - \alpha) K_2, \alpha \in [0, 1], \tag{8}$$

$K_1$ and $K_2$ are PDS (positive definite and symmetrical) multi-scale kernel functions. They are Gaussian functions.

$$K_1 = \exp\left(-\frac{\left\|x_i - x_j\right\|^2}{2{\sigma_1}^2}\right), \tag{9}$$

$$K_2 = \exp\left(-\frac{\left\|x_i - x_j\right\|^2}{2{\sigma_2}^2}\right), 0 < \sigma_1 < \sigma_2. \tag{10}$$

In order to adapt to complex classification problems, we assign $\sigma_1$ with a small value to measure samples with dynamic changes and assign $\sigma_2$ with a large value to measure samples with small changes. Then, the kernel function $K$ include 3 parameters: $\alpha$, $\sigma_1$ and $\sigma_2$.

We define an optimal function F($\alpha$, $\sigma_1$, $\sigma_2$):

$$F(\alpha, \sigma_1, \sigma_2) = P(K, Y_c) = \langle K, Y_c \rangle_F. \tag{11}$$

Let $\mathrm{S} = \{x_1, x_2, \ldots, x_n\}$ be the data set and $\mathrm{Y} = \{y_1, y_2, \ldots, y_n\}$ be the corresponding label to set S. $Y_C = YY^T$ is the target matrix. Then, $\mathrm{n}_+$ samples are class +1, and $n_-$ samples are class −1. P($K$, $Y_C$)(1) is the kernel polarization which is used to measure the similarity of the kernel matrix K and the target matrix $Y_C$. The centered label matrix [24] $Y_c$ is described as follows:

$$\mathrm{Y_c} = \begin{pmatrix} 4\frac{\mathrm{n}_-^2}{\mathrm{n}^2}\mathrm{e}_{\mathrm{n}_+ \times \mathrm{n}_+} & -4\frac{\mathrm{n}_+\mathrm{n}_-}{\mathrm{n}^2}\mathrm{e}_{\mathrm{n}_+ \times \mathrm{n}_-} \\ -4\frac{\mathrm{n}_+\mathrm{n}_-}{\mathrm{n}^2}\mathrm{e}_{\mathrm{n}_- \times \mathrm{n}_+} & 4\frac{\mathrm{n}_+^2}{\mathrm{n}^2}\mathrm{e}_{\mathrm{n}_- \times \mathrm{n}_-} \end{pmatrix}. \tag{12}$$

Eq. (7) can be represents as:

$$\begin{aligned} &F(\alpha, \sigma_1, \sigma_2) \\ &= \sum_{i,j=1}^{n} [K]_{i,j} [Y_c]_{i,j} \\ &= \frac{4}{n^2}\Bigg[\sum_{y_i = y_j = 1, i \neq j} n_-^2 \left(\alpha \exp\left(-\frac{\left\|x_i - x_j\right\|^2}{2\sigma_1^2}\right)\right. \\ &\quad \left. + (1 - \alpha) \exp\left(-\frac{\left\|x_i - x_j\right\|^2}{2{\sigma_2}^2}\right)\right) \\ &\quad + \sum_{y_i = y_j = -1, i \neq j} n_+^2 \left(\alpha \exp\left(-\frac{\left\|x_i - x_j\right\|^2}{2\sigma_1^2}\right)\right. \\ &\quad \left. + (1 - \alpha) \exp\left(-\frac{\left\|x_i - x_j\right\|^2}{2{\sigma_2}^2}\right)\right) \\ &\quad + \sum_{y_i \neq y_j} 2n_+n_- \left(\alpha \exp\left(-\frac{\left\|x_i - x_j\right\|^2}{2\sigma_1^2}\right)\right. \\ &\quad \left. + (1 - \alpha) \exp\left(-\frac{\left\|x_i - x_j\right\|^2}{2{\sigma_2}^2}\right)\right)\Bigg] \\ &\quad + \frac{4n_+n_-}{n}. \end{aligned} \tag{13}$$

**Theorem 1.** *Let* $\sigma_1$ *be a small fixed constant.* F($\alpha$, $\sigma_1$, $\sigma_2$) *has only two parameters,* $\alpha$ *and* $\sigma_2$. *In some conditions,* F($\alpha$, $\sigma_1$, $\sigma_2$) *has global maximization value.*

