Effective Face Recognition Using Bag of Features with Additive Kernels

Shicai Yang[a], George Bebis[b], Yongjie Chu[a] and Lindu Zhao[a]
[a]Southeast University, Institute of Systems Engineering, No.2 Sipailou road, Nanjing, China, 210096
[b] University of Nevada, Dept. of Computer Science & Engineering, Reno, 1664 N. Virginia Street, USA, 89557

Abstract. In the past decades, many different techniques have been used to improve face recognition performance. The most common and well-studied ways are to use the whole face image to build a subspace based on the reduction of dimensionality. Differing from methods above, we consider face recognition as an image classification problem. The face images of the same person are considered to fall into the same category. Each category and each face image could be both represented by a simple pyramid histogram. Spatial dense SIFT features and bag of features method, are used to build categories and face representations. In an effort to make the method more efficient, a linear SVM solver, Pegasos, is used for the classification in the kernel space with additive kernels instead of nonlinear SVMs. Our experimental results demonstrate that the proposed method can achieve very high recognition accuracy on the ORL, YALE and FERET databases.

Keywords: Face recognition, Bag of features, Pyramid histogram, Additive kernels

Address all correspondence to: Lindu Zhao, Southeast University, Institute of Systems Engineering, No.2 Sipailou road, Nanjing, China, 210096; Tel: +8625-83793776; Fax: +8625-83794731; E-mail: ldzhao@seu.edu.cn

1 Introduction

Face recognition has attracted much attention from the community of pattern recognition, machine learning, and computer vision due to its extensive potential applications as well as theoretical challenges. Many different techniques have been studied to improve face recognition performance over the past few decades. One of the most successful and well-studied ways are the appearance based face recognition methods. When using these methods, we usually take the whole face image to build a subspace based on the reduction of face space dimensionality using different subspace methods [1], such as PCA, Fisher LDA, ICA and LPP [2-5]. The above methods work in a linear space. A comparative study of these methods for face recognition could be seen in [6, 7]. However, the face manifold in subspace need not be linear and many research efforts have shown that the face images possibly reside on a nonlinear sub-manifold [8-11]. Since PCA and LDA effectively see only the Euclidean structure, kernel methods [12-14] and manifold learning methods [5, 8, 9, 15, 16] have been applied to recognize faces. However, they are computationally expensive and we cannot explicitly point out the exact structure of the manifold on which the face images possibly reside. And also, there is a computationally expensive but powerful tool for statistical signal modeling, called sparse coding, has been successful applied in face recognition systems and gets some beautiful results [17, 18]. Besides, there is another kind of methods called geometry feature-based methods [19, 20]. These approaches are less used nowadays.

Since the past couple of years, with the development of computer vision and related theories, more and more advanced local features have appeared for the purpose of general object recognition and classification, such as SIFT [21], LBP [22], HOG [23] and so on. As one of the most widely used local descriptors, SIFT features have been successfully applied on face recognition problems [24, 25] and it does have some advantages. However it is still hard to say SIFT features for face recognition outperform other methods.

As we know, there are two crucial issues involved in developing face recognition systems, namely, face representation and classifier design [1, 26]. Face representation is used to derive a set of features from the raw face images which minimizes the intra-class variations (i.e., within face instances of a same individual) and maximizes the inter-class variations (i.e., between face images of different individuals), while a successful classifier with high performance can find a good separation between different classes even if they share significant similarity with each other. If inadequate face representations are adopted, even the most sophisticated classifier fails to accomplish the face recognition task. On the other hand, if good face representations are adopted, but we do not have a good classifier, we still cannot achieve high recognition performance.

In this work, we consider face recognition as an image classification problem. All the face images of the same person are considered to fall into the same category. Hence, when the face images are classified into correct categories, the faces are correctly recognized. Based on this idea, we propose an improved bag of features representation by combining the spatial information to obtain a pyramid histogram of visual words for face representation. For the classification task, since the runtime complexity of a nonlinear SVM classifier is very high, we use a linear SVM solver, Pegasos [27], with the help of additive kernels, to get faster training and classification speeds. Our experiments on benchmark face databases clearly validate the face recognition performance of the proposed method. Though bag of features, spatial pyramid and Pegasos have been used to address computer vision problems respectively, in this paper we reuse bag of features as well as spatial pyramid to design the effective high-dimensional features for face representation, we also take advantages of additive kernel-based linear SVM to make the high-dimensional features display promise recognition performance efficiently. That is to say, we propose an effective and efficient method for face recognition.

The rest of the paper is structured as follows. In Section 2, we describe the face representation by using bag of features and spatial pyramid method. In Section 3 we describe the additive kernel based SVM classification. In Section 4 we present our experimental results. Conclusions are in Section 5.

2 Face representation via bag of features

The method of bag of features [28-30], which is also called bag of visual words, is inspired from natural language processing applications, where each text document is represented by a histogram of word occurrences in the document. To make the jump from words to “visual” words, local features are extracted from a set of training images, and then they are vector quantized using k-means clustering. The cluster centers obtained are referred as “visual words”, and the combination of the visual words founded forms a vocabulary. Compared with textual document based categorization, there is no available vocabulary for image based object categorization, so the vocabulary has to be learned from a training image set. Then the features are mapped to one or more visual words in the vocabulary and each category is represented by a visual word. The general steps for vocabulary generation are illustrated in Fig.1. Finally, each image is represented by a histogram representing the frequency of visual words in the image. We refer to the histogram as the bag of features representation of the image. This bag of features representation can be used to compute similarity between images or give a query to retrieve the most similar images in a database. Moreover, object classes (e.g., faces, cars) can be modeled by training a classifier using bag of features representation to learn the separating boundaries between object categories. Despite its simplicity and low computational complexity, bag of features has been surprisingly successful in various applications including object or scene detection [28, 31], classification [24], and retrieval [29].

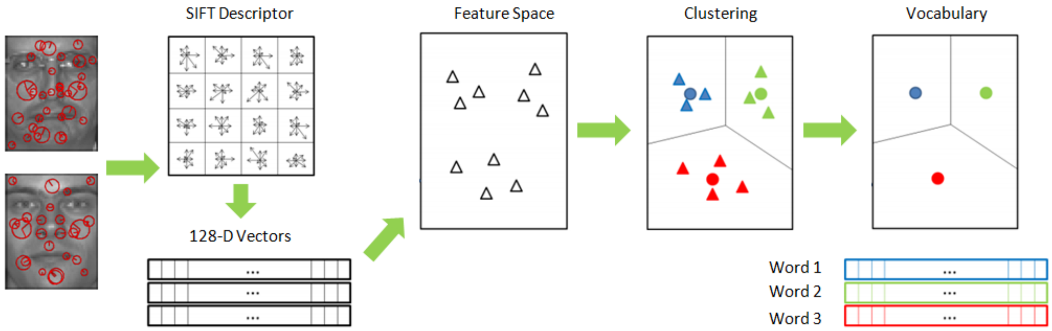

Fig. 1 General steps for vocabulary generation.

