www.s2is.org

International Journal on Smart Sensing and Intelligent Systems

ISSN 1178-5608

FACE RECOGNITION BASED ON IMPROVED SUPPORT VECTOR CLUSTERING

Yongqing Wang[1*] and Xiling Liu[2]

[1] Department of Computer Science and Applications, Zhengzhou Institute of Aeronautical Industry Management, Zhengzhou 450015, China

[2] College of Information & Business, Zhongyuan University of Technology, Zhengzhou 450007, China

Email: wyq-yongqing@163.com

Submitted: July 6, 2014   Accepted: Nov. 5, 2014   Published: Dec. 1, 2014

Abstract- Traditional methods for face recognition do not scale well with the number of training sample, which limits the wide applications of related techniques. We propose an improved Support Vector Clustering algorithm to handle the large-scale biometric feature data effectively. We prove theoretically that the proposed algorithm converges to the optimum within any given precision quickly. Compared to related state-of-the-art Support Vector Clustering algorithms, it has the competitive performances on both training time and accuracy. Besides, we use the proposed algorithm to handle classification problem, and face recognition, as well. Experiments on synthetic and real-world data sets demonstrate the validity of the proposed algorithm.

Index terms: Face recognition, biometric feature, Support Vector Clustering, kernel methods, classification, large-scale data.

I. INTRODUCTION

With the development of information technology, identification has become the frequently encountered problems in daily life, especially for the incremental need of high identification. Traditional ways of identity authentication, such as ID card, word of command and the password, tend to be forgotten, easy to crack and inconvenient to carry, have been confronted the severe challenge. To the contrary, biometric features are peculiar to the people, not easy to forge and lost, and easy carrying. Therefore, the identification technologies based on biometric recognition show unprecedented superiority, and become an international research focus in recent years.

a. Biometric recognition

Biometric recognition refers to the use of inherent physiological or behavioral characteristics of human body for personal identification in computer technology [1]. Inherent physiological characteristics, many of which are congenital, consist of face, fingerprint, palm print, iris, etc. Behavior habits, many of which are acquired, include speech, gait, handwriting, and keystroke action, etc. (seen in Figure 1).

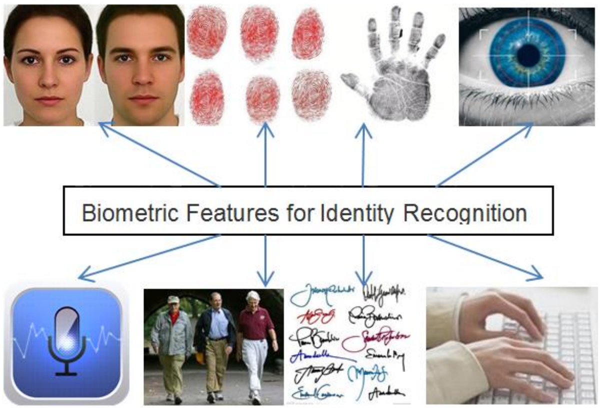

Figure 1. Biometric features recognition

The biometric features for identity authentication should involve the following three aspects.

(1) Accuracy: the biometric features can improve the accuracy of the identification.

(2) Reliability: forging of the biological characteristics is difficult.

(3) Applicability: the biometric features are feasible to application.

An ideal biometric recognition system should consist of various biological characteristics, include face, fingerprint, palm print, Iris, speech, gait, handwriting, keystroke action, etc.

Compared with other biometric features, face recognition has the following four merits.

(1) Non-mandatory: faces collection do not require special acquisition device, facial data can be got almost unconsciously, which means that the sampling way is non-mandatory.

(2) Non-contact: users do not need to directly contact with acquisition device.

(3) Concurrency: the sorting, judgment and recognition of multiple faces can be conducted in the actual application scenarios.

(4) Intuition: conforms to visual features, follows the rule “judge people with their appearance”.

Therefore, the research method discussed in this paper, is based on facial biometric identification.

b. Classification methods

Essentially, biometric identification belongs to classification problem in the field of artificial intelligence: according to the potential information provided by training sample, classifier classifies the testing data into two classes, one classified to yes, and one to no. So far, the methods of classification can be roughly divided into the following three classes.

(1) Classical statistical forecasting methods. Statistics is one of the key theoretical foundations of existing machine learning methods. In this method, the related form of parameters in the model is known, using the training samples to estimate parameters needs the form of samples’ distribution to be known, which has great limitations in applications. In addition, the traditional statistical research is the gradual theory when sample size tends to infinity, but in the actual problems, the sample size is limited, so some excellent theoretically statistical learning methods do not perform well in actual applications.

(2) Experiential nonlinear methods, such as the artificial neural network. For approximating real values, discrete values, or the vector-valued goal function, this method provides a solution with strong robustness. For certain types of problems, such as learning to explain the complex real-world sensor data, artificial neural network is known as the most effective learning way so far. It establishes the nonlinear model based on known samples, overcoming the difficulties of traditional parameter estimation method. But the lack of unified mathematical theory, excessive training data fitting and poor generalization performance are also important problems in the learning of artificial neural network.

(3) Statistical Learning Theory (SLT), which is a specialized theory for the research of machine learning law under small sample, compared with the traditional statistics. This theory developed a

new theoretical system on statistical problems with small samples, under which theory not only the requirements for asymptotic performance, but also the pursuit of the optimal results under the conditions of existing limited information are considered to obtain the statistical inference rules.

It should be noted that, during the period of 1992 to 1995, based on SLT theory, Vapnik et al. proposed successfully the Support Vector Machine (SVM) method [2-4].

Compared with the traditional statistical learning methods, SVM method has more solid mathematics theory foundation, which can effectively dealing with high-dimensional data under the condition of limited samples, and has the merits of strong generalization ability, convergence to the global optimal, non-sensitive to dimension, etc. Based on these merits, SVM has become one of the most popular research direction in the field of machine learning, and achieved widely research and application successfully in many fields, such as pattern classification, regression analysis, and estimation of density function, and so on.

The rest of this paper is organized as follows. We present a brief introduction to SVM in Section II. Section III provides a review on face recognition. We give a summary of Kernel Clustering in Section IV. Section V proposes the detailed improved SVC algorithm and experimental results, and section VI concludes this paper.