**Proof.**

$$\begin{aligned} \mathrm{F}(\alpha, \sigma_1, \sigma_2) &= \sum_{\mathrm{i,j}=1}^{\mathrm{n}} [\alpha \mathrm{K}_1 + (1 - \alpha)\mathrm{K}_2]_{\mathrm{i,j}} [\mathrm{Y_c}]_{\mathrm{i,j}} \\ &= \alpha \sum_{\mathrm{i,j}=1}^{\mathrm{n}} [\mathrm{K}_1]_{\mathrm{i,j}} [\mathrm{Y_c}]_{\mathrm{i,j}} + (1 - \alpha) \sum_{\mathrm{i,j}=1}^{\mathrm{n}} [\mathrm{K}_2]_{\mathrm{i,j}} [\mathrm{Y_c}]_{\mathrm{i,j}} \\ &= \alpha \mathrm{P}(\sigma_1) + (1 - \alpha)\mathrm{P}(\sigma_2). \end{aligned} \tag{14}$$

$$\mathrm{P}(\sigma) = \langle K, Y_c \rangle_F = \sum_{\mathrm{i,j}=1}^{\mathrm{n}} [\mathrm{K}_1]_{\mathrm{i,j}} [\mathrm{Y_c}]_{\mathrm{i,j}}. \tag{15}$$

Ref. [9] proves that $P(\sigma)$ has global maximum value in the case of certain conditions.

$$\sigma^* = \arg\max P(\sigma) = \arg\max \langle K, Y_c \rangle_F \tag{16}$$

As F($\alpha$, $\sigma_1$, $\sigma_2$) is a linear combination of P($\sigma$), and P($\sigma$) has global maximum value, F($\alpha$, $\sigma_1$, $\sigma_2$) also has global maximum value.

Thus:

$$Max(P(\sigma_1)) = M1, \tag{17}$$

$$Max(P(\sigma_2)) = M2, \tag{18}$$

$$F(\alpha,\sigma_1,\sigma_2) = \alpha P(\sigma_1) + (1-\alpha)P(\sigma_2) + \mathrm{C}. \tag{19}$$

So, we can draw from the above:

$$\begin{aligned}\mathrm{Max}(\mathrm{F}(\alpha,\sigma_1,\sigma_2)) &= \alpha \mathrm{Max}P(\sigma_1) + (1-\alpha)\mathrm{Max}(P(\sigma_2)) + C\\ &\alpha \mathrm{M1} + (1-\alpha)\mathrm{M2} + \mathrm{C}.\end{aligned} \tag{20}$$

We can see that $F(\alpha,\sigma_1, \sigma_2)$ has global maximum value. $\mathrm{F}(\alpha,\sigma_1, \sigma_2)$ is the kernel polarization of the kernel $K$. The kernel polarization criterion represents the similarity between the kernel matrix K and the target matrix. The greater the kernel polarization criterion value achieved, the better the kernel function will be. So, when selecting kernel function, it is better to choose a kernel that allows the kernel polarization value to reach as high a value as possible. So, at the point of the global maximum value of $\mathrm{F}(\alpha,\sigma_1, \sigma_2)$, the chosen kernel is the optimal required kernel. Additionally, when $\sigma_1$ and $\sigma_2$ are assigned some fixed value, we find that $\sigma_2$ linearly varies with $\alpha$.

A Gaussian kernel is a multi-scale kernel and has a good generalization capability. When choosing a Gaussian kernel with a low bandwidth, the SVM can classify the samples with dynamic changes. On the contrary, when choosing a Gaussian kernel with a high bandwidth, the SVM can classify the samples with smooth changes.

Eq. (8) is the multi-scale combined kernel function. We will use an optimal method to solve it. In our method, $\sigma_1$ and $\sigma_2$ are two parameters. One is a large value and the other is a small value. For the sake of simplifying the problem, the parameter $\sigma_1$ is assigned a small fixed constant. Then, $\alpha$ and $\sigma_2$ are two unknown parameters. We can obtain the maximum value of $F(\alpha,\sigma_1, \sigma_2)$ by the grid search method and the kernel function is optimal at the maximum.

$F(\alpha, \sigma_1, \sigma_2)$, $\alpha \in [0, 1]$, $\sigma_1$ is a small constant, and $0 < \sigma_1 < \sigma_2 < M$. $M$ is the upper bound of $\sigma_2$. According to the above we know that $F(\alpha, \sigma_1, \sigma_2)$ has a maximum value. In addition, the calculation complexity of $F(\alpha, \sigma_1, \sigma_2)$ is relatively low, so we can use the grid search method to solve it. Searching in the range of $\alpha \in [0, 1]$, $\sigma_2 \in [\sigma_1, M]$, we can obtain the optimal multi-scale kernel parameters at the maximum of $F(\alpha, \sigma_1, \sigma_2)$. In summary, the algorithm in this paper can be described as follows.

The grid algorithm for multi-scale kernel learning:

**Input:** a small constant P for parameter $\sigma_1$, the upper bound value M of parameter $\sigma_2$

Step 1: Set $\sigma_1$ with a small value P.