However, bag of features method eliminates all of the spatial information accompanying the features. Spatial pyramid histogram matching that restores some spatial information is introduced to solve such problems by Lazebnik *et al.* [31]. When a spatial pyramid is used in combination with a bag of features approach, there are several vectors constructed instead of one. Each vector is a bag of features for one region of the image. Once all vectors are computed, they are concatenated into one larger vector (histogram), which implicitly contains coarse spatial information. Then, each face image is represented by this histogram of visual words presented in the image.

We follow the approach of Bosch *et al.* [32] to build the spatial pyramid histogram to improve our bag of features face representation, that is, building spatially enhanced histograms at various resolutions and concatenating the results. More precisely, we first compute SIFT features densely at points on a regular grid with spacing $M(e.g.,2,4)$ pixels, shown in Fig.2 (the 2nd row). To allow for scale variation between images, multiple descriptors are computed for each grid point. In detail, at each grid point the descriptors are computed over four circular support patches with different radii $R(e.g.,4,8,12,16)$, consequently each point is represented by four SIFT descriptors. These dense features are vector quantized into K visual words using k-means clustering.

Then to build a spatial histogram with L levels, we first create the level 0 histograms (K dimensional vectors) with dense SIFT features over the entire image. Next, the image is divided into four regions equally and a level 1 histogram ($4K$ dimensional vector) is computed over each region. If we want to build a level l histogram, we can repeat this process by sub-dividing each region recursively, which is shown in Fig.2 (top row and the 3rd row). At last, we concatenate all the histograms together into a larger vector with the dimension of $\sum_{l=0}^{L} 4^l K$. This vector is just the histogram for our face representation, which is illustrated in Fig. 2 (bottom row).

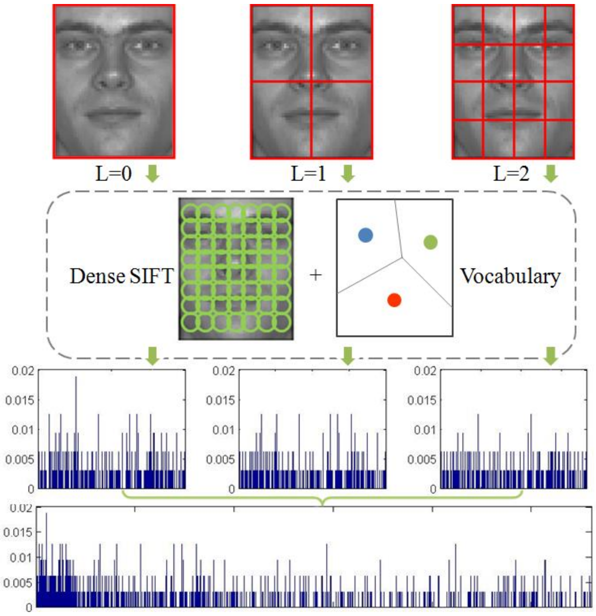

Fig. 2 Face representation via spatial pyramid based bag of features.

3 SVM classification with additive kernels

As a fundamental problem in pattern analysis, classification is very important for recognition tasks. A series of functions should be learned to separate data points from different classes. The SVM classifier, which aims at recovering a maximal margin separating hyper plane in the feature space, is a very popular and powerful tool for classification and has demonstrated state-of-the-art performance in many computer vision problems. SVMs can either operate explicitly in the input space leading to the linear SVM or implicitly in the feature face via the kernel mapping giving rise to the kernel SVM [33, 34]. Linear SVM is simple to train and easy to use as it only involves inner product with the input data. However, it cannot be applied to nonlinear, which leads to its limited applications to the real world problems where in most cases the data are inherently nonlinear. On the other hand, the nonlinear SVM can handle linearly inseparable data, but its complexity is multiplied with the number of support vectors, which is highly inefficient and leads to high computation costs. This is not favorable for prediction tasks on large scale data sets. Hence, it is highly desirable to have a classifier model with both the efficiency of linear SVM and the power of nonlinear SVM. One way to solve such problems is that, performing the linear classification in the embedding space, instead of nonlinear classification in the original space. Here we introduce the SVM solver, Pegasos [27], and additive kernels [35, 36] for the classification tasks of face representation histograms. Hereinto, additive kernels help us to perform an explicit (approximate) embedding of the data, while Pegasos is used for learning in the new embedding space.

Pegasos introduces a blocked gradient optimization approach for SVMs with linear kernels, which solves the linear SVM problem:

$$\min_{w} \frac{\lambda}{2}\|w\|^2 + \frac{1}{m}\sum_{i}^{n}\ell(w;(x_i, y_i)) , \qquad (1)$$

where x_i are data vectors in R^d , $y_i \in \{-1,+1\}$ are binary labels, $\lambda > 0$ is the regularization parameter and $\ell(w;(x_i, y_i)) = \max\{0, 1 - y_i \langle w, x_i \rangle\}$ is the hinge loss. The result of the optimization is a model $w \in R^d$ that yields the decision function $F(x) = sign\langle w, x \rangle$. In this method, every gradient descent step is accompanied with a projection step. Instead of computing the costly operation of the exact gradient, a random data point is selected, and an approximated gradient is computed, based solely on this data point. It is demonstrated that, for linear kernels, Pegasos can achieve state-of-the-art results as well as fast rate of convergence. From a theoretical convergence analysis, it is shown that this algorithm converges in $\mathrm{O}(d/(\lambda\varepsilon))$ where d is a bound on non-zero features in an example, λ is the regularization parameter, and ε is the error bound on estimation. That is why this algorithm is suitable for learning from large scale data sets, since the runtime does not depend directly on the size of the training set. Pegasos can also be adapted to employ non-linear kernels. Let $k(x, y)$ be a positive definite kernel. A feature mapping is a function $\Psi(x)$, which maps the data x to a Hilbert space $\mathcal{H}$, such that $k(x, y) = \langle \Psi(x), \Psi(y) \rangle$. Using this representation the nonlinear SVM learning objective function can be rewritten as:

$$\min_w \frac{\lambda}{2} \|w\|^2 + \frac{1}{m} \sum_i^n \ell(w; (\Psi(x_i), y_i)) . \qquad (2)$$

Thus the only difference with the linear case is that the feature mapping $\Psi(x)$ is used instead of the data x. So how to find or construct a feature mapping $\Psi(x)$ for SVM classification is what we are going to discuss next.

