Hand posture recognition based on heterogeneous features fusion of multiple kernels learning

Jiangtao Cao, Siquan Yu, Honghai Liu, Ping Li

Abstract As a rapid developing research topic in the machine vision field, image-based hand posture recognition has the potential to be an efficient and intuitive tool of human-computer interaction. For improving the accuracy of multi-class hand postures and extending the algorithm generalization,a novel hand posture recognition method is proposed by integrating the multiple image features and multiple kernels learning support vector machine(SVM). Firstly, three types of feature descriptors are extracted to describe the characteristics of a hand posture image. Shape context descriptor represents distribution characteristics of the edge points of the hand posture image. Pyramid histogram of oriented gradient describes characteristics of local and global shape effectively. The Bag of Feature(BOF) algorithm describes the surface texture characteristics of the posture image. Secondly, the Chamfer kernel and histogram intersection kernel are rebuilt to obtain the basis kernels of the features. And the combined kernel is constructed by weighting the basis kernels.So the heterogeneous features fusion realizes. Finally, the classification model and optimal fusion weights are calculated by using multiple kernels learning algorithm. The unknown category posture can be recognized by the trained multiple kernels of SVM. Experiments on Jochen Triesch's hand posture dataset demonstrate that the proposed method obtains higher recognition rate than the traditional single-kernel classifier and other recent methods.

Keywords Hand posture recognition · Heterogeneous feature fusion · Multiple kernel learning · SVM

1 Introduction

With the development of computer technology, some traditional human-computer interaction(HCI) technology, such as mouse, keyboard, cannot fully meet the needs of the people. Researchers pay more concern on seeking more effective interaction method. Because of its natural and intuitive interaction modality, the hand postures provide an attractive alternative to these cumbersome interface devices for HCI [19, 23]. Visual interpretation of hand postures can help in achieving the ease and naturalness desired for HCI. The main goal of hand posture recognition is to create a system which can identify specific human hand posture and convey information or for device control. The primary tasks of the hand posture recognition are to select the proper feature to describe the posture image and to design an efficient recognition algorithm with high recognition accuracy and less calculation cost.

Since hand posture is very rich in shape variation, feature selection is crucial to hand posture recognition. Many features have been applied to represent hand posture. Fang et al. utilized scale space as the feature to represent the hand posture image. The experiment demonstrated that the feature has a weak ability to distinguish similar gestures [16]. In the research of [15], shape context was used to extract the feature of hand posture image. The advantage of the feature is translation and scaling invariance, but it does not preserve rotation invariance. Ren et al. extracted gesture image features by using gradient direction histogram (HOG). The HOG has the ability of anti-noise and rotation, but it only describes local characteristics of image and cannot effectively represent the global information [24]. Wang et al. made use of Scale Invariant Feature Transform(SIFT) as feature to recognize hand postures. However the feature discards spatial information [32]. Different image features have various discriminative abilities. In order to improve the robustness of feature, image features can be described more comprehensive by using multiple feature fusion [25].

In recent years, there have been many recognition algorithms which are applied to hand posture recognition system. Support vector machine(SVM) is a kind of machine learning algorithm based on kernel function and statistical theory [29]. It maps the data to a high dimensional space, so that the data become linearly separable. Support vector machine has been well used in gesture recognition. A method for hand gesture recognition based on Bag of Feature (BOF) and multi-class SVM have been proposed in [14]. The experiments show that the system can achieve satisfactory real-time performance. Chen et al. present a multi-angle hand gesture recognition system for finger guessing games [10]. The system trained three SVMs by using images acquired from different cameras. Then the category of input gesture is determined by using the method of voting. The disadvantage of this method is that as species of gesture increases, the recognition rate will be significantly reduced.

As can be seen from the above literature, single feature and SVM classification algorithm is difficult to meet the needs of complex classification problems, especially for multi-source heterogeneous data classification [20]. In order to improve the generalization ability of SVM, the multi-kernel functions begun to attract extensive attention [3]. The Multiple Kernels Learning(MKL) algorithm learns the optimal kernel combination and the associated classifier simultaneously, providing an effective way of fusing informative features and kernels [22]. MKL has three advantages [18, 20]: (1) Multiple kernel fusion has better description ability of data features. (2) The generalization ability of multiple kernel fusion is stronger. (3) Multiple kernel fusion enhances the interpretability of the decision function. In recent years, there have been many multiple kernel learning achievements.Gehler et al. implement multi objective classification based on MKL [17]. Vedaldi et al. use a multiple

kernel learning method to achieve object detection, providing an effective way of fusing different types of features and kernels [31].

In this paper, a new hand posture recognition method is proposed by using MKL. The framework is designed as Fig. 1. Firstly three types of image features are extracted from the training set of hand posture image, *i.e.,* Shape Context (SC), Pyramid Histogram of Oriented gradient histogram (PHOG) and Bag of Feature (BOF). The basis kernel of three types of features is constructed by using Chamfer distance and histogram intersection kernel. Then the basis kernels are combined by using mixed-weighted linear summation with a product and a support vector machine model is trained by using a MKL algorithm to calculate the fusion weights. When an unknown category posture is input, the category of the gesture is discriminated by using three types of features and the fusion kernel function to implement gesture recognition. Since kernel weighted fusion can implement heterogeneous features fusion, that enhances generalization ability of support vector machine classification.

This paper is outlined as follows: in Section 2, the construction of basis kernel is given. First the definition of the three features of SC, PHOG and BOF are introduced. Then the Chamfer distance, histogram intersection kernel are chosen to construct the basis kernel. In Section 3, the basis kernels combination method is introduced. The MKL algorithm is listed to calculate the optimal combined weights in Section 4. In Section 5,the proposed method is evaluated on Jochen Triesch's hand posture dataset, and compared with other posture recognition methods. The conclusion of this paper with suggestions for further research is presented in Section 6.

2 Feature extraction and basis kernels construction

In hand posture recognition system, robust and high discriminative feature descriptor of the hand gesture images is a key ingredient to accurate posture recognition. In order to