II. SUPPORT VECTOR MACHINE

For the SVM classifier, the aim of training is to find out which samples are support vectors, thus determine the decision function to predict the new samples. As a result, the number of support vector is the main factor affecting the training speed. The kernel matrix of training samples, to be computed and stored by SVM, increases with the square of training samples’ number, which becomes the bottleneck of SVM for large-scale problems and limits its application. In recent years, many scholars are looking for more rapid and efficient algorithms of SVM, which can be used to solve the problem of large sample classification. The existing methods can be roughly divided into the following categories.

a. Decomposition-based algorithms

These algorithms have the following common characteristics. Dividing the original massive quadratic programming problem into many small sub-problems, according to a certain iterative strategy, solving the sub-problems repeatedly to achieve the approximate solution, by which to gradually converge to the optimal solution of the original problem. For example, the Chunking Algorithm proposed by Boser, Guyon and Vapnik, the Decomposition Algorithm proposed by Osuna et al., the subsequent SVM^{Light} Algorithm, SMO Algorithm by Platt, and LiBSVM by Chang, all of which achieved good effect in application. At present, there are still some scholars working on these decomposition-based strategies to design efficient SVM algorithms.

b. Online learning-based algorithms

At present, there are some research results on using SVM to handle online learning problems, where data or model needs to be updated continually. The main strategy in these algorithms can be described as follows. Because of the addition of non-support vector has no effect on decision function, so for the new samples, according to the complementary slack conditions in the optimal theory, to determine whether or not to update Lagrange multipliers in the existing model. For considering all the historical data, these methods have disadvantages of unable to control the number of support vector. Only considering the sparseness of solution of the SVM, can we possibly reduce the amount of calculation, and shorten the calculation time.

c. Deformation-based algorithms

In order to improve the learning speed of standard SVM, except for seeking quick solving quadratic programming methods, there are still many algorithms for the simplified and deformed model. For example, the linear programming SVM, the quadratic loss function SVM, the least squares SVM, the proximal SVM, and so on. In order to improve the generalization ability, there are many methods based on the variant of standard SVM model. For example, fuzzy SVM, multiple kernel SVM, and prior knowledge-incorporated SVM, and so on. These algorithms made beneficial attempts from several aspects to improve the operation performance of SVM, and have achieved good effect in application. But this method has not yet formed a mature theoretical system, there are still many aspects need to be further discussed.

d. Parallel-based algorithm

In recent years, there are lots of works devoted to parallel implementation method of training SVM. For example, Collobert [5] proposed parallel hybrid method of SVM, Dong [6] used block diagonal matrix to approximate the original kernel matrix, Zanghirati [7] proposed SVM^{light} parallel method, Huang [8] proposed modular network realization method of SVM, Cao [9] proposed SMO parallel method, and so on. These methods have achieved obvious improvement for massive data problems in practical application. However, can the local optimal solution of sub-problem guarantee the global optimal solution in parallel algorithm, is still a research topic to be urgently solved.

e. Data reduction-based algorithm

The lower bound of support vector's number is linear with the number of training sample. Taking some kind of strategy, therefore, by choosing the training samples most likely to be support vector, or deleting the training samples most unlikely to be support vector, or taking the above two methods at the same time to preprocess the training set, can reduce the size of training set, and then accelerates the training process without much loss of precision. But the experimental results show that, if the proportion of the support vector in the training set is large, the generalization ability of data reduction-based algorithm is lower than the standard SVM. Asharaf [10] claimed that, even though the decomposition or data-sampling techniques [11-14] can help to reduce the complexity of the optimization problem, they are still expensive for use in applications involving large data sets.

The quadratic matrix involved in the above five types of algebraic algorithms are required to be sparse, which results in the need for large memory and training process for many practical problems, where the conditions are not met. And that the series of intuitive interpretation-based geometric algorithms, can relieve the contradiction mentioned above.

f. SVM geometry methods

Different from the algebraic algorithm of SVM, which solves the dual problem in transformed feature space, the geometry algorithm of SVM solves the original problem in the sample space. After Bennett and Crisp putting forward the idea of finding a pair of nearest points between the convex hulls of two points' sets, there are many excellent SVM geometry algorithms based on the idea of nearest points. For example, projection method-based Swap algorithm is suitable only for linear separable problem, the rapid geometry iteration algorithm proposed by Keerthi can indirectly solve the inseparable problem, and the reduced convex hull algorithm proposed by

Mavroforakis can directly solve the inseparable problem. In addition, Tsang proved the equivalence between the Minimum Enclosing Ball (MEB) and SVM, introducing an excellent approximation MEB algorithm to solve the SVM classification and regression problems for massive data, called the Core Vector Machine (CVM) [15], whose time complexity is linear with the sample size, and the space complexity is independent of the number of samples. Therefore, CVM is suitable for handling large data classification problems. Although conceptually simple, a sophisticated numerical solver is required for its implementation, which is computationally expensive when applying on large-scale problems with very large core-sets. After that, the Simpler Core Vector Machines (SCVM) [16] replaces the numerical solver with an iterative algorithm, which results in a faster training than CVM with comparable accuracy on massive data sets. However, the training process of MEB problem solved in both CVM and SCVM tend to be redundant, which results in unnecessary costs both in time and space complexities.

In this paper we develop a $(1+\varepsilon)$-approximate algorithm for computing the MEB of a given points set, with which we can achieve a fast Support Vector Clustering process to handle classification problem, and face recognition, as well.

III. FACE RECOGNITION METHODS

Face recognition is a computer technology of identity authentication by comparison and analysis of human visual characteristic information, the most commonly used means of identification in daily life, and one of the most popular research topic in pattern recognition currently.

a. Classification of face recognition methods

Face recognition research originated in the 1960’s, and emerged a lot of excellent face recognition algorithms by now. According to the development history of facial recognition technology, the methods can be roughly divided into the following kinds.

a.i The method based on geometric features

These methods are the earliest human face recognition methods [17], which commonly use the geometric feature of face features (eyes, nose and mouth), and other local shape features on the face of the geometric distribution.