Since constructing features and feature space are based on those histograms computed from our bag of features representation model, the computation of histogram dissimilarity or distances should be discussed first, that is also related to the feature mapping Ψ . When computing the dissimilarity of two representation histograms, the most frequently used distance metric is the Euclidean distance. However, despite its mathematical simplicity and efficacy in many other applications, it is found that the Euclidean distance is not the most suitable similarity measure [37]. In particular, it is considered that the histogram intersection kernel and the Chi square measure can give significantly improved results. One of the common characteristics of histogram intersection and Chi square is that they are instances in a family of kernels called the additive kernels, which are defined as:

$$K(x,y)=\sum_{i=1}^{d}k(x_i,y_i)\ , \qquad (3)$$

where d is the dimension of the input histograms x,y , i is the component (bin) index, and $k(x,y)$ is an homogeneous positive definite kernel on the non-negative reals $\Re_0^+$. We call the kernel k a homogeneous kernel if $k(cx,cy)=ck(x,y)$ for any $c>0$. If we set $k(x,y)=\min(x,y)$, the histogram intersection kernel is obtained. When setting $k(x,y)=2xy/(x+y)$, it yields the Chi square kernel [36]. These two kernels are illustrated in Fig. 3. Kernels of histogram intersection, Chi square, Jensen-Shannon, and Hellinger’s are some common examples of additive kernels [36]. In fact, these additive kernels are becoming very popular in computer vision and machine learning applications nowadays. Recent advances have shown that additive kernels and explicit embeddings are the best performers in most visual classification tasks [36-38].

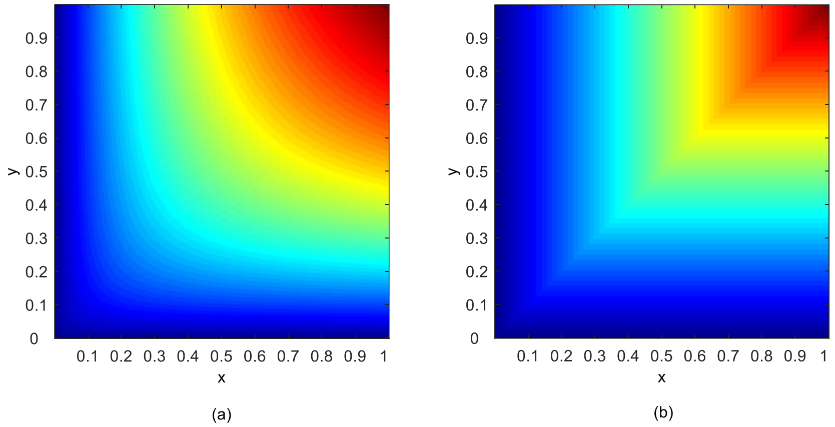

Fig. 3 Visualization of kernels for 1D features: (a) the Chi square kernel, (b) Histogram intersection kernel.

In our work, we use Chi square kernel as a measure of similarity between histograms, which is proved to be excellent for bag of features models. We can also see from Fig. 3 that, the Chi square kernel is much smoother than histogram intersection kernel. Since Chi square kernel is positive definite, it can be used as a kernel discriminative classification using SVMs. And also, it is an additive kernel, so the comparison between histograms could be a linear combination of functions of each coordinate of the histogram. However, using a kernel derived from chi square distance to train a SVM classifier is computationally very expensive as the test time scales linearly with the number of support vectors, which limited its application in practice. Fortunately, this limitation was removed for the SVM classifiers by recent research works [35-39]. In other words, it is possible to speed up classification for additive kernel based SVMs, which is firstly presented by Maji *et al.* [38]. We follow the work of Vedaldi and Zisserman [36], to make a closed-form approximated finite dimensional feature mapping

$\Psi(x)$ for our Chi square kernel that enables us to use this additive kernel as if it was linear. The idea is to approximate the function $k(x, y)$ as the product of two small vectors $\Psi(x)$ and $\Psi(y)$. For the Chi square kernel:

$$k(x, y) = \frac{2xy}{x+y} \approx \left\langle \Psi(x), \Psi(y) \right\rangle, \tag{4}$$

in order to find a good approximation of Ψ, the exact embedding function should be found first. Based on the work of Hein and Bousquet [40], for any homogenous kernels there exists a symmetric non-negative measure $\kappa(\lambda)d\lambda$ such that:

$$k(x, y) = \sqrt{xy}\int_{-\infty}^{+\infty} e^{-i\lambda \log(y/x)} \kappa(\lambda) d\lambda . \tag{5}$$

This can be rewritten as:

$$k(x, y) = \int_{-\infty}^{+\infty} \Psi_\lambda(x) * \Psi_\lambda(y) d\lambda = \left\langle \Psi(x), \Psi(y) \right\rangle, \tag{6}$$

with $\Psi_\lambda(x) = e^{-i\lambda \log(x)}\sqrt{x\kappa(\lambda)}$. Here $\Psi(x)$ is just the exact feature mapping of the data x, and this feature cannot be used directly for computations as it is a continuous function, so approximations are needed. A finite approximated feature mapping $\Psi(x)$ can be obtained by sampling from $\Psi(x)$. Due to the symmetries of the feature mapping $\Psi(x)$, $\Psi(x)$ can be reduced to a real vector with the dimension of $2n+1$. For instance, by taking $2n+1$ times sampling, a simple 3x approximation of Chi square kernel is given by:

$$\Psi(x) = \sqrt{x}\begin{bmatrix} 0.8 \\ 0.6\cos(0.6\log(x)) \\ 0.6\sin(0.6\log(x)) \end{bmatrix}. \tag{7}$$

The error is independent of the data dimension and decays exponentially fast with the approximation order for the Chi square kernel. An example of the error of this finite representation for one-dimensional histograms can be seen in Fig. 4.