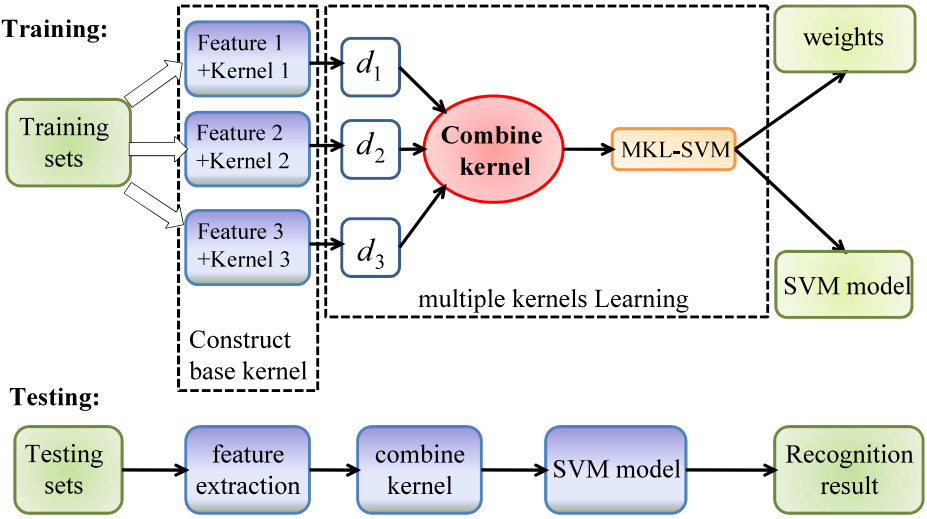

Fig. 1 The framework of proposed hand posture recognition algorithm

distinguish different hand postures effectively, three types of image features(SC, PHOG and BOF) are chosen. The SC feature is a rich descriptor of the sampled boundary points and offers a globally discriminative characterization. The PHOG feature can characterize the local and global shape features effectively. The BOF feature is a local descriptor of the appearance of the object in the image, and it can capture the local spatial of gradients within an image. These features can respectively represent the edge, shape, appearance characteristic of the image. So the combination of those three features can greatly improve the discriminative ability of the image feature. On the basis of the feature selection, the basis kernels are constructed by using the Chamfer distance and histogram intersection kernel.

2.1 Shape context feature and points kernel

Shape Context(SC) proposed by Belongie et al. [5] is robust to change of photometric properties and offers a globally discriminative characterization. The basic idea of the algorithm is to describe the shape information by sampled edge points. Firstly, the edge of the hand posture image is extracted and sampled to generate several sampling points. Then, the SC descriptor of each sampling point is calculated and represented by a log-polar histogram in a given polar coordinate. Finally the SC descriptors of all sampling points are combined to generate the SC descriptor of the hand shape. The feature extraction process is shown in Fig. 2. It can be seen that the SC features computed for two similar positions of two similar hand postures are similar.

The distance between SC descriptors of two gesture images can be calculated using *Chamfer* distance [22]. If the two posture images are represented by feature vectors **x** and

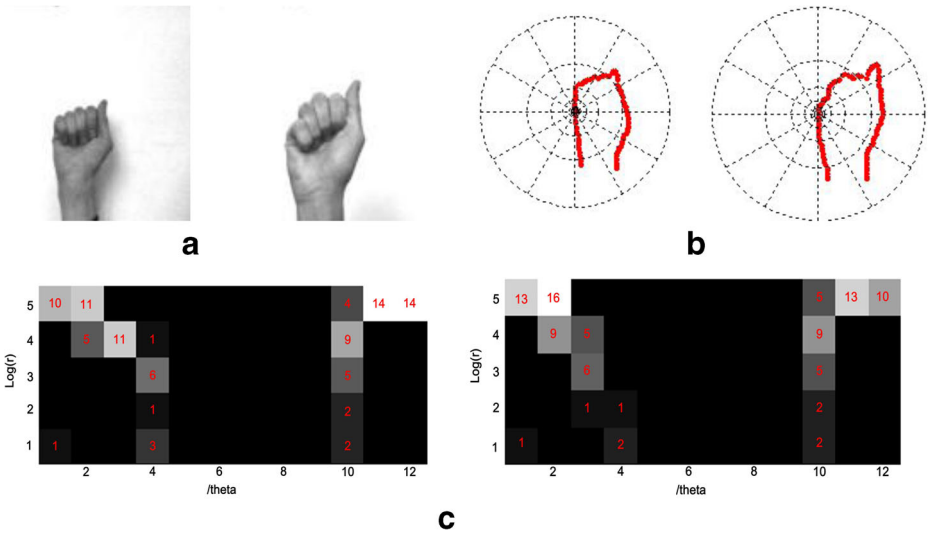

Fig. 2 The SC feature extraction of the hand posture. **a** Two images from the same category hand posture. **b** Diagram of log-polar histogram bins is used to compute the SC features. **c** Extracted feature on the 20-th sampling point in (**b**)

y respectively, the *Chamfer* distance between them is the average distance from the nearest descriptors pairs, as follows:

$$Chamfer(\mathbf{x}, \mathbf{y}) = \frac{1}{m} \sum_{i=1}^{m} \min_{\mathbf{y}_i} \left\| \mathbf{x}_i - \mathbf{y}_j \right\| \tag{1}$$

Where m is the number of sampling points in the hand posture image; $\mathbf{x}_i$ represents a feature vector in $\mathbf{x}$ corresponding to histogram representation of the i-th sampling point. $\mathbf{y}_j$ represents a feature vector in $\mathbf{y}$ corresponding to histogram representation the j-th sampling point.

The *Chamfer* distance is a symmetrical in the way which it is expressed. So it would lead to non-symmetric kernels, which means that it can not be directly used in a kernel. In this way, we use $g(\mathbf{x}, \mathbf{y}) = Chamfer(\mathbf{x}, \mathbf{y}) + Chamfer(\mathbf{y}, \mathbf{x})$ to solve the symmetry problem [22]. Finally, Point Kernel is computed as follows:

$$K_{point}(\mathbf{x}, \mathbf{y}) = g(\mathbf{x}_{sc}, \mathbf{y}_{sc}) \tag{2}$$

Where $\mathbf{x}_{sc}$ and $\mathbf{y}_{sc}$ are the SC descriptors of hand posture images.

2.2 Histogram of oriented gradient feature and shape kernel

Histogram of Oriented Gradient (HOG) was originally proposed by Dalal [13]. The HOG is a shape descriptor. It describes image features by calculation and statistical histogram of oriented gradient of local area in image. The HOG feature and SVM classifier have been widely applied in image recognition, especially in the pedestrian detection achieved great success. HOG feature extraction algorithm is usually divided into the following three steps: (1) Detect object edge in the image by Canny algorithm. (2) The image is divided into many sub-blocks, and then the gradient on edge points is calculated in each block. (3) Statistics the gradient histogram of the whole image as the image feature descriptor.

When the two images are described, shape kernel can be calculated by histogram intersection kernel [4]. The histogram intersection kernel of two images is defined as follows:

$$K_{int}(\mathbf{x}, \mathbf{y}) = \sum_{i=1}^{m} \min \left\{ \mathbf{x}_i, \mathbf{y}_j \right\} \tag{3}$$

where $\mathbf{x}$ and $\mathbf{y}$ is the histogram feature of the image X_{im} and Y_{im}, m is the block number in image. x_i, $y_i (i = 1, 2, , m)$ are the values in the ith block of histogram $\mathbf{x}$ and $\mathbf{y}$. The advantage of histogram intersection kernel is to improve the classification accuracy and needn't select parameters. Shape kernel is calculated as follows:

$$K_{shape} = K_{\text{int}} \left(\mathbf{x}_{hog}, \mathbf{y}_{hog} \right) \tag{4}$$

where $\mathbf{x}_{hog}$ and $\mathbf{y}_{hog}$ are HOG descriptors.

In order to describe the spatial orientation at different resolutions, we adopt four-level image pyramid for shape kernel. HOG descriptors and shape kernels are respectively generalized on each level of image pyramid to achieve the Pyramid HOG(PHOG) descriptor [7]. The extraction process of PHOG feature is shown in Fig. 3.