Step 2: Let $\alpha = 0 \rightarrow 1$ and $\sigma_2 = P + 1 \rightarrow M$, iterate and compute the polarization of $F(\alpha, \sigma_1, \sigma_2)$.

Step 3: find the maximum value of the Polarization of $F(\alpha, \sigma_1, \sigma_2)$, then the program comes to an end.

At the maximum point of $F(\alpha, \sigma_1, \sigma_2)$, the parameters $\alpha$, $\sigma_1$ and $\sigma_2$ decide one kernel function $K = \alpha K_1 + (1-\alpha)K_2$. This kernel is the optimal kernel that we require.

## 5. Experiments and analysis

To verify the proposed multi-scale kernel functions' generalization capability and the rationality of the algorithm, we performed some experiments. Six image sample sets were chosen in our experiment. Object image samples COIL-20 [23], hand written digit image samples Semeion and MNIST [24], facial image samples ORL [40], shape image samples MPEG-7 [26], and a sample of 15 natural scene categories [27].We give some example images from the 6 image data sets in Fig. 4. We summarize these data sets in table 1.

**Table 1**
Descriptions of six sample sets used in the experiments.

| Datasets | Sample size | Dimension | Number of classes |
|---|---|---|---|
| COIL-20 | 1440 | 1024 | 20 |
| MNIST | 2000 | 784 | 10 |
| Scene15 | 1500 | 21,504 | 15 |
| ORL | 400 | 1024 | 40 |
| Semeion | 1593 | 256 | 10 |
| MPEG-7 | 700 | 200 | 35 |

The rationality and performance of the multi-scale kernel learning is first verified. Then, the performance is compared with SVM, TSVM [34], LapSVM [35], and k-nearest neighbor. Specific configurations are presented in Section 5.2.

### 5.1. Sample sets and dispositions

COIL-20 is an image sample set [23] containing 1440 images. The images are divided into 20 object categories. 72 images are captured from different angles of each category. Images from the data set are resized to 32 × 32 pixel arrays and stored in a 1024-D array.

The MINIST [24] is handwritten digit image sample set, which contains a test set and training set. The test set includes 2000 images. Each image is 28 × 28 in resolution and is stored in a 784-D array. We used test set which contains ten classes (digit "0" to "9").

Semeion is a handwritten digit sample set containing 1593 handwritten digits which were written by 80 persons. Each image was normalized to 16 × 16 pixel arrays and was stored in a 256 array.

The ORL data set of faces [40] is a data set contains 400 facial images. These images from 40 different themes were captured at different times, changing the facial expressions, light, and details of the face. Each of the images is represented by a 1024-dimensional array.

Scene 15 is an image sample set [27] containing 1500 images from 15 natural scenes. The scenes are bedroom, CAL suburb, industrial, kitchen, living room, MIT coast, MIT forest, MIT highway, MIT Mountain, MIT inside city, MIT tall building, MIT open country, MIT Street, PAR office, and store. Locality-constrained linear coding [37] was applied to extract features.

MPEG-7 [33] is a shape image sample set. To assess the classification performance on this sample set, we used the Hungarian method to align the 700 images which were chosen from the MPEG-7 [33] sample set and we used the shape algorithm [36] to compute pair wise distances between images. Finally, each shape image was stored in a 200-D feature vector.

We randomly selected samples from the data sets in the experiment. Because the sizes of the data set are different, selecting the same number of samples for all datasets would be improper. Therefore, the percentage of training sample was chosen as a label for each sample set.

### 5.2. Performance evaluation and comparisons

First, some experiments were done to verify the rationality and performance of the proposed multi-scale kernel learning. $F(\alpha, \sigma_1, \sigma_2)$ Includes three parameters $\alpha$, $\sigma_1$, $\sigma_2$. $\sigma_1$ which were assigned a small fixed value. Therefore, there were only two unknown parameters. Then, the grid algorithm was used to learn the kernel and look at the relationship between the kernel polarization and the recognition rate of the multi-scale kernel.

Then, we compared the classification performance of four other approaches and our method.