Firstly, calculating the distances, edge curvatures, and angles between the specified feature points respectively to form eigenvectors. Secondly, using the Euclidean distance classifier for feature matching, recognition result can be output by the nearest neighbor method. Identification method based on geometric features is simple, rapid and easy to understand, but the extraction of stable geometric features from the image is difficult.

a.ii The method based on correlation matching

These methods include template matching method and the contour method.

(1) Template matching method: Firstly, determining facial feature points by integral projection method, extracting local feature templates (such as the eyes and nose templates). Secondly, conducting local template matching, computing the correlation coefficient to classify. In terms of recognition rate, identification method based on correlation matching is better than the method based on geometric features, but the use of multiple templates and multi-scale will increase the complexity of computing and storage.

(2) Contour method: such as strength curve reflects the face of the concavo-convex information. Contour method depicts the multilevel gray value of face image, by which to make authentication and identification for facial image. The requirements of this method on background, hair color, and surface illumination are hard to meet.

a.iii The method based on subspace analysis

Commonly-used methods of linear subspace include feature subspace, differential subspace, independent component subspace, etc. In addition, there are local characteristics analysis, factor analysis, etc. These methods are also extended to nonlinear subspace and hybrid linear space. Turk adopted Eigenface method to realize face recognition [18] in 1991. Albert put forward the Topological PCA (TPCA) method, and made the recognition rate increased. Bartlett proposed Independent Component Analysis (ICA) to recognize faces, obtained a better recognition effect [19]. Penev put forward the Local Feature Analysis method (LFA), whose face recognition effect is superior to Eigenface method. Based on Linear Discriminant Analysis (LDA), Belhumeur proposed Fisherface method in 1997, and obtained better effect [20].

In addition, the common-used algorithms for facial recognition and identification include the methods based on statistics, such as: K-L algorithm [21], singular value decomposition (SVD) [22], hidden markov method (HMM) [23], the method based on neural network [24], etc.

b. The technological process of face recognition

Face recognition system mainly includes five parts: face images collection, face detection, face images preprocessing, face images feature extraction and face images matching and identification (see Figure 2).

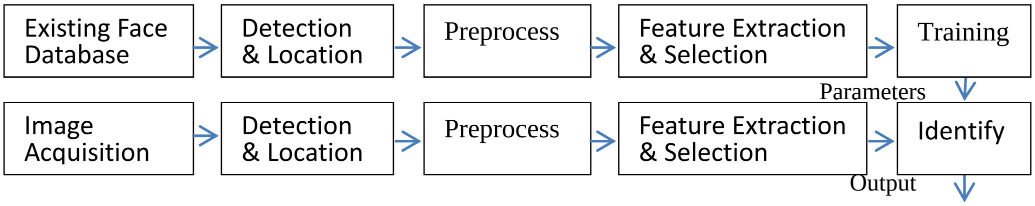

Figure 2. The flow diagram of face recognition

b.i Face images collection

Different face images can be collected by cameras, such as static images, dynamic images, different positions, different expressions, and so on.

b.ii Face detection

In practice, face detection is mainly used for the preprocess of face recognition, namely making accurate image size and location of the human faces in images. The characteristics contained in face images are very rich, such as histogram features, color features, template characteristics, structure and Haar features, etc. Face detection is to pick out the useful information, and use these characteristics to achieve face detection.

b.iii Face images preprocessing

Face image preprocessing is a process of image processing for feature extraction based on the results of face detection. Due to the limitation of various conditions and random disturbance, original images collected by system often cannot be used directly, it must be preprocessed in the early stages of image processing on gray scale correction, noise filtering and image preprocessing. For face images, the pretreatment process mainly includes the light compensation of face image, gray level transformation, histogram equalization, normalization, geometric correction, filtering and sharpening, etc.

b.iv Face images feature extraction

The characteristics used for face recognition systems include visual characteristics, the characteristics of pixels, face image transform coefficient, and face image algebra characteristics, etc. Facial feature extraction is conducted according to some features of faces. Face feature extraction method can be divided into the following two categories, feature extraction based on geometric features, and algebra features. Both of the two methods have its own advantages and shortcomings, the chosen strategy is still a challenging topic.

b.v Face images matching and recognition

Searching and matching the characteristics extracted from face images with the feature template stored in the database, when the similarity exceeds a given threshold, the matching results are output. The process can be divided into two processes, one is confirmation, a process of image comparison by one-to-one, another is identification, a process of image matching and comparison by one-to-many.

The procedures we discuss detailed in this paper are face detection and face recognition.

IV. KERNEL CLUSERING METHODS

As a kind of common tool of data analysis and unsupervised machine learning methods, clustering aims at dividing the data set into several classes (or clusters), keeping the maximum similarity between the data of each same class, and the maximum difference between the data of each pair of different class [25]. According to the basic ideas the clustering algorithms adopt, they can be roughly divided into five types [26], the partition clustering, the hierarchical clustering, the density-based clustering, the grid-based clustering, and the model-based clustering. At present, the research on the clustering algorithm is deepening, and the kernel clustering and spectral clustering are two methods that have attracted much attention in recent years [27].

The main idea of kernel clustering method is adopting a nonlinear mapping φ, such that the data points in input space can be mapped into a high-dimensional feature space, selecting appropriate

Mercer kernel function instead of nonlinear mapping of the inner product to cluster in the feature space. The kernel clustering method is universal, and has great improvement to the classical clustering methods. The adopted nonlinear mapping can increase the linear separable probability on input data points, which can achieve more accurate clustering, and faster convergence speed, as well. Under the condition the classical clustering algorithms fail, the kernel clustering algorithms can always work. The kernel trick idea in kernel clustering method can be illustrated in Figure 3 below, where the left is the original input space, and the right is the kernel-induced space.

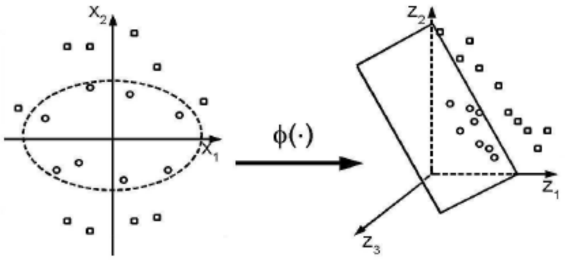

Figure 3. The kernel trick: Nonlinear problems are linear in kernel-induced space

Kernel trick can link many different kernel methods together, one of the most well-known is Support Vector Data Description (SVDD), where SVM and MEB were firstly connected.

a. Support vector data description

The idea of SVDD can be formulated as follows. Formulating Binary SVM as a QP to maximize the margin between two classes, and the consequent generalization ability is always better than the other machine learning methods.