To illustrate the necessity of construction image pyramid the correlation of HOG descriptors and PHOG descriptors of two posture images using histogram intersection kernel are

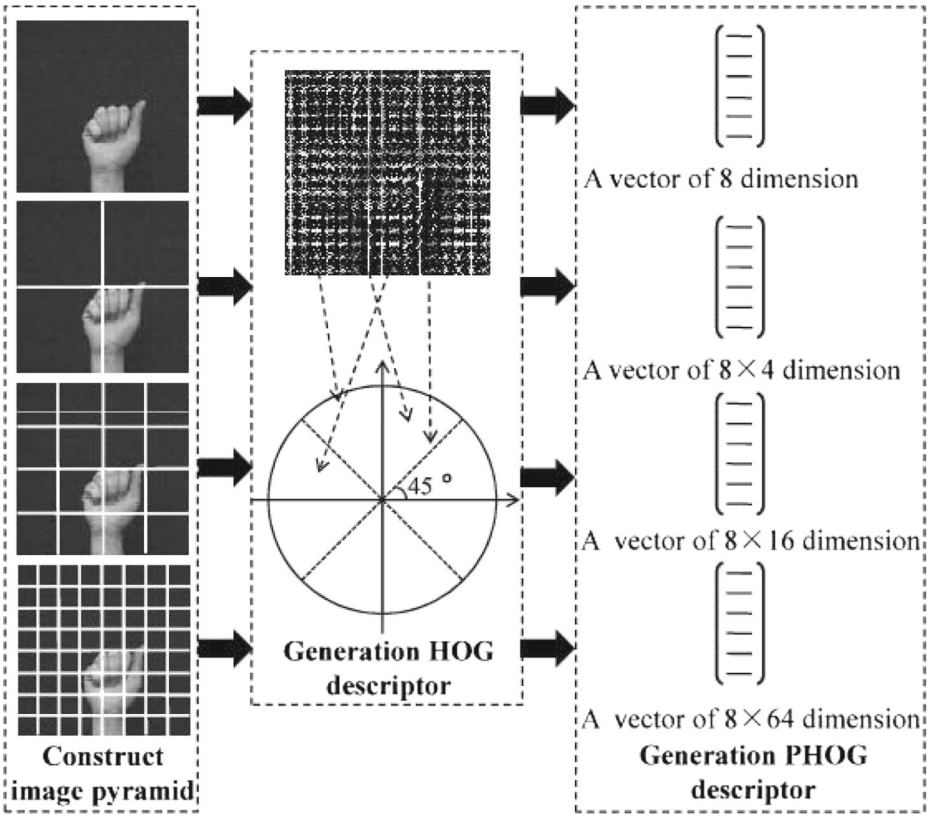

Fig. 3 The extraction process of PHOG feature for describing a hand posture

compared and shown in Fig. 4. The upper right corners of the Fig. 4a and b are the input posture images. The blue bar is the correlation using PHOG descriptor after combining each level, and the purple bar represents correlation when using HOG without image pyramid construction. We can see that PHOG descriptors represent hand postures in different degree of details according to various pyramid levels, they have a better ability to describe the characteristic of the images.

2.3 Bag of feature and appearance kernel

BOF algorithm describes the local distribution of gradients within an image [11]. BOF Calculation usually goes through three steps: feature point detection, feature description, codebook and descriptors generation. Firstly, the image is divided into many sub blocks. Then, the Scale-invariant feature transform (SIFT) descriptor is extracted in each block. Finally, all the descriptors from training data are clustered by a clustering algorithm and these K clustering centers are chosen as a visual codebook. Each visual word in the codebook represents a small similar patch of images. All the SIFT descriptors in a given image can be mapped to the visual vocabulary by using Euclidean distance as the criterion and generate statistics histogram as feature vector. In this way, each hand posture image can

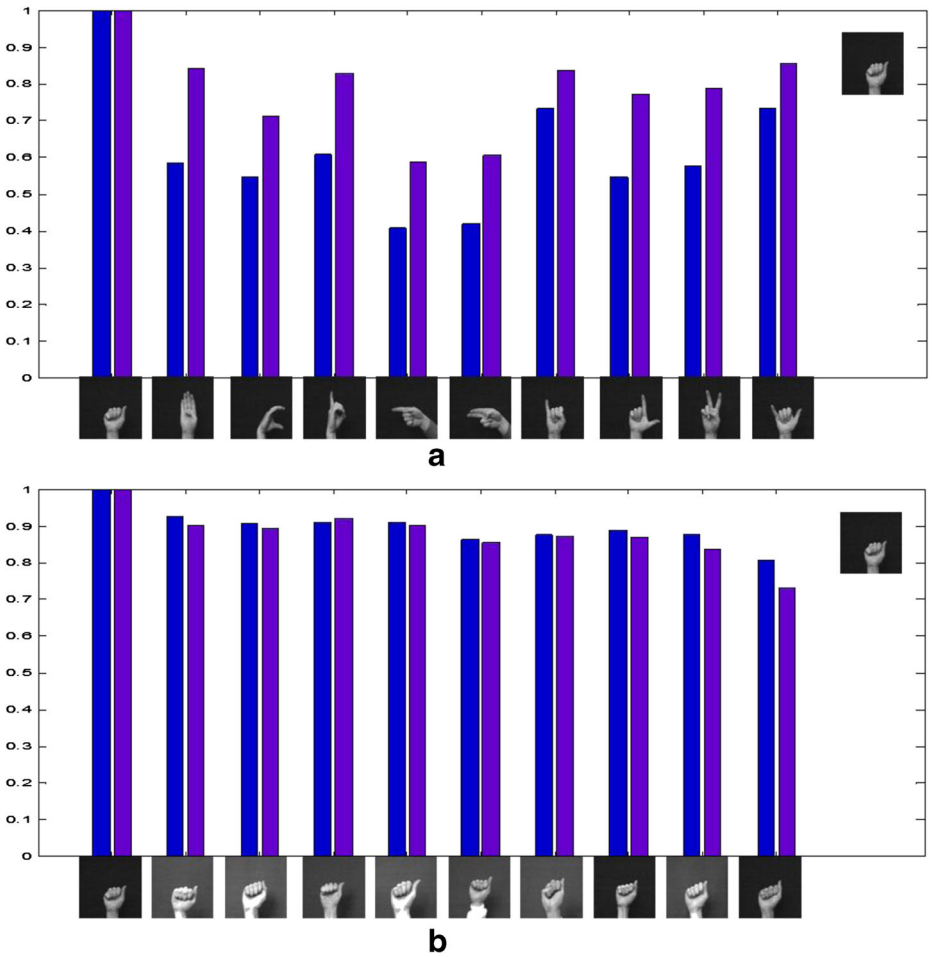

Fig. 4 The comparison the correlation of HOG descriptor and PHOG descriptor **a** Comparison the correlation of HOG descriptor and PHOG descriptor of two different gesture images. **b** Comparison the correlation of HOG descriptor and PHOG descriptor of two same gesture images

be represented by a vector. In this paper, we use K-means++ [1] algorithm to improve the traditional BOF algorithm. K-means++ algorithm chooses the initial cluster centers with a principle of maximum distance. Compared with the K-means algorithm, K-means++ algorithm has a advantage of better stability. According to the given parameter K, the k=means++ method can cluster the better clustering centers. The BOF extraction is shown in Fig. 5.