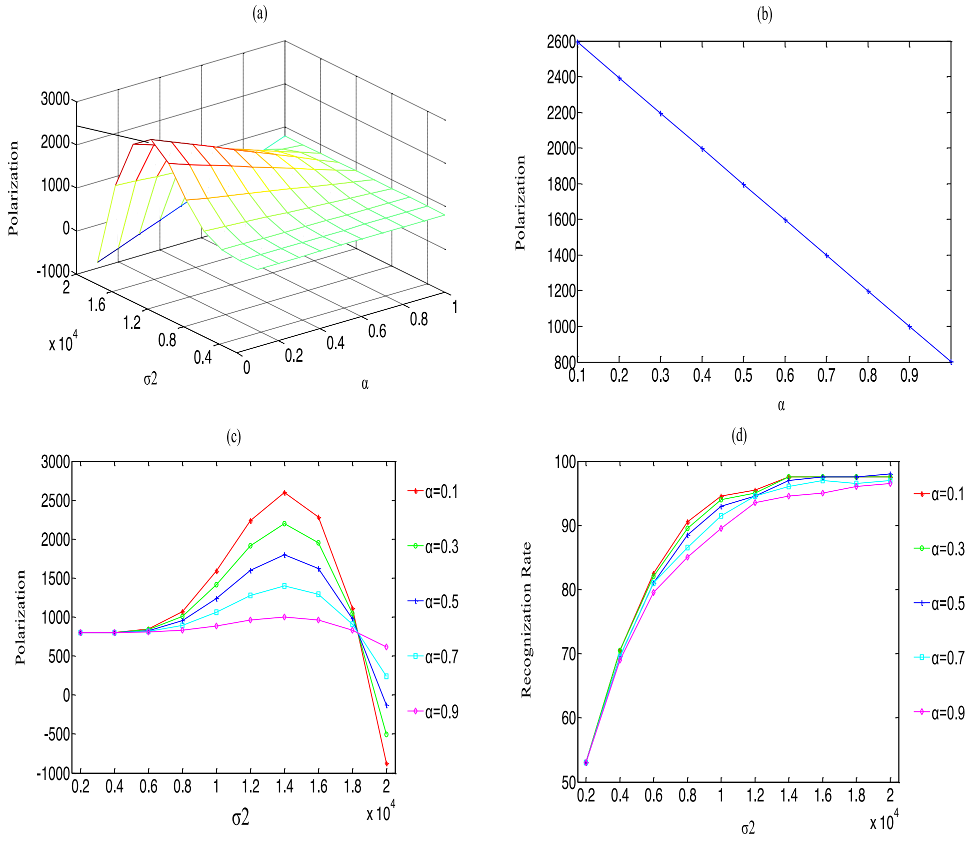


**Fig. 1.** The polarization values and the classification rates of the multi-scale kernel vary with parameters on image data set of MINIST. (a) Polarization values vary with parameter $\alpha$ and $\sigma_2$; (b) When $\sigma_2$ is 14,000, Polarization values vary with parameter $\alpha$; (c) Polarization value vary with parameter $\sigma_2$; (d) Recognition values vary with parameter $\sigma_2$.

(1) Multi-scale kernel learning with SVM (MSKL).We used the grid algorithm to select Parameters $\alpha$ and $\sigma_2$. Then, the optimal multi-scale kernel was selected. The penalty parameter was adjusted to the optimum value.
(2) SVM with RBF kernel (SVM). The penalty parameter C and kernel parameters $\sigma$ were adjusted to the optimum values.
(3) Laplacian Support Vector Machines (LapSVM) [35]. LapSVM extends SVM through using the Laplacian graph to represent the geometry of unlabeled and labeled samples. We adjusted the related parameters to the optimal values.
(4) K-nearest neighbor algorithm (kNN). This method classifies the sample through searching the nearest neighbor. The sample belongs to the same category with its nearest neighbors. We adjusted the parameter k to the optimal value.
(5) Transductive SVM (TSVM) using the Gaussian kernel function. TSVM improves the classification performance of SVM through using unlabeled data. All the parameters were adjusted to the optimal values.

### 5.3. Experimental results

Fig. 1 (a), shows that polarization criterion values vary with the parameters $\alpha$ and $\sigma_2$. We can see that the polarization values have global maximum value. Fig. 1(b) shows polarization values linearly vary with $\alpha$. Fig. 1(c) shows polarization values vary with $\sigma_2$. Fig. 1(d), shows recognition rates vary with $\sigma_2$. In Fig. 1(c) and (d), we can see that their curves are similar. Recognition rate variation is approximately consistent with the polarization values. The results clearly show that learning the kernel using the polarization criterion is reasonable.

Fig. 1 has similar curves to Fig. 2; Fig. 2(a) shows polarization criterion values vary with the parameters $\alpha$ and $\sigma_2$. We found that the polarization values have global maximum value. In Fig. 2(b), when $\sigma_1$ and $\sigma_2$ are assigned with fixed values, we can see that polarization values linearly vary with $\alpha$. Fig. 2(c) and (d) show polarization values and recognition rates vary with $\sigma_2$. We can see that their curves are similar. Although the positions of the maximum values in the two figures are slightly different, the recognition rate variation is approximately consistent with the polarization values. The polarization criterion proposed by Baram [8] indicates that the greater the polarization criterion values, the better the classification performance of the SVM will be. That is, polarization criterion variation is approximately consistent with recognition rate. The experimental results verify the rationality of selecting parameters by the polarization criterion.