Given a training data sets $S = \{(x_i, y_i) \mid i = 1,\dots,m\}$, where $x_i \in R^d$ and $y_i \in \{+1,\text{-}1\}$, the primal for the Binary SVM problem can be formulated as

$$\begin{aligned} &\min_{w,\rho,b,\xi_i} \; ||w||^2 + b^2 - 2\rho + C\sum_{i=1}^{m}\xi_i^{\,2} \\ &s.t. \quad y_i(w'\phi(x_i) + b) \ge \rho - \xi_i, i = 1,\dots,m. \end{aligned} \qquad (1)$$

The corresponding dual is

$$\begin{aligned} &\min_{\alpha_i} \; \sum_{i,j=1}^{m}\alpha_i\alpha_j(y_i y_j k(x_i, x_j) + y_i y_j + \frac{\delta_{ij}}{C}) \\ &s.t. \; \sum_{i=1}^{m}\alpha_i = 1, \; \alpha_i \ge 0, \; i = 1,\dots,m, \end{aligned} \qquad (2)$$

Where δ_{ij} is the Kronecker delta function, defined as following

$$\delta_{ij} = \begin{cases} 1, & if\ i = j, \\ 0, & if\ i \neq j. \end{cases} \qquad (3)$$

We denote the pair (x_i, y_i) as z_i to simplify the notation. Introducing a modified feature map $\tilde{\phi}(z_i) = [y_i \phi'(x_i)\ \ y_i\ \ \frac{e'_i}{\sqrt{C}}]'$ and the associated kernel function $\tilde{k}(z_i, z_j) = y_i y_j k(x_i, x_j) + y_i y_j + \frac{\delta_{ij}}{C}$, then the dual of Binary SVM with form (2) can be rewritten as

$$\begin{aligned} \min_{\alpha_i} \quad & \sum_{i,j=1}^{m} \alpha_i \alpha_j \tilde{k}(z_i, z_j) \\ s.t. \quad & \sum_{i=1}^{m} \alpha_i = 1,\ \alpha_i \geq 0,\ \ i = 1,\ldots,m. \end{aligned} \qquad (4)$$

A simple process of SVDD can be illustrated in Figure 4 below. The left figure presents the decisive curve of banana-shaped data without outliers under the parameter C = 1 in two-dimensional plane. The solid points represent the support vectors, the dotted line represents the boundary of the data description, and the grey value represents the distance from the center of sphere, where the deeper means the closer to the center. We can see that, only three support vectors are needed to describe all of the data set. Introducing a new outlier in the right picture (located with an arrow), large change occurred in the decisive curve, where the new outlier comes into the support vector of the new decisive curve.

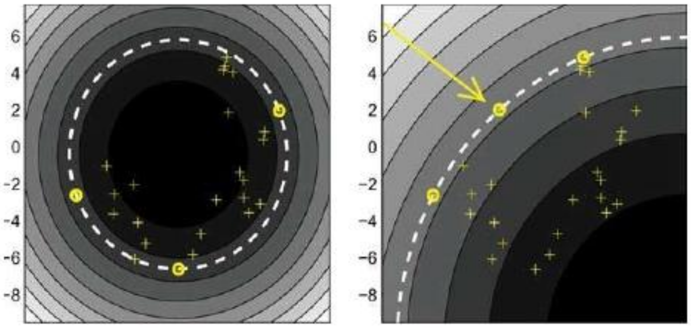

Figure 4. SVDD without outliers (left) and SVDD with outliers (right)

b. Support vector clustering

Support Vector Clustering (SVC) belongs to the method of kernel clustering, whose foundational tool for clustering is Support Vector Machine (SVM) [28]. Based on the Support Vector Domain Description (SVDD) algorithm [29], Ben-Hur (2001) proposed an unsupervised nonparametric clustering algorithm, called SVC [30], whose basic idea is formulated as follows.

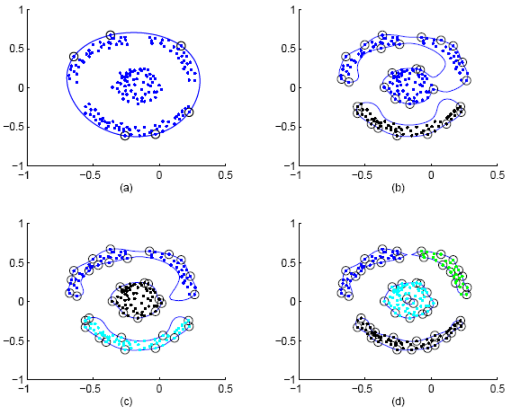

Figure 5. The clustering procedure of SVC on 183 points with C=1

The data points are mapped from input space to a high dimensional feature space using a Gaussian kernel, and seeking for the smallest sphere that encloses all the image of the data in feature space. This sphere is mapped back to data space, where it forms a set of contours which enclose the data points. These contours are interpreted as cluster boundaries. Points enclosed by each separate contour are associated with the same cluster. As the width parameter of the Gaussian kernel is decreased, the number of disconnected contours in data space increases, leading to an increasing number of clusters. Since the contours can be interpreted as delineating the support of the underlying probability distribution, SVC algorithm can be viewed as one identifying valleys in this probability distribution [28, 29]. The shape of the enclosing contours in input space is governed by two parameters: q, the scale parameter of the Gaussian kernel, and C, the soft margin constant. Figure 5 above demonstrates the effects of these two parameters [30], where the support vectors are designated by small circles, and cluster assignments are represented by different grey scales of the data points. (a): q = 1 (b): q = 20 (c): q = 24 (d):q = 48.

SVC algorithm has two significant advantages. The first is SVC can handle clusters with the boundary in arbitrary shape, and the second is can analyze the noise data points and separate the overlapping clusters, in which circumstances many other clustering algorithms are hard to tackle. But there is still a bottleneck of weak scalability with the number of training sample data size in SVC algorithm, so many new SVC algorithms are designed to improve the computing efficiency [30, 31].