Since the BOF feature is described by histograms, the correlation of two images can be measured by histogram intersection kernel. The calculation formula of appearance kernel is as follows:

$$K_{app} = K_{\text{int}}(\mathbf{x}_{bof}, \mathbf{y}_{bof}) \tag{5}$$

where $\mathbf{x}_{bof}$ and $\mathbf{y}_{bof}$ are BOF descriptors.

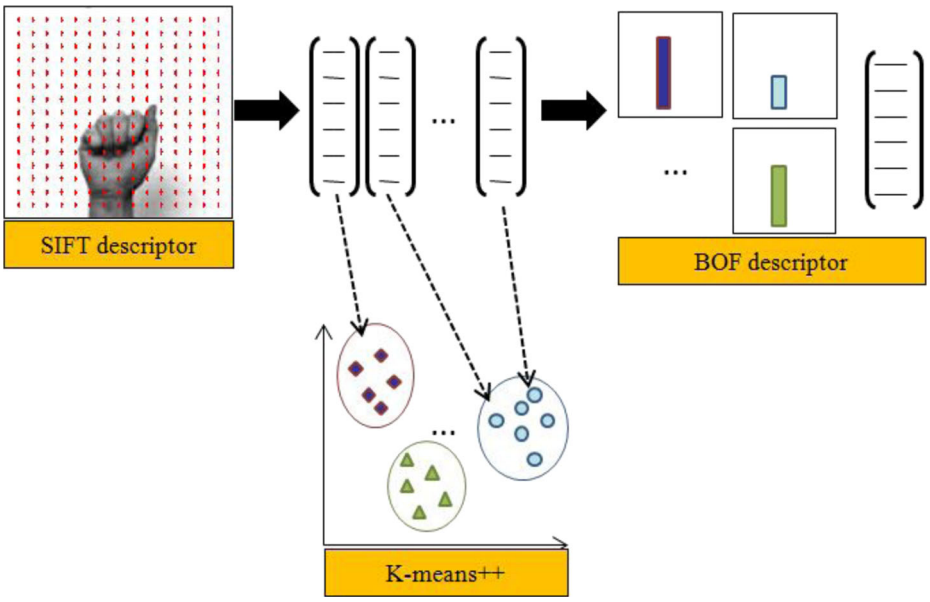

Fig. 5 The BoF feature computation for describing a hand posture

There are three types of feature are used (including SC,PHOG,BOF). And a four-level spatial pyramid for the PHOG kernel is used. Then there are six basis kernels in the proposed method.

3 Basis kernel combination method

Once the basis kernel is constructed, the combination way of the basis kernels becomes the main task. Kernel matrices reflect similarity between the samples. However, only those distance functions that fulfill Mercer's theorem are permitted as the valid kernels.

Some properties of Mercer's kernels which are relevant for this paper are the following.

$K_1(\mathbf{x}, \mathbf{y})$ and $K_2(\mathbf{x}, \mathbf{y})$ are two Mercer's kernels, where $\mu > 0$. Then, the following kernels are valid Mercer's kernels.

$$K(\mathbf{x}, \mathbf{y}) = K_1(\mathbf{x}, \mathbf{y}) + K_2(\mathbf{x}, \mathbf{y}) \tag{6}$$

$$K(\mathbf{x}, \mathbf{y}) = \mu K_1(\mathbf{x}, \mathbf{y}) \tag{7}$$

In this way, a new kernel function is obtained by direct summation or multiplication of the (weighted) kernels. The combination way of basis kernels can be divided into two steps. Firstly, the basis kernels of the same feature in multi-layer image pyramid are combined. Secondly, the basis kernels of the different features are combined. Many combination methods have been described [6, 8]. According to Mercer's theory, kernel of two different features can be combined by linear weighted method. Then the task is focus on learning linear combination weights of given basis kernels [28].

For using the extracted feature to describe the image spatial characteristics, four-level image pyramid and four basis kernels are adopt. Then the basis kernels are combined in different levels of Pyramid. The dimension of feature vectors in different levels is different, so they are heterogeneous features. The heterogeneous features in dimension can be unified by calculating the basis kernel. The combined shape kernel in different level is represented as:

$$K_{shape}(\mathbf{x}, \mathbf{y}) = \sum_{l=1}^{m} \gamma_l K_{shape}^{(l)}(\mathbf{x}, \mathbf{y}) \tag{8}$$

where γ_l is the weight of level l and is constrained to be positive ($\gamma_l > 0$). $K_{shape}^{(l)}(\mathbf{x}, \mathbf{y})$ is the basis kernel of l-th level. m is the number of image pyramid.

Once the basis kernel of different types of descriptor is calculated, the weighted summation version to combine different descriptors is gotten.

$$K_{opt}(d, \gamma) = \sum_{f=1}^{n} d_f K_{fopt}(\mathbf{x}_f, \mathbf{y}_f) \tag{9}$$

$$\begin{aligned} K_{opt}(d, \gamma) &= d_1 K_{point}(\mathbf{x}, \mathbf{y}) + d_2 K_{shape}(\mathbf{x}, \mathbf{y}) + d_3 K_{app}(\mathbf{x}, \mathbf{y}) \\ &= d_1 K_{point}(\mathbf{x}, \mathbf{y}) + d_2 \sum_{l=1}^{3} \gamma_l K_{shape}^{(l)}(\mathbf{x}, \mathbf{y}) + d_3 K_{app}(\mathbf{x}, \mathbf{y}) \end{aligned} \tag{10}$$

where K_{opt} is the fusion kernel. K_{fopt} is the basis kernel of f-th feature. d_f is the fusion weight. n is the number of basis kernels. Thus, the fusion weight calculation is the key to solve the fusion kernel. This problem could be solved by multiple kernels learning algorithm.

4 Weights learning

4.1 The traditional SVM algorithm

Support vector machine classification principle is to maximize distance between all the different samples of geometric. This problem can be solved by using the following optimization problem.

$$\begin{aligned} &\min_{w,b,\xi_i} \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{N}\xi_i \\ &s.t. y_i[\langle\varphi(x_i), w\rangle + b] \geq 1 - \xi_i, i = 1, 2, ..., N \\ &\xi_i \geq 0, i = 1, 2, ..., N \end{aligned} \tag{11}$$

where w defines the optimal classification hyperplane, $\langle\cdot\rangle$ represents inner product. b is bias of hyperplane. The parameter C determines the ability of the classifier regularization, which is complexity of classifier. ξ_i is a slack variable which describes the fault tolerance capability of classifier. This problem can be solved by its dual problem. The decision function for any given test vector $\mathbf{x}$ is as follows:

$$f\left(x, a^*, b^*\right) = \operatorname{sgn}\left(\sum_{i=1}^{n} y_i a_i^* K\left(x_i, x\right) + b^*\right) \tag{12}$$

where $x_i (i = 1, \ldots, n)$ are support vectors, $a_i^* (i = 1, \ldots, n)$ are Lagrange coefficients. $y_i (i = 1, \ldots, n)$ are labels of categories. b^* is the bias of classification hyperplane. $K(x, y) = \langle \varphi(x), \varphi(y) \rangle$ is the kernel function.