Fig. 3 indicates the classification performance of five different methods on the six sample sets. We can see that the proposed MSKL performs better than the other 4 methods in all cases. MSKL is a multi-scale multiple kernel learning method and the other four methods are single kernel methods. The multiple kernel learning method performs better than single kernel method in classification when the data sets are heterogeneous or in a non-flat distribution of samples. In addition, our method is more stable than the other four methods. Thus, the proposed method is effective for image classification.

Learning the kernel and selecting parameters by the optimal method in the classification problem is reasonable and effective. Additionally, the proposed combined kernel method is simple with low complexity. It is a very practical method for selection of parameters and learning the kernel.

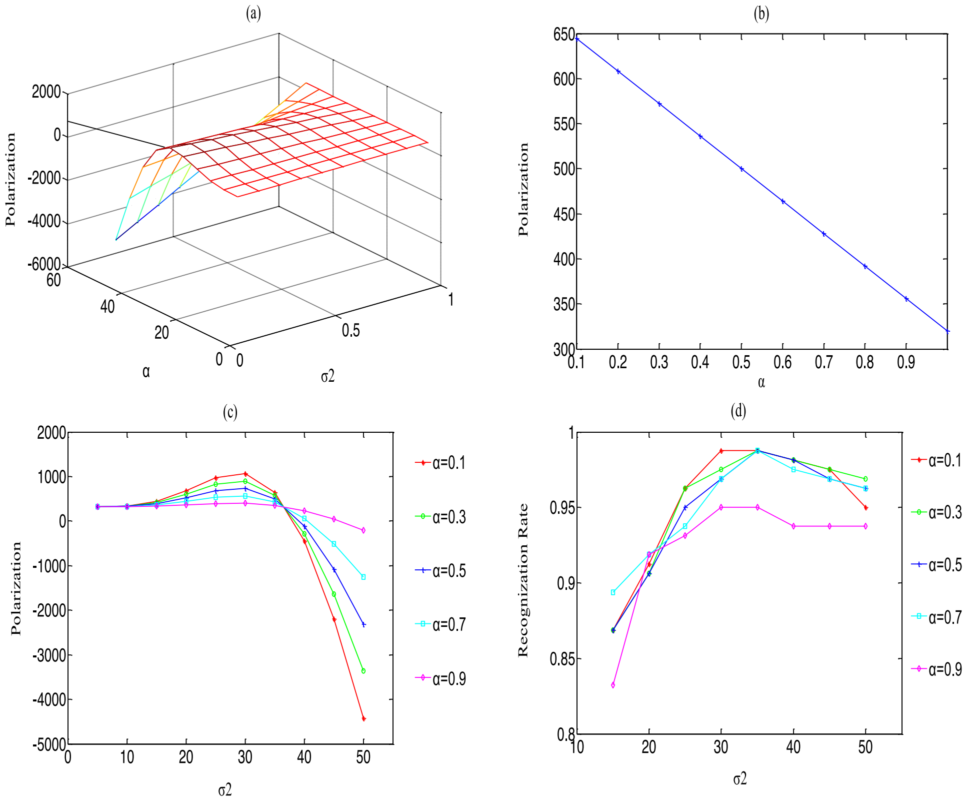


**Fig. 2.** The polarization values and the classification rates of the multi-scale kernel vary with parameters on image data set of Semeion. (a) Polarization values vary with parameter $\alpha$ and $\sigma_2$; (b) When $\sigma_2$ is 30, Polarization values vary with parameter $\alpha$; (c) Polarization value vary with parameter $\sigma_2$; (d) Recognition values vary with parameter $\sigma_2$.