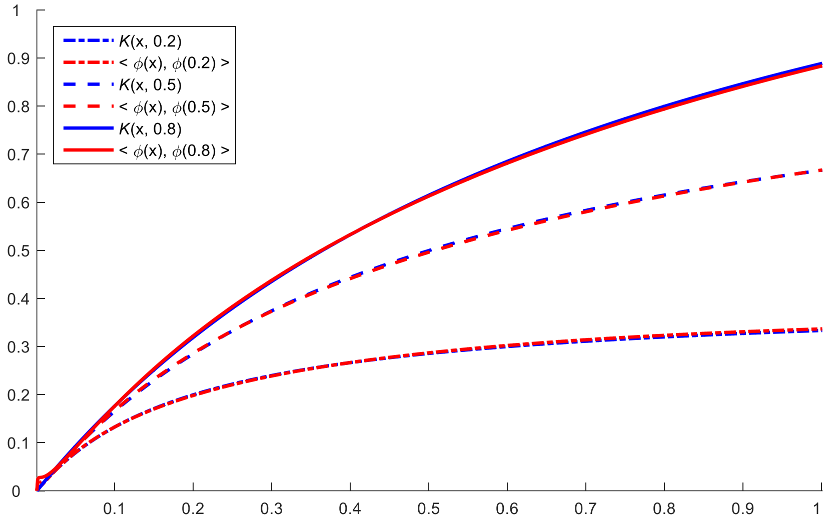

Fig. 4 The exact Chi square features (red line) vs. approximate features (blue line)

4 Experiments and results

The proposed approach is extensively tested for face recognition on the following widely used benchmark data sets such as the ORL face database [41], the YALE face database [3] and the FERET face database [42]. The ORL database is used to evaluate the performance of the proposed method under conditions where the poses varied. The Yale database is used to examine the performance when both facial expressions and illuminations are varied. The FERET database is employed to test the recognition performance with a large number of classes in a dataset.

In the experiments, we set the parameters in our method as follows. The number of histograms is set to be 3, namely three-level histograms are built. The value of *C* in

SVM is set to be a fixed value from the set {0.001, 0.01, 0.1, 1, 10}, and finally fixed to be 10 in our experiments, which is obtained by cross-validation, and it is found that various values of C make little difference on the final performance of our method.

Another important parameter in our proposed approach is the vocabulary size K. Larger K will be helpful for the proposed method to display better recognition performance, meanwhile a larger K will be much more time-consuming but produces little improvement in performance. Therefore, the selected value of K is a tradeoff between effectiveness and efficiency. From our knowledge, K could be set to 3000 or 5000, which is quite reasonable. Additionally, according to different image sizes, sampling strategy can be used to get a fixed number of features in a block so as to decrease time complexity for large size images. In our experiments, we set the largest value of K to be 3000, 3000 and 8000 for 3 different face databases, respectively.

4.1 Experiments on the ORL database

The ORL dataset contains 400 face images of 40 persons, with 10 per person, which were taken at different time, under different lighting conditions, and with different facial expressions and details. Moreover, all the images were taken against a dark homogenous background with the subjects in an upright, frontal position, with tolerance for some tilting and rotation of up to about 20 degrees. These images are gray scale with a resolution of 92x112. The original images with no scaling or cropping are used in our experiments. Fig. 5 shows some face examples from the database.

For each experiment, we randomly select 5 images per person for training and the rest for testing (200 images for training and 200 for testing). Recognition is performed using the SVM classifier mentioned above in the approximated additive kernel space. In order to assess recognition performance, we repeat each of the experiments for 5 times, and the reported results refer to the average performance in such 5 runs. The experimental results with different vocabulary sizes and different numbers of training samples are listed in Table 1 and Table 2. It can be seen from Fig. 6(a) that, the proposed method has very good recognition performance with nearly perfect recognition accuracies of 99%, and even 100% while the vocabulary size was set properly. And it should be also pointed out that, the recognition performance of the proposed method is extremely stable on this dataset since the standard deviations always keep under 0.4%. We have compared our results to the other reported results from several state-of-the-art methods under the same conditions. The best average recognition rates of different methods are reported in Table 3. Some results are directly taken from published papers indicated by citations. It is worth mentioning two points here about the comparisons in this work. First, we also conduct some experiments about FLDA with additive kernel and LBP with additive kernel, but we do not report these results to be compared since the performances are uncompetitive. Second, since we choose the best average recognition rates as our final results, it may include some bias, but other results still have big gap in performance with our proposed methods. Moreover, we have investigated the effect of the number of training samples on the recognition performance. Fig. 6(b) shows the recognition performance of the proposed method, corresponding to different numbers of training samples while the vocabulary size is set to be 1500. As we can see, there is a significant performance improvement (around 8%) by using two face samples instead of one. The good results indicate that our approach is very effective and robust with respect to alignment.

Fig. 5 Eight face samples randomly selected from the ORL database.

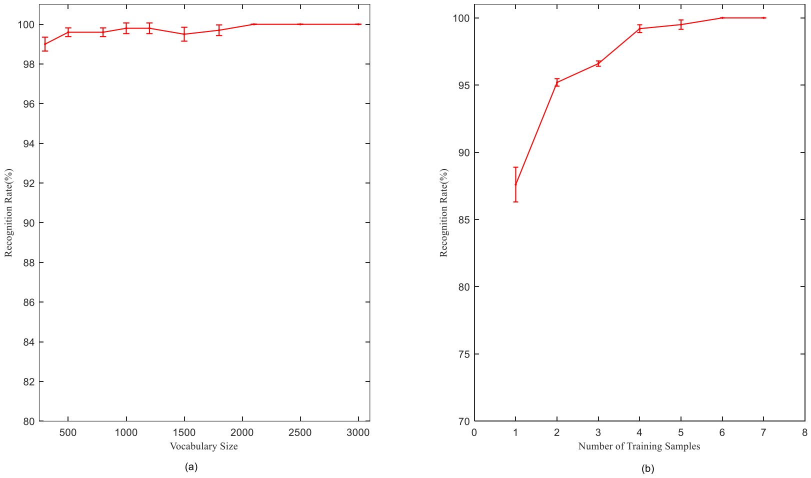

Fig. 6 Recognition rates vs. vocabulary size (a) and number of training samples (b) on the ORL dataset.

Table 1 Results on the ORL dataset with different sizes of vocabulary.

Vocabulary Size K	300	500	800	1000	1200	1500	1800	2100	2500	3000
Recognition Accuracy %	99	99.6	99.6	99.8	99.8	99.5	99.7	**100**	**100**	**100**
Standard Deviation %	0.35	0.22	0.22	0.27	0.27	0.35	0.27	**0.00**	**0.00**	**0.00**

Table 2 Results on the ORL dataset with different numbers of training samples.