4.2 Multiple kernels learning

The core problem of multiple kernel learning is to calculate the combined kernel weights with the training data. The input posture of unknown category will be predicted to right category by using the final combined kernel function and the optimal classification hydroplanes. A commonly used method is grid search [21]. This method finds the optimal weights by using the weight exhaustive method in a given range. The disadvantage of this method is time consuming. Here, we adopt a method based on the method of Varma [30]. This method solves the problem of support vector machine optimization problem of multiple kernels by using minimax strategy. The combined kernel function is looked as a single kernel functions to train support vector machine.

In single kernel support vector machine, minimax optimization primal problem is the form as (9). Here, the problem is rewritten as: $\min_{(d_f, \gamma_l)} T(d_f, \gamma_l)$ Subjecting to $d_f \geq 0$ and $\gamma_l \geq 0$; $T(d_f, \gamma_l)$ is

$$T(d_f, \gamma_l) = \begin{cases} \min\limits_{w,b,\xi_i} \frac{1}{2} \|w\|^2 + C \sum\limits_{i=1}^{N} \xi_i \\ s.t. y_i [\langle \varphi(x_i), w \rangle + b] \geq 1 - \xi_i, \\ \xi_i \geq 0, i = 1, 2, ..., N \end{cases} \tag{13}$$

According to the theory of nonlinear programming, the problem can be solved by using the gradient descent method. The decreasing speed of the function value is the fastest along the orientation of negative gradient, so selecting the gradient ∇T as the descent direction of the function. In order to calculate the value of ∇T , the problem is converted into dual problem, which is as follows:

$$W(d_f, \gamma_l) = \begin{cases} \max\limits_{a} -\frac{1}{2} \sum\limits_{i,j} a_i a_j y_i y_j K_{opt}(x_i, x_j) + \sum\limits_{i} a_i \\ s.t. 0 \leq a_i \leq C \sum\limits_{i} a_i y_i = 0 \end{cases} \tag{14}$$

Derivative T on γ_l and d_f are equivalent to derivative W on γ_l and d_f.

$$\begin{aligned} \frac{\partial T}{\partial \gamma_l} &= \frac{\partial W}{\partial \gamma_l} = -\frac{1}{2} d_2 a^{*T} \frac{\partial (Y K_{shape} Y)}{\partial \gamma_l} a^* \\ &= -\frac{1}{2} d_2 a^{*T} Y K_{shape}^{(l)} Y a^* \end{aligned} \tag{15}$$

$$\frac{\partial T}{\partial d_f} = \frac{\partial W}{\partial d_f} = -\frac{1}{2} a^{*T} \frac{\partial (Y K_{opt} Y)}{\partial d_f} a^* = -\frac{1}{2} a^{*T} Y K_{fopt} Y a^* \tag{16}$$

where K_{fopt} is the kernel matrix for every feature; Y is a diagonal matrix of the label. Once the gradient of T is computed, the values of γ_l and d_f (f=1,2,3) can be calculated by the method of gradient descent while W obtains maximum. In the whole process, we firstly

train γ_l on fixed and initialized d_f . Then train d_f on the fixed γ_l in the same way. The iteration step of gradient descent method determines by the method of optimal step size. The implementation process of obtaining fusion kernel weights is as follows:

Input

SC,PHOG,BOW descriptors

Chamfer distance and histogram intersection kernel

Initialization

basis kernel weights $d_f = 1, \gamma_l = 1$

for class=1 to $Num\_classes$ do

1: $n = 0$ where $n = iteration\ number$

2: $\gamma_l = 1, d_f = 1$

Repeat,

3: Use (10) to construct a combined kernel K_{opt}.

4: Use the theory of single kernel support vector machine to train the model and calculate Lagrange coefficient matrix $a*$.

5: Updata the weights:

$$\gamma_f(n+1) = \max\left[0, \gamma_f(n) - \lambda_n \frac{\partial T}{\partial \gamma_f}\right]$$
$$= \max\left[0, d_f(n) + \frac{\lambda_n}{2} d_2 a^{*T} Y K^{(l)}_{shape} Y a^*\right]$$

where λ_n is the step of gradient descent, calculated by optimal step method

6: n=n+1

until convergence

ends

Once the values of γ_l and d_f are calculated, the combined kernel K_{opt} can be ascertained. A new vector$x$ can now be classified by the decision function as follows:

$$f\left(x, a^*, b^*\right) = \operatorname{sgn}\left(\sum_{i=1}^{n} y_i a_i^* K_{opt}\left(x_i, x\right) + b^*\right) \tag{17}$$

5 Experiments

5.1 Database

In order to verify the effectiveness of the algorithm, we evaluated the method for gesture recognition on Triesch gesture database [26]. . The database contains 10 kinds of gesture which represent A, B, C, D, G, H, I, L, V, and Y respectively, performed by 24 different people in different background. The backgrounds of the image for each person are of three types: uniform light, uniform dark and complex. Among the 720 images, two were lost by Triesch. Total number of the images is 718. Image pixel value is 128 × 128. Some sample images in the database are shown in Fig. 6. In this paper, the LibSVM toolbox [9] is used to solve the parameters of support vector machine (SVM).