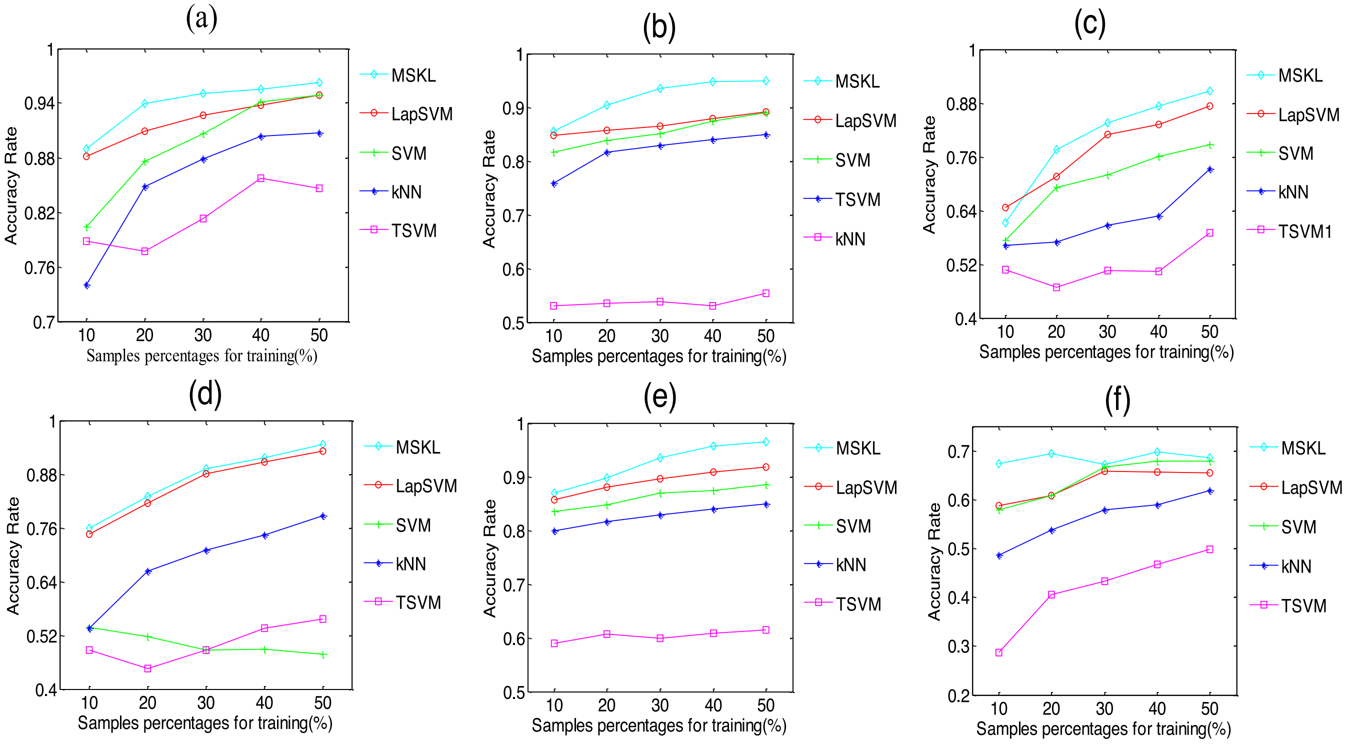
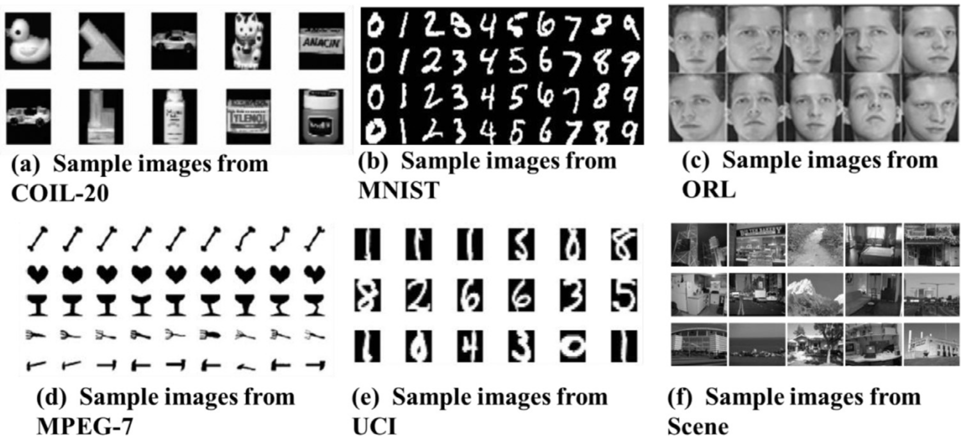


**Fig. 3.** Classification accuracy rates of 5 different methods (MSKL, LapSVM, SVM, TSVM, kNN) on 6 data sets (a) Object image samples COIL-20, (b) handwritten digit samples MNIST, (c) face image samples ORL, (d) shape image samples MPEG-7, (e) handwritten digit samples Semeion, (f) scene 15.


(a) Sample images from COIL-20

(b) Sample images from MNIST

(c) Sample images from ORL

(d) Sample images from MPEG-7

(e) Sample images from UCI

(f) Sample images from Scene

**Fig. 4.** Six image sample sets are used in our experiments. They are chosen from (a) Object image samples COIL-20, (b) handwritten digit samples MNIST, (c) face image samples ORL, (d) shape image samples MPEG-7, (e) handwritten digit samples Semeion, (f) scene 15.