V. IMPROVED MEB ALGORITHM FOR SVC

Utilizing the concept of Core Set (seen in Figure 6), the MEB algorithm adopted in references [30, 31, 34-36] has the time and space complexities of $O(\frac{m}{\varepsilon^2}+\frac{1}{\varepsilon^4})$ and $O(\frac{1}{\varepsilon^2})$, respectively.

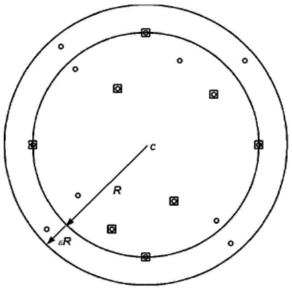

Figure 6. Inner circle is the exact MEB contains all the square points, whose $(1+\varepsilon)$ expansion contains all the points, so the square points is the core set of all points

However, we found that the final core vectors obtained for formulating the final decision function is always more than necessary in implementations, which results in some redundancies in the process of storing and training. This claim can be demonstrated clearly in Figure 6 above, where the number of squire points is larger than 3, the necessary points' number in 2-D space to determine the circle at most.

a. The SMEB algorithm

We formulate the SMEB algorithm in Table 1 below.

Table 1. The improved MEB algorithm

SMEB	Outputs a $(1+\varepsilon)$-approximation of MEB(S)
1	Given an $\varepsilon > 0$, pick any $p \in S,\ S_0 \leftarrow \{p\}$; Outputs S_0, c_0, R_0;
2	Terminate if there is no training point z such that $\varphi(z)$ falls outside the $(1+\varepsilon)$-ball $B(c_t, (1+\varepsilon)R_t)$;
3	Find $z \in S \setminus S_t$ such that $\varphi(z)$ is furthest away from c_t, set $S_t = S_t \cup \{z\}$;
4	Find new MEB(St), set $c_{t=}c_{MEB(S_t)}, R_t = r_{MEB(S_t)}$, and increment t by 1, if $t < 48/\varepsilon^2 - 2,$ go back to Step 2; otherwise go to Step 5;
5	Find $y \in S_t$ such that $\|\varphi(y) - c_t\| < R_{t,}$ set $S_t = S_t \setminus \{y\}$;
6	Increment t by 1 and go back to Step 2.

b. The analysis on time and space complexities

We conclude the analysis on time and space complexities in Theorems 1 to 4 below.

Theorem 1. In the process of SMEB Algorithm, when the iteration satisfies $i \geq \frac{48}{\varepsilon^2} - 2$, if one point q falls into the interior of current MEB, i.e., $\|q - c_i\| < r_i$, it will fall into the interior of subsequence MEBs, i.e., $\|q - c_{i+j}\| < r_{i+j}$, $j \in Z^+$.

Theorem 2. SMEB Algorithm can achieve a $(1+\varepsilon)$ -approximate MEB for training data set S within $O(\frac{1}{\varepsilon^2})$ iterations.

Theorem 3. In the iterations of SMEB Algorithm, there exists a subset $P \subset S$, whose points are at distance at most $(1+\varepsilon)r_{B(S)}$ from center $c_{B(S)}$, and the size of P is $O(\min\{\frac{1}{\varepsilon^2}, d\})$.

Theorem 4. The time and space complexities of SMEB Algorithm are $O(d^4 + d^2 m + \frac{d^3}{\varepsilon^2} + \frac{dm}{\varepsilon^2})$ and $O(\min\{\frac{1}{\varepsilon^4}, d^2\})$.

The detailed proofs of these theorems are omitted here for conciseness, interested readers can refer to Wang [37, 38].

c. Experiments on synthetic data

Experiments are performed on five synthetic data sets, which follow a uniform distribution on the interval (0, 10) (Table 2). All experiments do not adopt the probabilistic speedup method utilized in CVM for simplicity. We use Matlab 7.0 on a PC with Pentium-4 3.20 GHz CPU, 1GB of RAM running Windows XP to implement our experiments.

Table 2: Data sets used in the experiments

data sets	**data 1**	**data 2**	**data 3**	**data 4**	**data 5**
dimension	2	2	2	2	2
number	10	100	1000	10000	100000

We compare CVM, SCVM and SMEB on Optimum Bias Ratio and Training Time at different values of ε on the five data sets, where Optimum Bias Ratio (OBR) is defined as $OBR = \frac{\|\text{experimental value - optimum}\|}{\|\text{optimum}\|}$. Fig. 7 – Fig. 11 demonstrate the different performances with different values of ε for different data listed in Table 2, from which we can see that SMEB algorithm is usually faster than CVM and SCVM on the same ε with comparable accuracies, which implies that SMEB is more suitable for solving larger data problems (seen in Figure 11).

We conclude from the results that all algorithms have low OBR's for $\varepsilon \in [10^{-6}, 10^{-4}]$, and SMEB has accuracies comparable with the other two algorithms, especially when ε gets small enough. When utilized to handle larger data set, e.g., data 5, the SMEB algorithm is usually faster than CVM and SCVM for the same value of ε, with comparable accuracies, which implies that SMEB is more suitable for solving larger data problems (Fig. 10). When ε decreases, the SMEB tends to be closer to the exact optimal solution, but at the expense of higher time and space complexities. Such a tradeoff between efficiency and approximation quality is typical of all approximation schemes.