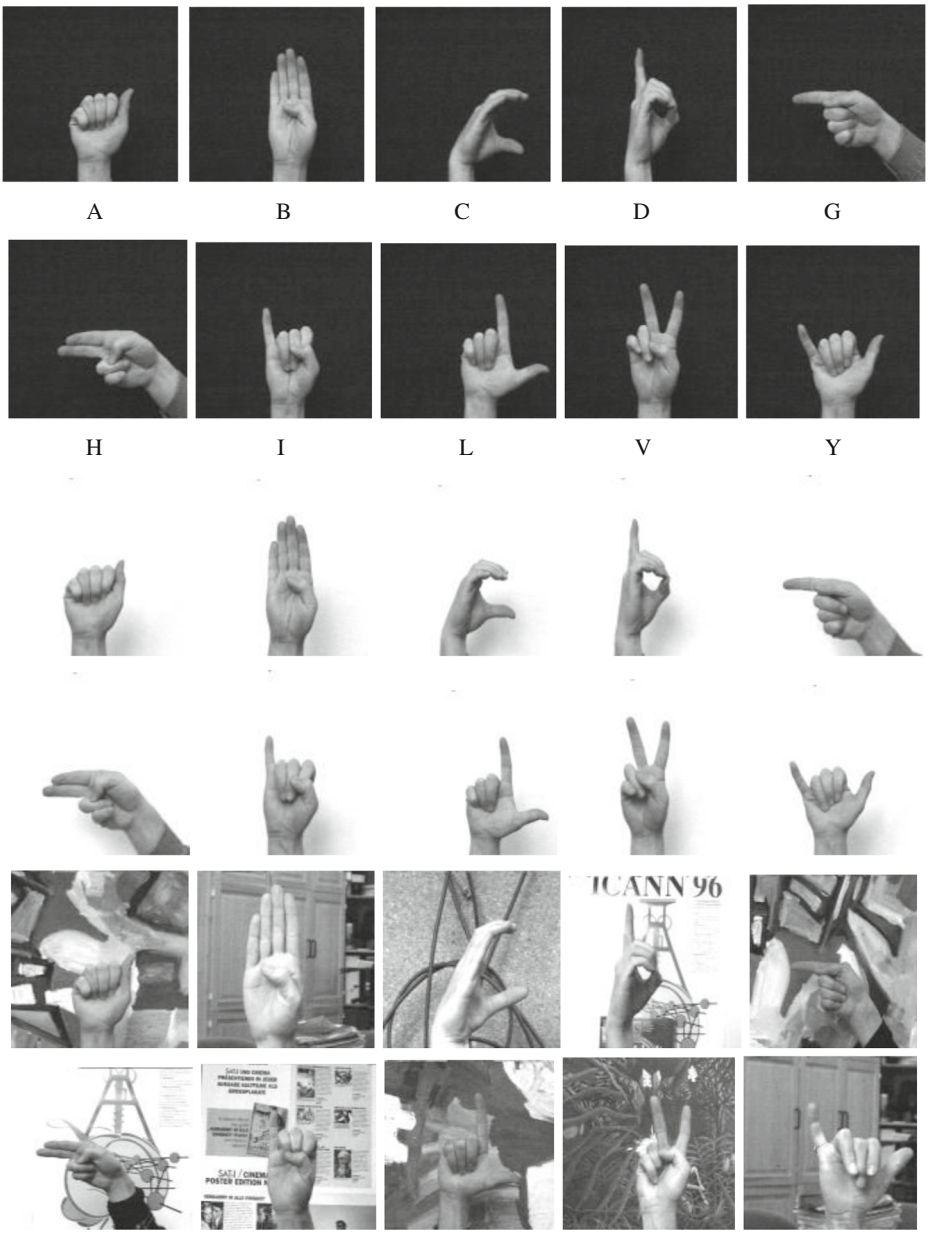

Fig. 6 The samples of the images from Jochen Triesch's database

5.2 Recognition rate of uniform background

We first take expert on multiple kernels and single kernel recognition rate comparison experiment on uniform background gesture images. The pictures in uniform light and uniform

Table 1 The Recognition rate(%) of single kernel and multiple kernel in uniform background

Training Number	SC	HOG	BOW	PHOG	MKL
120	69.71	77.08	76.25	80.83	81.25
140	72.50	79.17	80	83.33	84.17
160	75	81.67	83.75	85.42	87.50
180	77.50	83.75	84.17	88.75	92.92
200	80	85.83	85.83	92.08	95.83
220	82.08	86.25	86.25	95.83	99.17
240	82.50	86.25	87. 5	95.83	99.17

black background were selected in this experiment. When computing basis kernels, SC features are computed on 100 sampled points per image. For log-polar histogram, we adopt 5 and 12 bins for log$r$ and θ , respectively. The level of pyramid of PHOG feature is four, and the number of blocks in each level are 64, 16, 4 and 1. The HOG descriptor is discretized into 8 orientation bins. So the dimensions of descriptor in each level are 512, 128, 32 and 8 respectively. For Bow, we construct a codebook with 100 visual words. With the number of training samples change, the recognition rate will change. The numbers of training sample are chosen as 120, 140, 160, 180, 200, 220 and 240 respectively. The number of test images is 240. The results of recognition rates are shown in Table 1, and the recognition rate curve and confusion matrix are shown in Fig. 7.

Figure 7 reports the average recognition rate of hand posture image with uniform background when the number of training samples changes. The x-axis gives the number of

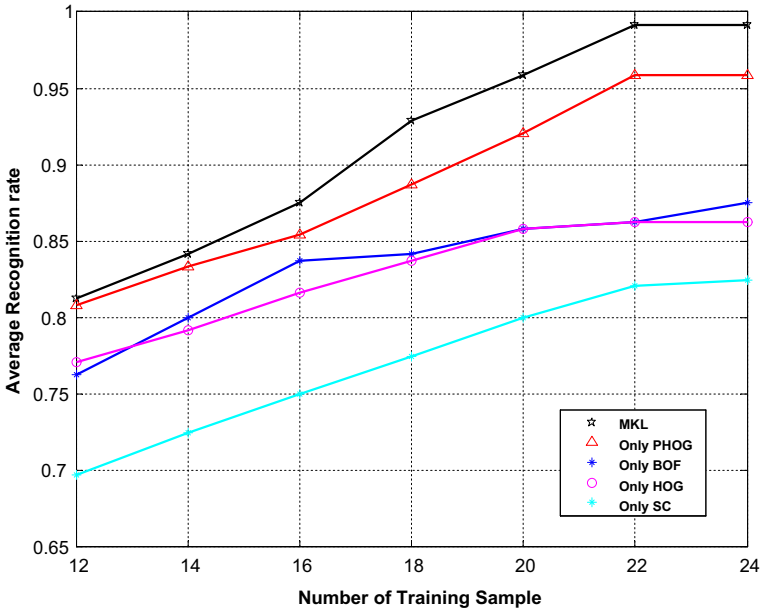

Fig. 7 The recognition rate comparison between MKL and single kernel in uniform background

training samples and the y-axis represents the recognition rate. The black line shows the performance of our method. Red, blue, purple and cyan carves are corresponding to the recognition rates of single kernel of PHOG, BOW, HOG and SC respectively. We can find that all kinds of single kernels can obtain average recognition rate of more than 80 % when the number of training samples increases to 22 per class. With the changes of the number of training samples, the recognition rate of MKL method is always the highest. In the single kernel methods, shape kernel works best, the recognition rate is 95.83 % when the training number is 240. In addition, point kernel and appearance kernel are also useful and can obtain a performance of more than 82.08 %. The recognition rate can reach up to 99.17 % when our method is used. The proposed method can improve the recognition rate significantly. The confusion matrix of our method is show in Fig. 8, when the number of training samples is 240.

5.3 Recognition rate of complex background

Besides the experiments in uniform background samples, we also use the posture image with the complex background. In our method, no any special operation is performed when the features are extracted in complex background. In order to describe more detail the experimental parameters are changed as follows. The sample of SC algorithm for edge sampling variation is taken as 150. The parameters of PHOG descriptors are not changed. The number of clusters BOW descriptor is changed to K=200. The numbers of training samples are