## 6. Conclusions and further study

We proposed an optimal method using kernel polarization to learn a multi-scale kernel and proposed a grid algorithm. The optimal method shows that a higher recognition rate can be obtained by our method and algorithm. Additionally, the optimal method can be used to select parameters and learn the kernel at the same time. The multi-scale multiple kernel learning using our method has a better generalization and higher recognition rate than single kernel learning, because it can handle both samples with dynamic changes and samples with smooth changes.

In addition, the multi-scale kernel method is very practical and efficient and is worthy of further research. Researchers can find more multi-scale kernels as the base kernel to construct better multiple kernels with good generalization performance. That multi-scale multiple kernel methods can be used to solve many problems that single kernel methods can't handle well.

## Acknowledgments

We would like to thank all the people who help us. This research is supported by the Intelligence and control laboratory of Hangzhou Dianzi University.

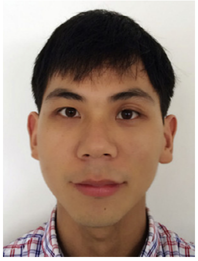
## References

[1] J. Shawe-Taylor, N. Cristianini, Kernel methods for pattern analysis [*M*], Cambridge University Press, UK, 2004.
[2] P. Honeine, Online kernel principal component analysis: A reduced-order model [J], IEEE Trans. Pattern Anal. Mach. Intell. 34 (9) (2012) 1814–1826.
[3] B. Schölkopf, J. Smola A, Learning With kernels: Support Vector machines, regularization, optimization, and beyond [*M*], MIT press, Cambridge United States, 2002.
[4] C. Cortes, V. Vapnik, Support-vector networks. [*J*], Mach. Learn. 20 (3) (1995) 273–297.
[5] O. Chapelle, V. Vapnik, O. Bousquet, et al., Choosing multiple parameters for support vector machines [*J*], Mach. Learn. 46 (1-3) (2002) 131–159.
[6] J. Paul, P. Dupont, Kernel methods for heterogeneous feature selection [*J*], Neurocomputing 169 (2015) 187–195.
[7] N. Cristianini, J. Kandola, A. Elisseeff, et al., On kernel-target alignment [M], Innovations in Machine Learning, Springer Berlin Heidelberg 194 (2006) 205–256.
[8] Y. Baram, Learning by kernel polarization. [*J*], Neural Comput. 17 (6) (2005) 1264–1275.
[9] M. Tian, W. Wang, An efficient Gaussian kernel optimization based on centered kernel polarization criterion [*J*], Inf. Sci. 322 (2015) 133–149.
[10] A. Zien, C.S. Ong, in: Proceedings of the 24th International Conference on Machine learning, ACM, 2007, pp. 1191–1198.
[11] S.O. Cheng, A.J. Smola, R.C. Williamson, et al., Learning the kernel with hyper kernels [*J*], J. Mach. Learn. Res. 6 (1) (2005) 1043–1071.
[12] G.R.G. Lanckriet, Nello Christianini, PeterL. Bartlett, et al., Learning the kernel matrix with semi-definite programming. [*J*], J. Mach. Learn. Res. 5 (1) (2002) 323–330.
[13] F.R. Bach, G.R.G. Lanckriet, M.I. Jordan, in: Proceedings of the twenty-first international conference on Machine learning, ACM, 2004, p. 6.
[14] S. Sonnenburg, in: Proceedings of the 18th International Advances on Neural Information Processing Systems, 2006, pp. 1273–1280.
[15] W. Luo, J. Yang, W. Xu, et al., Higher-level feature combination via multiple kernel learning for image classification [*J*], Neurocomputing 167 (2015) 209–217.
[16] Y. Gu, H. Liu, Sample-screening MKL method via boosting strategy for hyper spectral image classification [*J*], Neurocomputing 173 (2016) 1630–1639.
[17] N. Kingsbury, D.B.H. Tay, M. Palaniswami, in: Proceedings of the IEEE Workshop on Machine Learning for Signal Processing, IEEE, 2005, pp. 43–48.
[18] C. Cortes, M. Mohri, A. Rostamizadeh, Algorithms for learning kernels based on centered alignment [J], J. Mach. Learn. Res. 13 (2012) 795–828.
[19] T. Wang, S. Tian, H. Huang, et al., Learning by local kernel polarization [*J*], Neurocomputing 72 (13) (2009) 3077–3084.
[20] L. Wang, Feature selection with kernel class separability [J], IEEE Trans. Pattern Anal. Mach. Intell. 30 (9) (2008) 1534–1546.
[21] H. Xiong, M.N.S. Swamy, M.O. Ahmad, Optimizing the kernel in the empirical feature space [J], IEEE Trans. Neural Netw. 16 (2) (2005) 460–474.
[22] C.H. Nguyen, T.B. Ho, An efficient kernel matrix evaluation measure [J], Pattern Recog. 41 (11) (2008) 3366–3372.
[23] Nene S.A., Nayar S.K., Murase H. Columbia object image library (COIL-20) [*R*]. Technical report CUCS-005-96, 1996.
[24] Y. LeCun, L. Bottou, Y. Bengio, et al., Proceedings of the IEEE, in: Gradient-based learning applied to document recognition [J], 86, 1998, pp. 2278–2324.
[25] S. Sonnenburg, G. Rätsch, C. Schäfer, et al., Large scale multiple kernel learning [J], J. Mach. Learn. Res. 7 (2006) 1531–1565.
[26] J. Latecki L, Shape data for the MPEG-7 core experiment CE-Shape-1 [*J*]. http://www.cis.temple.edu/-latecki/TestData/mpegTshapeB.tar.gz, 2002 [2013-06-26].
[27] S. Lazebnik, C. Schmid, J. Ponce, Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories [C], in: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2, IEEE, 2006, pp. 2169–2178.
[28] A. Rakotomamonjy, F. Bach, S. Canu, et al., More efficiency in multiple kernel learning [C], in: Proceedings of the 24th International Conference on Machine Learning, ACM, 2007, pp. 775–782.
[29] R. Alain, R.B. Francis, C. Stephane, Simple MKL [J], J. Mach. Learn.Res. 58 (9) (2008) 1–34.
[30] D. Zheng, J. Wang, Y. Zhao, Non-flat function estimation with a multi-scale support vector regression [J], Neurocomputing 70 (1) (2006) 420–429.
[31] Z. Yang, J. Guo, W. Xu, et al., Multi-scale support vector machine for regression estimation [C], in: Proceedings of the International Symposium on Neural Networks, Springer, Berlin Heidelberg, 2006, pp. 1030–1037.
[32] B. Li, D. Zheng, L. Sun, et al., Exploiting multi-scale support vector regression for image compression [J], Neurocomputing 70 (16) (2007) 3068–3074.
[33] A. Pozdnoukhov, M. Kanevski, Multi-scale support vector algorithms for hot spot detection and modeling [J], Stoch. Environ. Res. Risk Assessment 22 (5) (2008) 647–660.
[34] T. Joachims, Transductive inference for text classification using support vector machines [C], ICML 99 (1999) 200–209.
[35] M. Belkin, P. Niyogi, V. Sindhwani, Manifold regularization: A geometric framework for learning from labeled and unlabeled examples [J], J. Mach. Learn. Res. 7 (2006) 2399–2434.
[36] S. Belongie, J. Malik, J. Puzicha, Shape matching and object recognition using shape contexts [J], IEEE Trans. Pattern Anal. Mach. Intell. 24 (4) (2002) 509–522.
[37] J. Wang, J. Yang, K. Yu, et al., Locality-constrained linear coding for image classification [C], in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 2010, pp. 3360–3367.
[38] A. Troncoso, M. Arias, C. Riquelme J, A multi-scale smoothing kernel for measuring time-series similarity [J], Neurocomputing 167 (2015) 8–17.