No. of Training Samples	1	2	3	4	5	6	7
Recognition Accuracy %	87.6	95.2	96.6	99.2	99.5	100	100
Standard Deviation %	1.29	0.28	0.20	0.29	0.35	0.00	0.00

Table 3 Comparison of the propose method with other methods on the ORL database.

	FLDA [25]	2DPCA [43]	LBP [44]	HOG [45]	SIFT [25]	PDSIFT [25]	S2FF [45]	Proposed

Rates %	92.5	96.0*	98.0	95.5	90.0	95.5	96.2	**100**

Note that, the result indicated by the asterisk was obtained when they used the first five images of each person for training.

4.2 Experiments on the YALE database

The Yale face database contains 165 images with 11 different images for each of the 15 distinct subjects. The 11 images per subject are taken under different facial expressions, lighting conditions or configurations: center-light, with glasses, happy, left-light, without glasses, normal, right-light, sad, sleepy, surprised, and wink, as illustrated in Fig. 7. The image size is 243x320 pixels. In our experiments, the original gray scale images are used without any normalization, cropping or scaling. We randomly select 6 images from each subject for training and the rest for testing (90 images for training and 75 for testing). Similarly, we repeat each of the experiments for 5 times and take the average performance. The recognition rates with different sizes of vocabulary are reported in Table 4 and illustrated in Fig. 8(a). These results indicate that, we can get the best average recognition rate of 99.7% with the standard deviation under 0.6% when the vocabulary size is 1500. And also, if the vocabulary size is set to be 300, the average accuracy of 97.3% with the standard deviation of zero can be obtained, which means that the recognition performance is highly stable and only 2 images are placed in the wrong category. We have also compared our results to the other reported results from several state-of-the-art methods, which are reported in Table 6. Results are directly taken from published papers indicated by citations. Clearly, our face recognition framework achieves the best face classification performances, which are almost perfect. In addition, we have explored the relationships between the recognition performance and the number of training samples, which is shown in Fig. 8(b). It is revealed from Table 5 that, there is also a significant performance improvement (24%) by using two face samples instead of one.

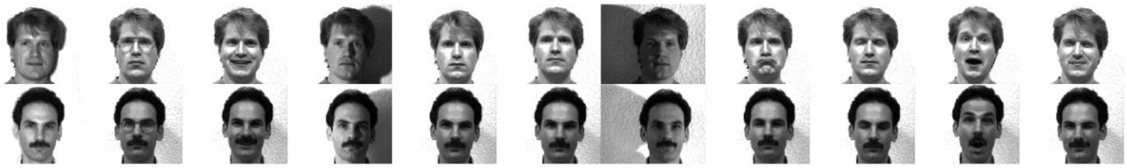

Fig. 7 Face samples of the first two subjects selected from the YALE database.

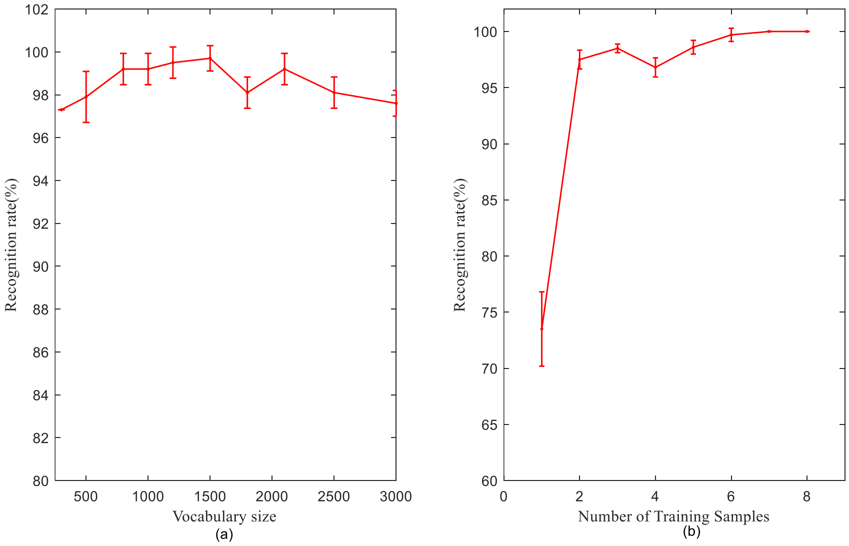

Fig. 8 Recognition rates vs. vocabulary size (a) and number of training samples (b) on the YALE dataset.

Table 4 Results on the YALE dataset with different sizes of vocabulary.

Vocabulary Size K	300	500	800	1000	1200	1500	1800	2100	2500	3000
Recognition Accuracy %	97.3	97.9	99.2	99.2	99.5	**99.7**	98.1	99.2	98.1	97.6
Standard Deviation %	**0.00**	1.19	0.73	0.73	0.73	0.59	0.73	0.73	0.73	0.60

Table 5 Results on the YALE dataset with different numbers of training samples.

No. of Training Samples	1	2	3	4	5	6	7	8
Recognition Accuracy %	73.5	97.5	98.5	96.8	98.6	99.7	100	100
Standard Deviation %	3.31	0.84	0.38	0.85	0.61	0.59	0.00	0.00

Table 6 Comparison of the propose method with other methods on the ORL database.

	PCA [46]	2DPCA [46]	2DLDA [46]	Fuzzy FLDA [47]	S-LDA [48]	HKLBP [49]	Proposed
Rates %	94.0	94.5	95.8	94.8	81.7*	87.5*	**99.7**

Note that, the result indicated by the asterisk was obtained when they used the images with a small size around 32x32 for training and testing.

4.3 Experiments on the FERET database

The FERET database is a standard dataset for face recognition technologies, which contains a large number of images acquired during different photo session and has a good variety of gender, ethnicity and age groups. For our evaluation set, we randomly choose 400 images of 200 persons from subjects 450~700. Each person has two images with different race, age, expression, illumination, occlusion, etc. One of them (Fa) is used for training, while the other one (Fb) is used for testing. So there is only a single image from each person used for training. All faces are normalized and masked to include the face region yielding an image size of 48x60 pixels. Some representative examples are shown in Fig. 9. We have investigated the effect of vocabulary sizes on recognition performance, which is shown in Fig. 10. As mentioned before, the vocabulary size will definitely influence the performance of the proposed algorithm. When the vocabulary size is small, the performance is not so good. However, increasing vocabulary size increases computation requirements. In our experiments, when the size is set to be 7000, our algorithms get the best average recognition rate of 98.6%, which is also nearly perfect. All the results with different vocabulary sizes are listed in Table 7. The cumulative match curves (CMC) are also demonstrated on the selected FERET database in Fig. 11. It is shown that, the proposed algorithm can achieve 100% accuracy in Rank-3 while the vocabulary size is set properly. These results demonstrate that the proposed method is highly robust to the extrinsic imaging conditions, and also effective for the face recognition problems with single-image-training and large number of classes. To further validate the effectiveness of our method, we have compared its Rank-1 recognition performance with other known results. Table 8 displays the comparisons between our method and these state-of-the-art methods indicated by citations on the FERET database. From the table, we can see that the result of our method is impressively better.