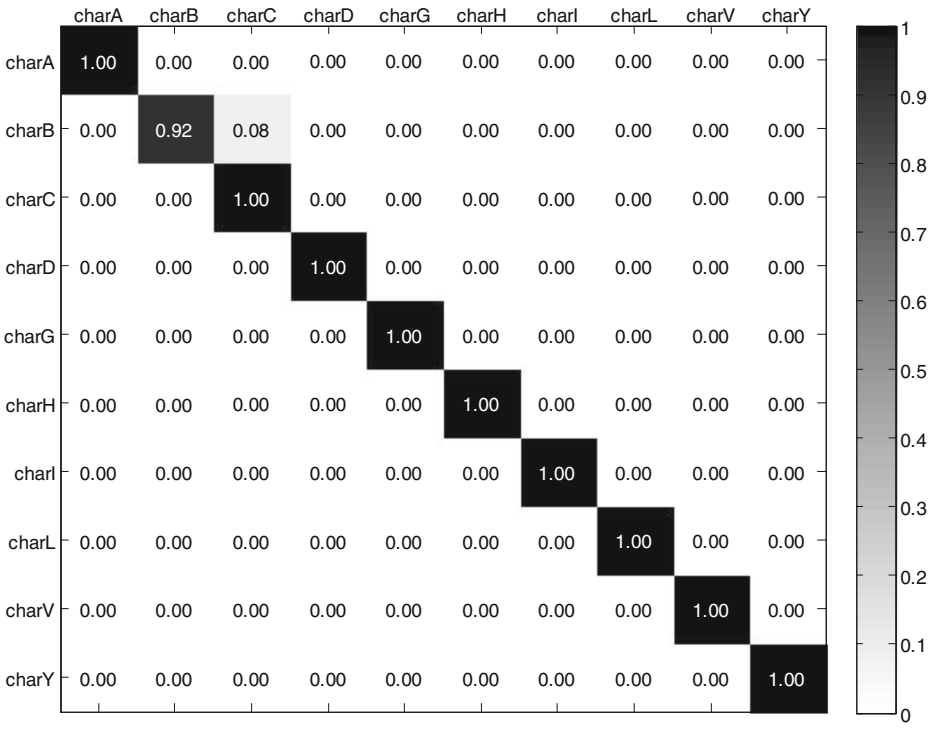

Fig. 8 The confusion matrix when the number of training sample is 240

Table 2 The Recognition rate in complex background

Training Number	Training sample ratio	Recognition rate
180	25 %	59.83 %
200	28 %	62.12 %
220	31 %	65.08 %
240	33 %	68.38 %
260	36 %	70.64 %
280	39 %	74.02 %
300	42 %	76.84 %
320	45 %	79.61 %
340	47%	82.43 %
360	50 %	83.02 %

taken as 180, 200, 220, 240, 260, 280, 300, 320, 340 and 360 respectively. The number of test sample is 358. Because the background is very complex, different training sample will lead to a significant influence on recognition results. Therefore, we need to randomly extract different training samples from the database in 10 experiments. In the process of extraction of training samples, the ratio of the number of complex background sample and uniform background sample is 1:2. Carve of the recognition rate is shown in Table 2 and Fig. 9. In Fig. 9, the light blue line represents the recognition rate change curve with different training samples, thick blue line is the average recognition rate curve. As can be seen

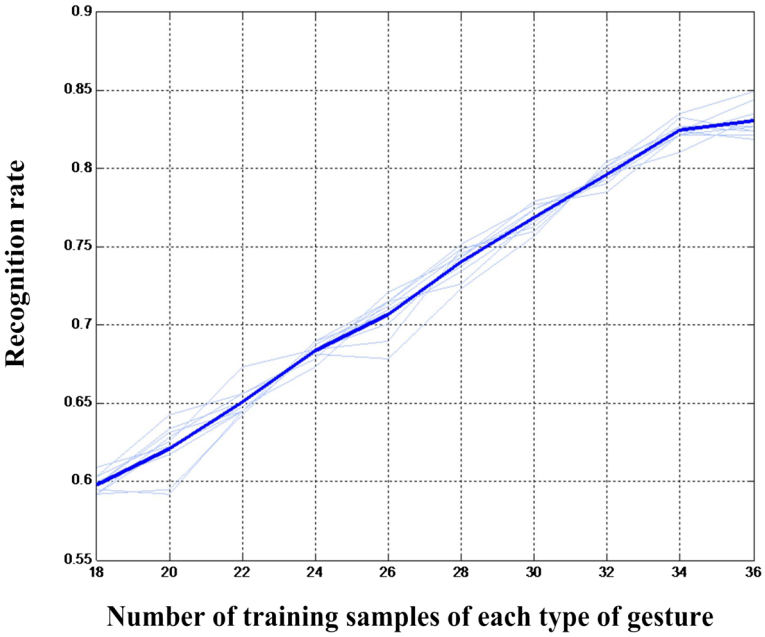

Fig. 9 Recognition rate curve of complex background with different training number

Table 3 The comparison of run times of different methods

Method	Runtime of uniform (s) background		Runtime of complex background (s) background	
	Training	Testing	Training	Testing
Only shape kernel	1.140585	0.422193	1.328581	0.453046
Only point kernel	0.503738	0.290238	0.636884	0.303003
Only appearance kernel	0.291132	0.202385	0.328527	0.230499
Our method	4.893025	0.660427	6.780336	0.763643

from the graph, the final average rate of proposed recognition method tends to be stable and the average recognition rate reaches 83.02 %.

5.4 Comparisons of efficiency

In order to test the efficiency of proposed method, the experiment was carried out to test the running time of the algorithm. And the results are compared with the single kernel methods. Test images with uniform background and complex background are chosen from Triesch's database. The number of training images are 240, 360 respectively, and the number of testing images are 240 and 358 respectively. The experimental parameters utilized the optimal parameters obtained in the above experiment. The experiment was performed on a PC with a CPU of 3.0GHz and 2-GB memory and by using matlab2009 software. Table 4 lists the run times for training and testing (Table 3).

This is directly caused by the corresponding kernel-function type and algorithm. Because the training time includes the kernel matrix time and training support vector machine time. Although point kernel is the most complex, but shape kernel method need to calculate the weights of different layers of PHOG descriptors. This directly leads to shape kernel method consuming the longest time. Comparison of four methods for testing time can be seen that the testing time of MKL is slightly higher than the other three kinds of single kernel methods. This is due to the fact that the computation and combination base kernel cost a

Table 4 Comparison with related works in recent years

Literature	Method	Recognition rate of uniform background	Recognition rate of complex background
Triesch [27]	elastic graph matching	90 %	–
Yuan [33]	PCA+Garbor,SVM	91.5 %	–
Agnes [2]	MCT feature+AdaBoost	89.97 %	81.25 %
Zhang [34]	Compressive Sensing+Zernike	92.1 %	–
Chuang [12]	BoF+Spectral-HIK	99.97 %	80.01 %
Single method1	Point kernel+SVM	82.50 %	68.89 %
Single method1	Apperance kernel+SVM	86.25 %	75 %
Single method1	Shape kernel+SVM	95.83 %	78.61 %
Our approach	MKL	99.17 %	83.02 %

small amount of time. However, the increase in testing time for hand posture recognition is negligible.

5.5 Comparison with other related methods

In comparison with the other methods which used Jochen Triesch database recently, the results are shown in Table 4. In addition, we also designed an experiment to compare the recognition performance between single kernel method and multiple kernel method in complex background. The comparison results are shown in Table 4. The results showed that the proposed hand posture recognition method based on multiple kernel outperformed the single kernel method. With the same testing condition, the recognition rate was increased from 68.89 % to 83.02 %. Furthermore, the proposed hand posture recognition method achieved better recognition rate than other recent methods in both complex background and simple background.