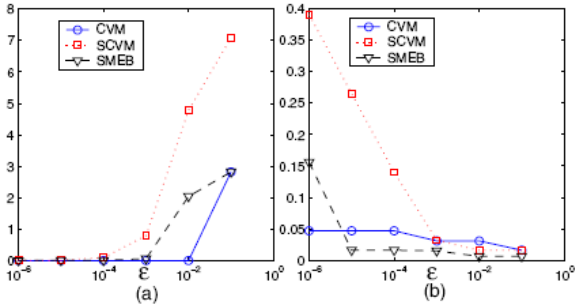

Figure 7. Performance with different value of ε for data 1, (a) for Optimum Bias Ratio (%), (b) for training time (s)

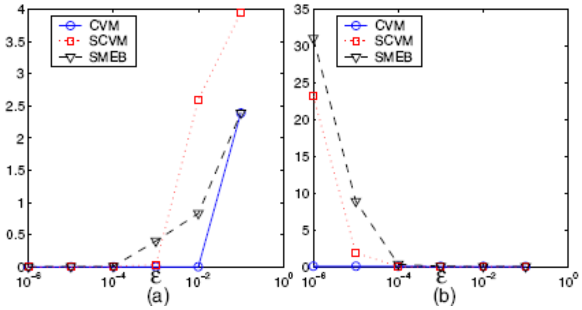

Figure 8. Performance with different value of ε for data 2, (a) for Optimum Bias Ratio (%), (b) for training time (s)

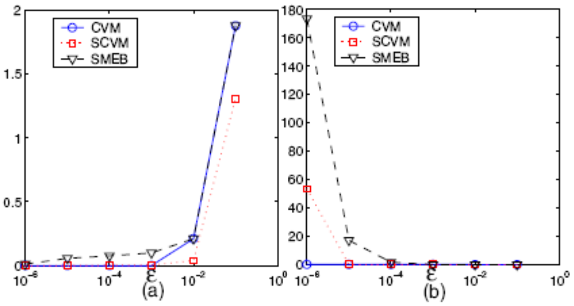

Figure 9. Performance with different value of ε for data 3, (a) for Optimum Bias Ratio (%), (b) for training time (s)

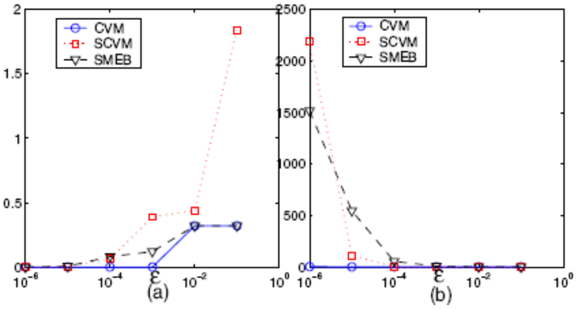

Figure 10. Performance with different value of ε for data 4, (a) for Optimum Bias Ratio (%), (b) for training time (s)

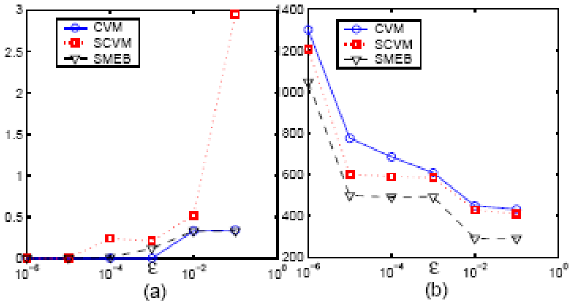

Figure 11. Performance with different value of ε for data 5, (a) for Optimum Bias Ratio (%), (b) for training time (s)

d. Experiments on face recognition

We implemented the proposed SMEB algorithm to handle kernel clustering problem, and then we can deal with face recognition in two steps.

d.i Face detection

According to the geometry features and the gray value of faces, we can accordingly locate the faces in vary backgrounds (seen in Figure 12, 13).

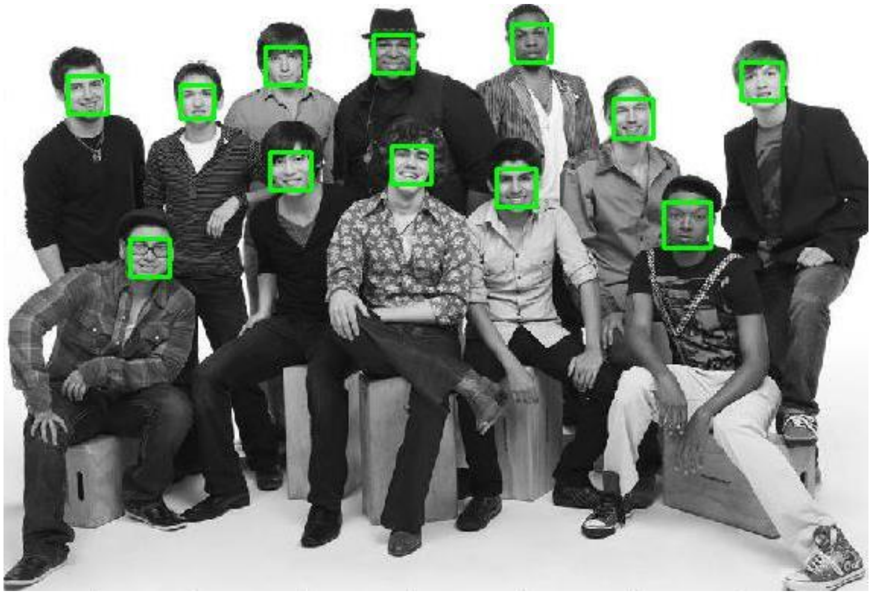

Figure 12. Face detection without misdetection

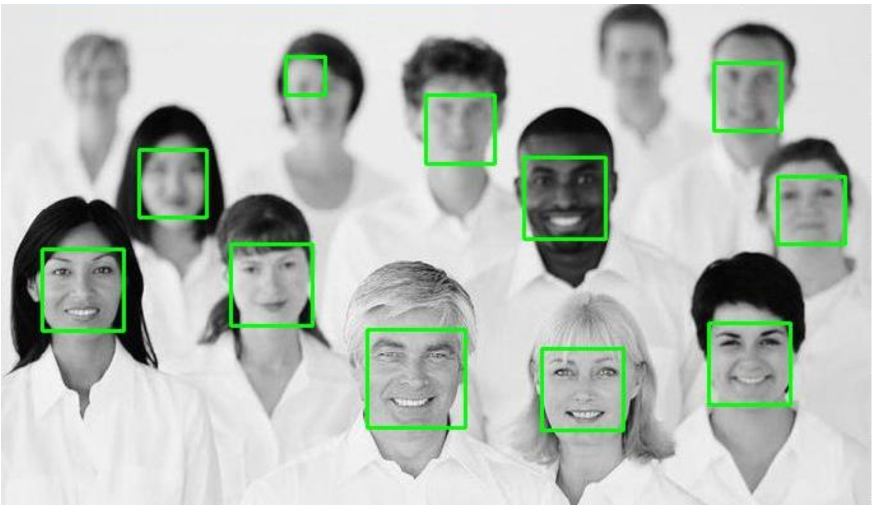

Figure 13. Face detection with minor misdetection

d.ii Face recognition

We collect the detected faces to construct the training and testing libraries, where the related face images are downloaded from internet, consist of 10 Hollywood movie stars, total 30 images (seen in Figure 14, 15), where 20 images are chosen for training, 10 images for testing. The images are preprocessed to be 175×222 pixels for unification. The testing results are shown in Figure 16.