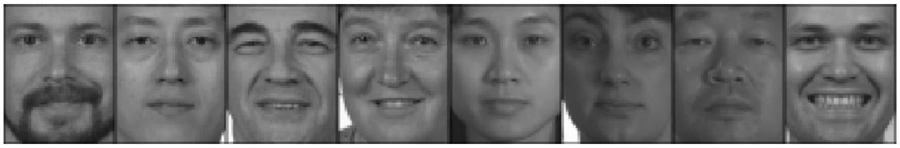

Fig. 9 Eight face samples randomly selected from the FERET database.

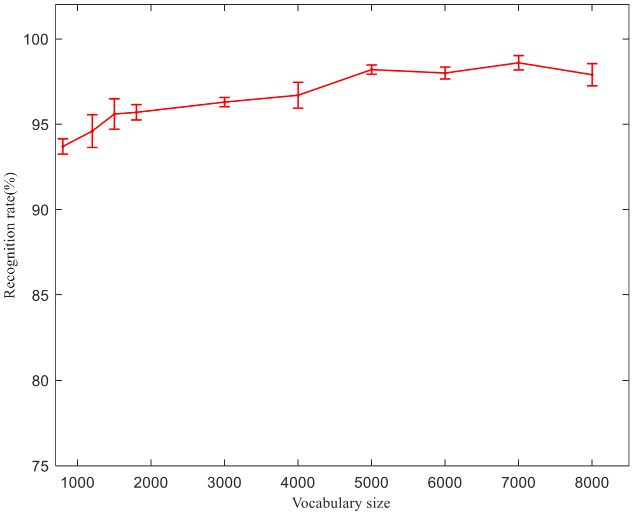

Fig. 10 Recognition rates vs. vocabulary size on the FERET dataset.

Table 7 Results on the FERET dataset with different sizes of vocabulary.

Vocabulary Size K	800	1200	1500	1800	3000	4000	5000	6000	7000	8000
Recognition	93.7	94.6	95.6	95.7	96.3	96.7	98.2	98	**98.6**	97.9

Accuracy %										
Standard Deviation %	0.45	0.96	0.89	0.45	0.27	0.76	0.27	0.35	0.42	0.65

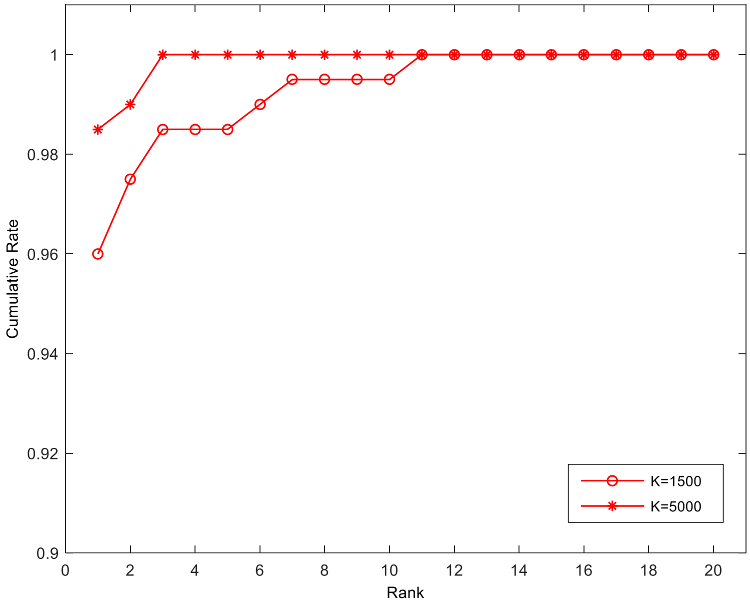

Fig. 11 Cumulative match curves on the FERET database when the vocabulary size is 1500 and 5000.

Table 8 Comparison of the propose method with other methods on the FERET database.

	HOG-EBGM [50]	LBP [44]	S2FF [45]	LGT [51]	LGBPHS [52]	HGPP [53]	LLGP [54]	Proposed
Rates %	95.5	97.0*	91.5	97.0	98.0*	97.6	98.0*	**98.6**

Note that, the result indicated by the asterisk was obtained when they used a weighting strategy for different facial parts as well as different patterns.

5 Conclusions

In this paper, the face representation via spatial pyramid based bag of features for face recognition has been proposed, then approximated additive kernel based linear SVM has been introduced for efficient classification, i.e., we propose an new method for face recognition problem. Extensive experiments demonstrate that the proposed method can achieve very good performance and outperforms most of state-of-the-art methods. It is also shown that, the proposed method can well handle different sizes of images as well as large number of categories, and make full use of a single image per subject for training. Moreover, it is very robust to the variations of expressions, illuminations and poses etc. Although high performance is achieved by the proposed method, we should also point out a drawback of our method. That is, the good results sometimes lie in a large number of local descriptors and the high-dimensional histogram features. In the future, we will improve this problem and evaluate our algorithms on some "real-world" face recognition tasks.

Acknowledgement

This work is partially supported by the National Key Technology R&D Program of China during the 12th Five-Year Plan Period (No.2013BAD19B05), the National Natural Science Foundation of China (No.71390333). The first author thanks China Scholarship Council (CSC) for supporting the joint-training doctoral students program. We also thank the anonymous reviewers, whose comments have helped improving this paper.

References

[1] S. Z. Li and A. K. Jain, *Handbook of Face Recognition*, Springer, New York, 2004.

[2] M. Turk and A. Pentland, Eigenfaces for Recognition, *J. Cognitive Neuroscience,* vol.3, no.1, pp.71-86, 1991.

[3] P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman, Eigenfaces vs. Fisherfaces: Recognition using Class Specific Linear Projection, *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol.19, no.7, pp.711-720, 1997.