6 Conclusion

In this paper, a novel hand posture recognition algorithm is proposed. In feature extraction process, three types of image features are extracted: shape context, pyramid histogram of oriented gradient, bag of features. The three kinds of heterogeneous features are combined by the proper kernel function to construct basis kernels of the features. Then, the basis kernels are fused. The value of the fusion weights are obtained by using multiple kernels learning algorithm. The multiple features and multiple kernels fusion improve the generalization ability of support vector machine. With the experiment results and the comparison with other related works of hand posture recognition algorithms, the proposed method improved the recognition accuracy under the simple background and the complex background.Since there is not a unified selection standard of hand image feature and kernel function, some works need to be studied from theory to application. Also the basis kernel combination algorithm of different levels and different descriptors should be considered in the future.

Acknowledgments This work is supported by the National Natural Science Foundation of China under Grant No.61103123 and No.61203021.

References

1. Arther D, Vassilvitskiis S (2007) k-means++: The advantages of careful seeding. In: Proceedings the eighteenth annual ACM-SIAM symposium on Discrete algorithms, Pennsylvania, USA, pp 1027–1035
2. Agnes J, Yann R, Sebastien M (2006) Hand posture classification and recognition using the modified census transform. In: Proceedings Intl. Conf. on Automatic Face and Gesture Recognition, IEEE, Southampton, UK, pp 351–356
3. Bach F, Lanckriet G, Jordan M (2004) Multiple kernel learning, conic duality and the smo algorithm. In: Proceedings Intl. Conf. on Machine Learning, Banff, Canada, pp 6–13
4. Barla A, Odone F, Verri A (2003) Histogram intersection kernel for image classification. In: Proceedings Intl. Conf. on Image Processing, Catalonia, Spain, pp 513–516
5. Belongie S, Malik J, Puzicha J (2002) Shape matching and object recognition using shape contexts. Trans Pattern Anal Mach Intell 24(4):509–522
6. Bosch A (2007) Image classification for a large number of object categories. PhD dissertation, University of Girona, Spain

7. Brehar R, Nedevschi S (2013) Pedestrian detection in traffic scenes using multi-attitude classifiers. In: Proceedings Intl. Conf. on Intelligent Transportation Systems, Hague, pp 1077–1083
8. Camps-Valls G, Gomez-Chova L, Munoz-Mari J, Vila-Frances J, Calpe-Maravilla J (2006) Composite kernels for hyperspectral image classification. IEEE Geosci Remote Sens Lett 3(1):93–97
9. Chang C, Lin C (2001) LIBSVM: A library for support vector machines[Online]. Available: http://www.csie.ntu.edu.tw/cjlin/libsvm.
10. Chen Y, Tseng K (2007) Multiple-angle hand gesture recognition by fusing SVM classifiers. In: Proceedings Intl. Conf. on Automation Science and Engineering, Scottsdale AZ, USA, pp 527–530
11. Chuang Y, Chen L, Chen G (2011) Hierarchical bag-of-features for hand posture recognition. In: Proceedings Intl. Conf. on Image Processing, Brussels, pp 1777–1781
12. Chuang Y, Chen L, Chen G (2013) Hierarchical Bag of Features with Spectral-HIK filter based hand posture recognition. J Zhejiang Univ (Eng Sci) 47(9):1531–1536
13. Dalal N, Triggs B (2005) Histograms of oriented gradients for human detection. In: Proceedings Intl. Conf. on Computer Vision and Pattern Recognition, San Diego, USA, pp 886–893
14. Dardas N, Georganas N (2011) Real-time hand gesture detection and recognition using bag-of-features and support vector machine techniques. IEEE Trans Instrum Meas 60(11):3592–3607
15. Deng L, Lee D, Keh H, Liu Y (2010) Shape context based matching for hand gesture recognition. In: Proc. IET Intl. Conf. on Frontier Computing. Theory, Technologies and Applications, pp 436–444
16. Fang Y, Wang K, Cheng J, Lu H (2007) A real-time hand gesture recognition method. In: Proceedings Intl. Conf. on Multimedia and Expo, pp 995–998
17. Gehler P, Nowozin S (2009) Feature combination for multiclass object classification. In: Proceedings Intl. Conf. on Computer Vision, Kyoto, Japan, pp 221–228
18. Hu M, Chen Y, Kwok J (2009) Building sparse multiple-kernel SVM classifiers. IEEE Trans Neural Netw 20(5):827–839
19. Ju Z, Liu H (2011) A unified fuzzy framework for human hand motion recognition. IEEE Trans Fuzzy Syst 19(5):901–903
20. Lanckriet G, Cristianini N, Bartlett P, Ghaoui E, Jordan MI (2004) Learning the kernel matrix with semidefinite programming. J Mach Learn Res 5(1):27–72
21. Li F, Petro P (2005) A Bayesian hierarchical model for learning natural scene categories. In: Proceedings Intl. Conf. on Computer Vision and Pattern Recognition, Los Alamitos, USA, pp 524–531
22. Li X, Sun X, Sun H, Li Y, Wang H (2012) Generalized multiple kernel framework for multiclass geospatial objects detection in high-resolution remote sensing images. Opt Eng 51(1):1–10
23. Murthy G, Jadon R (2009) A review of vision based hand gesture recognition. Intl J Inf Technol Knowl Manag 2(2):405–410
24. Ren Y, Gu C (2011) Hand gesture recognition based on HOG characters and SVM. Bull Sci Technol 27(2):211–214
25. Sun Q, Zeng S, Liu Y, Heng P, Xia D (2005) A new method of feature fusion and its application in image recognition. Pattern Recognit 38:2437–2448
26. Triesch J, Malsburg C, Marcel S (2014) Hand posture and gesture datasets; Jochen Triesch static hand posture database[Online]. Available: http://www.idiap.ch/resources/gestures/, July 8
27. Triesch J, Von DMC (2002) Classification of hand postures against complex backgrounds using elastic graph matching. Image Vis Comput 20(1):937–943
28. Tuia D, Camps-Valls G, Matasci G, Kanevski M (2010) Learning relevant image features with multiple kernel classification. IEEE Trans Geosci Remote Sens 48(10):3780–3791
29. Vapnik VN (1995) The nature of statistical learning theory. Springer Science+Business Media, LLC, New York, pp 83–102
30. Varma M, Ray D (2007) Learning the discriminative power-invariance trade-off. In: Proceedings Intl. Conf. on Computer Vision, Rio de Janeiro, Brazil, pp 1–8
31. Vedaldi A, Gulshan V, Varma M, Zisserman A (2009) Multiple kernels for object detection. In: Proceedings Intl. Conf. on Computer Vision, Kyoto, Japan, pp 606–613
32. Wang C, Wang K (2008) Hand posture recognition using Adaboost with SIFT for human robot interaction, vol 370. Springer, Berlin
33. Yuan H, Chih H, Hsiang C (2009) Vision based hand gesture recognition using PCA+Gabor filters and SVM. In: Proceedings Intl. Conf. on Intelligent Information Hiding and Multimedia Signal Processing, New York, USA, pp 1–4
34. Zhang H, LI H, Zhou M (2013) Hand posture recognition based on multi-feature and compressive sensing. J Hunan Univ(Nat Sci) 40(3):87–92