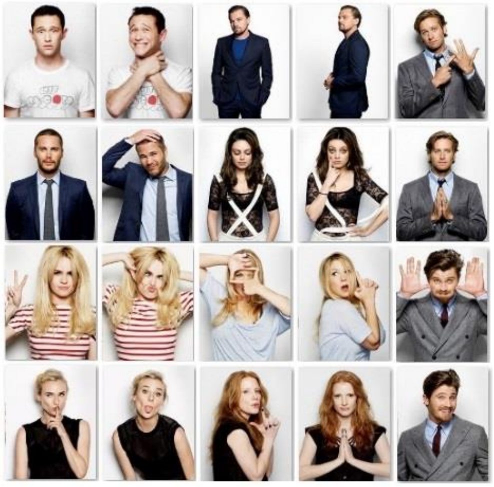

Figure 14. Training images

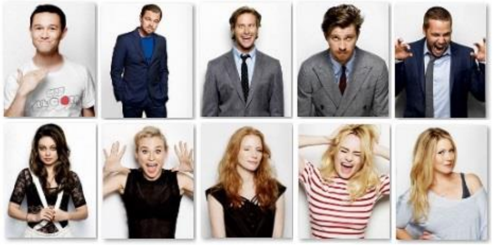

Figure 15. Testing images

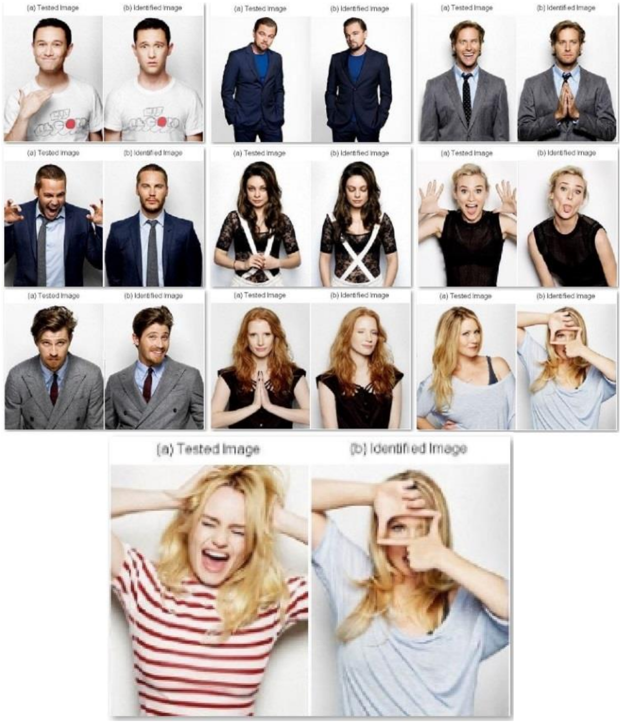

Figure 16. Testing results: odd columns stand for tested images, even columns stand for identified images, and bottom row stands for misrecognition (1 out of 10)

VI. CONCLUSIONS

In this paper we develop a (1+ε)-approximate algorithm for computing the MEB of a given points set within any precision in $O(1/\varepsilon)$ iterations. We prove theoretically that the proposed SMEB algorithm has time complexity of $O(\frac{m}{\varepsilon^2}+\frac{1}{\varepsilon^3})$, which is linear in the number of training samples m for a fixed ε, and space complexity of $O(\frac{1}{\varepsilon^2})$, which is independent of m for a fixed ε. Compared to CVM and SCVM algorithms, it has the competitive performances in both training time and accuracy. Besides, by use of the proposed SMEB algorithm, we can achieve a fast Support Vector Clustering process to effective handle Classification problem and Face Recognition. Experiments on both synthetic and real-world data sets demonstrate the validity of the proposed algorithm.

ACKNOWLEDGMENTS

This work was supported in part by the National Natural Science Foundation of China (Grant No. 41001235), the Aeronautical Science Foundation of China (Grant No. 2011ZC55005), the Project of Henan Provincial Audit Office (Grant No. 20120922), the Scientific & Technological Research Key Project of Henan Provincial Education Office (Grant No. 13A520404), the Scientific & Technological Project of Henan Provincial Science & Technology Office (Grant No. 132102210468), and the Project of Science and Technology Bureau of Zhengzhou (Grant No. 20120435 and 20130713).

The authors are grateful to the anonymous referees for their valuable comments and suggestions to improve the presentation of this paper.

REFERENCES

[1] Jie Tian and Xin Yang, Biometric identification technology theory and applications, Electronic Industry Press, Beijing, 2005.

[2] C. Cortes and V. Vapnik, “Support vector networks”, Machine Learning, Vol. 20, 1995, pp. 273-297.

[3] V. Vapnik, “Three Fundamental Concepts of the Capacity of Learning Machines”, Physica A, Vol. 200, 1993, pp. 537-544.

[4] V. Vapnik, Statistical Learning Theory, J. Wiley, New York, 1998.

[5] R. Collobert, S. Bengio and Y. Bengio, “A Parallel Mixture of SVMs for Very Large Scale Problems”, Neural Computation, Vol. 14, No. 5, 2002, pp. 1105-1114.

[6] J. X. Dong, A. Krzyzak and C. Y. Suen, “A fast parallel optimization for training support vector machine”, Proc. of the 3rd Int. Conf. Machine Learning Data Mining, Vol. LNAI 2734, 2003, pp. 96-105.

[7] G. Zanghirati and L. Zanni, “A parallel solver for large quadratic programs in training support vector machines”, Parallel Comput., Vol. 29, No. 4, 2003, pp. 535-551.