[4] M. S. Bartlett, J. R. Movellan, and T. J. Sejnowski, Face Recognition by Independent Component Analysis, *IEEE Transactions on Neural Networks*, vol.13, no.6, pp.1450-1464, 2002.

[5] X. He, S. Yan, Y. Hu, *et al.*, Face Recognition using Laplacianfaces, *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol.27, no.3, pp.328-340, 2005.

[6] X. Wang and X. Tang, A Unified Framework for Subspace Face Recognition, *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol.26, no.9, pp.1222-1228, 2004.

[7] K. Delac, M. Grgic, and S. Grgic, Independent Comparative Study of PCA, ICA, and LDA on the FERET Data Set, *International Journal of Imaging Systems and Technology*, vol.15, no.5, pp.252-260, 2005.

[8] H. S. Seung and D. D. Lee, Cognition: The Manifold Ways of Perception, *Science*, vol.290, no.5500, pp.2268-2269, 2000.

[9] J. B. Tenenbaum, V. d. Silva, and J. C. Langford, A Global Geometric Framework for Nonlinear Dimensionality Reduction, *Science*, vol.290, no.5500, pp.2319-2323, 2000.

[10] A. Shashua, A. Levin, and S. Avidan, Manifold Pursuit: A New Approach to Appearance Based Recognition, *Proceedings of the International Conference on Pattern Recognition (ICPR)*, Quebec, Canada, pp. 590-594, 2002.

[11] S. T. Roweis and L. K. Saul, Nonlinear Dimensionality Reduction by Locally Linear Embedding, *Science,* vol.290, no.5500, pp.2323-2326, 2000.

[12] M. H. Yang, N. Ahuja, and D. Kriegman, Face Recognition using Kernel Eigenfaces, *Proceedings of the International Conference on Image Processing (ICIP)*, Vancouver, BC, Canada, pp. 37-40, 2000.

[13] F. R. Bach and M. I. Jordan, Kernel Independent Component Analysis, *Journal of Machine Learning Research*, vol.3, pp.1-48, 2002.

[14] K. B. Schölkopf, A. Smola, and K.-R. Müller, Nonlinear Component Analysis as a Kernel Eigenvalue Problem, *Neural Computation*, vol.10, no.5, pp.1299-1319, 1998.

[15] D. L. Donoho and C. Grimes, Hessian Eigenmaps: Locally Linear Embedding Techniques for High-Dimensional Data, *Proceedings of the National Academy of Sciences of the United States of America (PNAS)*, vol.100, no.10, pp.5591-5596, 2003.

[16] M. Belkin and P. Niyogi, Laplacian Eigenmaps for Dimensionality Reduction and Data Representation, *Neural Computation*, vol.15, no.6, pp.1373-1396, 2003.

[17] M. Yang and L. Zhang, Gabor Feature Based Sparse Representation for Face Recognition with Gabor Occlusion Dictionary, *Proceedings of the European Conference on Computer Vision (ECCV)*, Crete, Greece, pp. 448-61, 2010.

[18] J. Wright, A. Y. Yang, A. Ganesh, *et al.*, Robust Face Recognition via Sparse Representation, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol.31, no.2, pp.210-227, 2009.

[19] L. Wiskott, J. M. Fellous, N. Kuiger, *et al.*, Face Recognition by Elastic Bunch Graph Matching, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol.19, no.7, pp.775-9, 1997.

[20] R. Brunelli and T. Poggio, Face Recognition: Features versus Templates, *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol.15, no.10, pp.1042-52, 1993.

[21] D. G. Lowe, Distinctive Image Features from Scale-Invariant Keypoints, *International Journal of Computer Vision,* vol.60, no.2, pp.91-110, 2004.

[22] T. Ojala, M. Pietikainen, and T. Maenpaa, Multiresolution Gray-Scale and Rotation Invariant Texture Classification with Local Binary Patterns, *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol.24, no.7, pp.971-87, 2002.

[23] N. Dalal and B. Triggs, Histograms of Oriented Gradients for Human Detection, *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR),* San Diego, CA, USA, pp. 886-893, 2005.

[24] Z. Li, J. Imai, and M. Kaneko, Robust Face Recognition Using Block-Based Bag of Words, *Proceedings of the International Conference on Pattern Recognition (ICPR),* Istanbul, Turkey, pp. 1285-1288, 2010.

[25] C. Geng and X. Jiang, Face Recognition using Sift Features, *Proceedings of the IEEE International Conference on Image Processing (ICIP),* Cairo, Egypt, pp. 3313-3316, 2009.

[26] W. Zhao, R. Chellappa, P. J. Phillips, *et al.*, Face Recognition: A Literature Survey, *ACM Computing Surveys,* vol.35, no.4, pp.399-459, 2003.

[27] S. Shalev-Shwartz, Y. Singer, N. Srebro, *et al.*, Pegasos: Primal Estimated Sub-Gradient Solver for SVM, *Mathematical Programming,* vol.127, no.1, pp.3-30, 2011.

[28] G. Csurka, C. R. Dance, L. Fan, *et al.*, Visual Categorization with Bags of Keypoints, *ECCV Workshop on Statistical Learning in Computer Vision*, Prague, Czech Republic, 2004.

[29] J. Sivic and A. Zisserman, Video Google: A Text Retrievalapproach to Object Matching in Videos, *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Nice, France, pp. 1470 - 1477, 2003.

[30] F.-F. Li and P. Perona, A Bayesian Hierarchical Model for Learning Natural Scene Categories, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, San Diego, CA, United states, pp. 524-531, 2005.

[31] S. Lazebnik, C. Schmid, and J. Ponce, Beyond Bags of Features: Spatial Pyramid Matching for Recognizing Natural Scene Categories, *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, New York, NY, USA, pp. 2169-2178, 2006.

[32] A. Bosch, A. Zisserman, and X. Munoz, Image Classification using Random Forests and Ferns, *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Rio de Janeiro, Brazil, pp. 1779-1786, 2007.

[33] B. Scholkopf and A. J. Smola, *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*, The MIT Press, Cambridge, MA, USA, 2002.

[34] V. N. Vapnik, *Statistical Learning Theory*, John Wiley & Sons, New York, 1998.

[35] S. Maji and A. C. Berg, Max-Margin Additive Classifiers for Detection, *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Kyoto, Japan, pp. 40-47, 2009.

[36] A. Vedaldi and A. Zisserman, Efficient Additive Kernels via Explicit Feature Maps, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR),* San Francisco, USA, pp. 3539-3546, 2010.