[39] J. Bi, T. Zhang, P. Bennett K, Column-generation boosting methods for mixture of kernels [C], in: Proceedings of the tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, 2004, pp. 521–526.

[40] F.S. Samaria, A.C. Harter, Parameterisation of a stochastic model for human face identification [C], in: Proceedings of the Second IEEE Workshop on Applications of Computer Vision, IEEE, 1994, pp. 138–142.

**Jian Bao** was born in Wenzhou, Zhejiang, China in 1962. She received the B.S. degree in automation from East China University of Science and Technology, Shanghai, in 1982 and the M.S. degree in computer science from Zhejiang University in 1999. She is a full professor in the Institute of Intelligent and Software Technology at Hangzhou Dianzi University. Her areas of research are related to intelligent control, neural networks, optimization algorithm and the embedded systems.



**Yangyang Chen** is a M.S student or a researcher in the Institute and Software Technology at Hangzhou DianZi University. He received his B.E. in applied mathematics from Xinyang Normal University in 2010 and 2014. His research interests include machine learning and artificial intelligence.

**Li Yu** is a M.S student at Wuyi University. She received her B.E. in Electronic and Information Engineering from Xinyang Normal University in 2010 and 2014. Her research interests include machine learning and Mobile Communications.

**Chunwei Chen** is a M.S student or a researcher in the Institute and Software Technology at Hangzhou DianZi University. He received his B.E. in Soft Engineering from Zhongshan Institute in 2010 and 2014. His research interests include machine learning and artificial intelligence.