[8] B. H. Guang, K. Z. Mao, C. K. Siew and D. S. Huang, “Fast modular network implementation for support vector machines”, IEEE Trans. Neural Netw., Vol. 16, No. 6, 2005, pp. 1651-1663.

[9] L. J. Cao, S. S. Keerthi and J. Q. Zhang, “Parallel Sequential Minimal Optimization for the Training of Support Vector Machines”, IEEE Trans. Neural Netw., Vol. 17, No. 4, 2006, pp. 1039-1049.

[10] S. Asharaf, M. N. Murty and S. K. Shevade, “Multiclass Core Vector Machine”, Proc. of ICML’07, 2007, pp. 41-48.

[11] Francesco Orabona, “Better Algorithms for Selective Sampling”, in Proceedings of ICML’11, 2011, pp. 433-440.

[12] Y. Liu, “Combining integrated sampling with SVM ensembles for learning from imbalanced datasets”, Information Processing and Management, Vol. 47, No. 4, 2011.

[13] M. Volpi, “Memory-Based Cluster Sampling for Remote Sensing Image Classification”, IEEE Transactions on Geoscience and Remote Sensing, Vol. 50, No. 8, 2012.

[14] Abhimanyu Das and David Kempe, “Submodular meets Spectral: Greedy Algorithms for Subset Selection, Sparse Approximation and Dictionary Selection”, Proceedings of ICML’11, 2011, pp. 1057-1064.

[15] I. W. Tsang, J. T. Kwok and P. M. Cheung, “Core vector machines: Fast SVM training on very large data sets”, Journal of Machine Learning Research, Vol. 6, 2005, pp. 363-392.

[16] I. W. Tsang, A. Kocsor and J. T. Kwok, “Simpler Core vector machines with Enclosing Balls”, Proc. 24th Int. Conf. Machine Learning, 2007, pp. 911-918.

[17] Ziquan H., “Algebraic feature extraction of image for recognition”, Pattern Recognition, Vol. 24, No. 3, 1991, pp. 211-219.

[18] Turk M. and Penfland A., “Face recognition using eigenfaces”, Proc. IEEE Conf. on Computer Vision and Pattern Recognition, 1991, pp. 586-591.

[19] Bartlett M. S. and Movellan J. R., “Face Recognition by Independent Component Analysis”, IEEE Transactions on Neural Network, Vol. 13, No, 6, 2002, pp. 1450-1464.

[20] Belhumeur P. N., Hespanha J. P., and Kriegman D. J., “Eigenfaces vs. Fisherfaces: Recognition Using Class Specific Linear Projection”, IEEE Transactions no Pattern Analysis and Machine Intelligence, Vol. 19, No. 7, 1997, pp. 711-720.

[21] Zhaoqi Bian and Xuegong Zhang, Pattern Recognition, Tsinghua Press, Beijing, 2004.

[22] Yunhong Wang, Tieniu Tan, “Face Identification Based on Singular Value decomposition and Data Fusion”, CHINESE JOURNAL OF COMPUTERS, Vol. 23, No. 6, 2000, pp. 649-653.

[23] Nefian A. V., Hayes M. H., “Hidden Markov Models for Face Recognition”, IEEE Intern. Conf. on Acoustics, Speech and Signal Processing, 1998, pp. 2721-2724.

[24] Shuxian Zhu and Renjie Zhang, “Face recognition base on RBF neural network”, OPTICAL INSTRUMENTS, Vol. 30, No. 2, 2008, pp. 31-33.

[25] Rui Xu and Wunsch, D., “Survey of Clustering Algorithms”, IEEE Transaction on Neural Networks, Vol. 16, No. 3, 2005, pp. 645-678.

[26] Han J. and Kamber M., Data Mining: Concepts and Techniques, Second Edition, Morgan Kaufmann, San Francisco, 2006.

[27] Filippone M., Camastra F., and Masulli F., “A Survey of Kernel and Spectral Methods for Clustering”, Pattern Recognition, Vol. 41, No. 1, 2008, pp. 176-190.

[28] Burges C. J. C., “A Tutorial on Support Vector Machines for Pattern Recognition”, Data Mining and Knowledge Discovery, Vol. 2, No. 2, 1998, pp. 121-167.

[29] Tax D. M. J., Duin R. P. W., “Support Vector Domain Description”, Pattern Recognition Letters, Vol. 20, No. 11-13, 1999, pp. 1191-1199.

[30] Ben-Hur A., Horn D., Siegelmann H. T., Vapnik V., “Support Vector Clustering”, Journal of Machine Learning Research, Vol. 2, No. 12, 2001, pp. 125-137.

[31] Scholkopf B., Williamson R., Smola A., Shawe-Taylor J., Platt J., “Support Vector Method for Novelty Detection”, Advances in Neural Information Processing System, Vol. 12, 2000, pp. 582-588.

[32] Lee J. and Lee D., “An Improved Cluster Labeling Method for Support Vector Clustering”, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 27, No. 3, 2005, pp. 461-464.

[33] Camastra F. and Verri A., “A Novel Kernel Method for Clustering”, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 27, No. 5, 2005, pp. 801-805.

[34] Tsang I. W., Kwok J. T., and Cheung P. M., “Core vector machines: Fast SVM training on very large data sets”, Journal of Machine Learning Research, Vol. 6, 2005, pp. 363-392.

[35] S. Asharaf, M. N. Murty, and S. K. Shevade, “Multiclass Core Vector Machine”, in Proc. of ICML'07, 2007, pp. 41-48.

[36] I. W. Tsang, J. T. Kwok, J. M. Zurada, “Generalized Core Vector Machines”, IEEE Trans. on Neural Networks, Vol. 17, No. 5, 2006, pp. 1126-1140.

[37] Y. Q. Wang, Y. Li, and L. Chang, “Approximate Minimum Enclosing Ball Algorithm with Smaller Core Sets for Binary Support Vector Machine”, in Proc. CCDC’2010, 2010, pp. 3404-3408.

[38] Y. Q. Wang, X. T. Niu, and L. Chang, “Multiclass Core Vector Machine with Smaller Core Sets”, in Proc. CCDC’2010, 2010, pp. 525-530.