[37] J. X. Wu, W. C. Tan, and J. M. Rehg, Efficient and Effective Visual Codebook Generation Using Additive Kernels, *Journal of Machine Learning Research,* vol.12, pp.3097-3118, 2011.

[38] S. Maji, A. C. Berg, and J. Maliks, Classification using Intersection Kernel Support Vector Machines Is Efficient, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Anchorage, AK, USA, 2008.

[39] J. Wu, A Fast Dual Method for Hik SVM Learning, *Proceedings of the European Conference on Computer Vision (ECCV),* Crete, Greece, pp. 552-565, 2010.

[40] M. Hein and O. Bousquet, Hilbertian Metrics and Positive Definite Kernels on Probability Measures, *Proceedings of the 10th International Workshop on Artificial Intelligence and Statistics (AISTATS)*, Hastings, Christ Church, Barbados, pp. 136-143, 2005.

[41] F. S. Samaria and A. C. Harter, Parameterisation of a Stochastic Model for Human Face Identification, *Proceedings of the 2nd IEEE Workshop on Applications of Computer Vision*, Sarasota, FL, USA, pp. 138-142, 1994.

[42] P. J. Phillips, M. Hyeonjoon, S. A. Rizvi, *et al.*, The FERET Evaluation Methodology for Face-Recognition Algorithms, *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol.22, no.10, pp.1090-104, 2000.

[43] J. Yang, D. Zhang, A. Frangi, *et al.*, Two-Dimensional PCA: A New Approach to Appearance-Based Face Representation and Recognition, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol.26, no.1, pp.131-137, 2004.

[44] T. Ahonen, A. Hadid, and M. Pietikainen, Face Recognition with Local Binary Patterns, *Proceedings of the European Conference on Computer Vision (ECCV)*, Prague, Czech Republic, pp. 469-481, 2004.

[45] E. Meyers and L. Wolf, Using Biologically Inspired Features for Face Processing, *International Journal of Computer Vision*, vol.76, no.1, pp.93-104, 2008.

[46] S. Noushath, G. Hemantha Kumar, and P. Shivakumara, $(2D)^2$LDA: An Efficient Approach for Face Recognition, *Pattern Recognition*, vol.39, no.7, pp.1396-1400, 2006.

[47] K.-C. Kwak and W. Pedrycz, Face Recognition using a Fuzzy Fisherface Classifier, *Pattern Recognition*, vol.38, no.10, pp.1717-1732, 2005.

[48] D. Cai, X. He, Y. Hu, *et al.*, Learning a Spatially Smooth Subspace for Face Recognition, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Minneapolis, MN, USA, 2007.

[49] X. Li, W. M. Hu, Z. F. Zhang, *et al.*, Heat Kernel Based Local Binary Pattern for Face Representation, *IEEE Signal Processing Letters*, vol.17, no.3, pp.308-311, 2010.

[50] A. Albiol, D. Monzo, A. Martin, *et al.*, Face Recognition using HOG-Ebgm, *Pattern Recognition Letters*, vol.29, no.10, pp.1537-1543, 2008.

[51] Z. Lei, S. Z. Li, R. Chu, *et al.*, Face Recognition with Local Gabor Textons, *International Conference on Advances in Biometrics (ICB)*, Seoul, Korea, pp. 49-57, 2007.

[52] W. Zhang, S. Shan, W. Gao, *et al.*, Local Gabor Binary Pattern Histogram Sequence (Lgbphs): A Novel Non-Statistical Model for Face Representation and Recognition, *Proceedings - 10th IEEE International Conference on Computer Vision, ICCV 2005, October 17, 2005 - October 20, 2005*, Beijing, China, pp. 786-791, 2005.

[53] B. Zhang, S. Shan, X. Chen, *et al.*, Histogram of Gabor Phase Patterns (HGPP): A Novel Object Representation Approach for Face Recognition, *IEEE Transactions on Image Processing*, vol.16, no.1, pp.57-68, 2007.

[54] S. Xie, S. Shan, X. Chen, *et al.*, Learned Local Gabor Patterns for Face Representation and Recognition, *Signal Processing*, vol.89, no.12, pp.2333-2344, 2009.

Captions of figures in the manuscript:

Fig. 1 General steps for vocabulary generation.

Fig. 2 Face representation via spatial pyramid based bag of features.

Fig. 3 Visualization of kernels for 1D features: (a) the Chi square kernel, (b) Histogram intersection kernel.

Fig. 4 The exact Chi square features (red line) vs. approximate features (blue line)

Fig. 5 Eight face samples randomly selected from the ORL database.

Fig. 6 Recognition rates vs. vocabulary size (a) and number of training samples (b) on the ORL dataset.

Fig. 7 Face samples of the first two subjects selected from the YALE database.

Fig. 8 Recognition rates vs. vocabulary size (a) and number of training samples (b) on the YALE dataset.

Fig. 9 Eight face samples randomly selected from the FERET database.

Fig. 10 Recognition rates vs. vocabulary size on the FERET dataset.

Fig. 11 Cumulative match curves on the FERET database when the vocabulary size is 1500 and 5000.

Biographies:

Shicai Yang is a research faculty in Hikvision Digital Technology Co., Ltd, in China. He received Bachelor degree in 2006, and his doctoral degree in Systems Engineering in 2013 from Southeast University. His research interests focus on computer vision, machine learning and deep learning.

George Bebis received the BS degree in Mathematics and the MS degree in Computer Science from the University of Crete in 1987 and 1991, respectively, and the PhD degree in Electrical and Computer Engineering from the University of Central Florida in 1996. He is currently a Professor in the Department of Computer Science and Engineering at the University of Nevada, Reno. His research interests include computer vision, image processing, pattern recognition and genetic algorithms.

Yongjie Chu received the Bachelor degree in business management and Master degree in operations research from Qufu Normal University, Rizhao, China, in 2006 and 2010, respectively. He is currently pursuing the Ph.D. degree in Systems Engineering at Southeast University, China. His research interests include machine learning, pattern recognition, and face recognition.

Lindu Zhao received his Bachelor degree in industrial psychology from Hangzhou University, China, in 1988, Master degree in computer application technology from Jiangsu University, China, in 1993, and Ph.D. degree in systems engineering from Southeast University, in 1997. Now he is a professor with Institute of Systems Engineering, Southeast University. His research interests include analysis and decision of complex systems, information fusion, machine learning and emergence management, etc.