

## Research Article

# A Support Vector Data Description Committee for Face Detection

Yi-Hung Liu,<sup>1</sup> Yung Ting,<sup>1</sup> Shian-Shing Shyu,<sup>2</sup> Chang-Kuo Chen,<sup>2</sup>  
Chung-Lin Lee,<sup>2</sup> and Mu-Der Jeng<sup>3</sup>

<sup>1</sup> Department of Mechanical Engineering, Chung Yuan Christian University, Chungli 320, Taiwan

<sup>2</sup> Institute of Nuclear Energy Research, Atomic Energy Council, Taoyuan 325, Taiwan

<sup>3</sup> Department of Electrical Engineering, National Taiwan Ocean University, Keelung 200, Taiwan

Correspondence should be addressed to Yi-Hung Liu; [lyh@cycu.edu.tw](mailto:lyh@cycu.edu.tw)

Received 4 November 2013; Accepted 24 December 2013; Published 19 February 2014

Academic Editor: Chung-Hao Chen

Copyright © 2014 Yi-Hung Liu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Face detection is a crucial prestage for face recognition and is often treated as a binary (face and nonface) classification problem. While this strategy is simple to implement, face detection accuracy would drop when nonface training patterns are undersampled. To avoid these problems, we propose in this paper a one-class learning-based face detector called support vector data description (SVDD) committee, which consists of several SVDD members, each of which is trained on a subset of face patterns. Nonfaces are not required in the training of the SVDD committee. Therefore, the face detection accuracy of SVDD committee is independent of the nonface training patterns. Moreover, the proposed SVDD committee is also able to improve generalization ability of the original SVDD when the face data set has a multicluster distribution. Experiments carried out on the extended MIT face data set show that the proposed SVDD committee can achieve better face detection accuracy than the widely used SVM face detector and performs better than other one-class classifiers, including the original SVDD and the kernel principal component analysis (Kernel PCA).

## 1. Introduction

Face detection plays a key role in human-computer interaction since it is a prior step to face recognition. Given an image, the objective of face detection is to locate the faces in the image and return the location of each face. Due to complex backgrounds, variations in facial details, and lighting conditions, face detection has been considered one of the most challenging pattern recognition problems. A large body of works has been presented to tackle this difficult problem in the past decades, which had been nicely surveyed in [1–3].

*1.1. Related Works.* Appearance-based approach has dominated the recent advances in face detection [3], which consists of two main steps: first, a sliding window is used for scanning the whole image in a serial fashion [4, 5]; then a preselected face detector performs a binary (face and nonface) classification task on each window image to verify whether a face is present or not in each window. Previous works

based on this approach were focused on addressing issues such as (1) exploiting robust features, for example, Haar-like features [6], Bayesian feature [7], spectral histogram [8], local binary pattern- (LBP-)based spatial histogram [9], and principal component analysis (PCA) and its nonlinear version: Kernel PCA [10], (2) seeking face detectors with good generalization ability, such as neural networks (NNs) [11–13], Bayesian classifier [7, 14], and support vector machine (SVM) [4, 8, 10, 15–25], and (3) further improving efficiency of a given face detector by boosting-based techniques, in which the AdaBoost is probably the most famous and has been used in the Viola-Jones face detector [6, 26]. In this work, we aim at dealing with the second issue, the face detector design, and propose a novel face detector called support vector data description (SVDD) committee.

*1.2. Problem Description.* Appearance-based face detection typically treats the face detection task as a binary classification problem [2, 3, 5]: face and nonface classification. Accordingly, two-class classifiers were adopted. According to previous

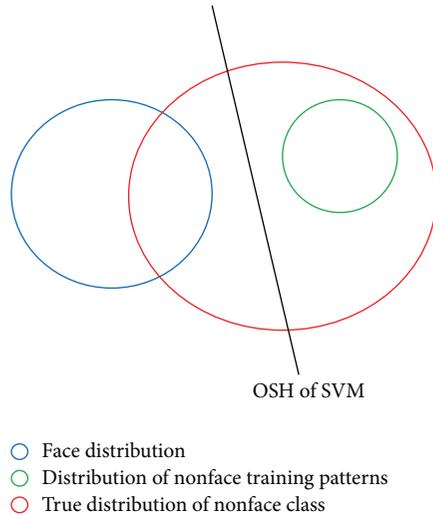


FIGURE 1: This figure shows that if an SVM is trained on a training set in which the nonface training set does not represent the nonface class, some of nonface testing patterns that lie inside of the red boundary would be classified as face patterns (i.e., falling into the wrong side of the SVM's OSH), resulting in a high false positive rate in the testing stage.

works, the two-class classifier SVM by Vapnik [27] has been frequently used as the face detector. The success of SVM in face detection should be attributed to the use of kernel tricks and its learning strategy based on structural risk minimization (SRM). However, SVM may still suffer from a critical problem when applying it to face detection: the high false negative rate due to unrepresentative nonface training patterns, described as follows.

To train an SVM, one has to prepare a training set containing face (positive) and nonface (negative) patterns in prior [22]. The training set is then used to train an SVM to find an optimal separating hyperplane (OSH) with maximal margin of separation in a kernel-induced feature space. However, compared with face class, nonface class's distribution has a relatively large variation due to rich nonface patterns. In other words, it is easy to collect a set of face training patterns which can represent the face class; however, collecting a set of nonface training patterns which are representative enough is difficult because any patterns that do not belong to the face class are nonface patterns. In other words, the nonface class is most likely undersampled: distribution of the collected nonface patterns used for training is not identical to the true distribution of the nonface class. If an SVM is trained by such a training set in which the nonface training set is unrepresentative, many nonface patterns would fall into the wrong side of the OSH in testing stage, resulting in numerous false positives, as illustrated in Figure 1.

**1.3. Presented Work.** To avoid this critical problem, we choose to adopt the one-class learning strategy to deal with aforementioned undersampled problem of nonface training patterns. One-class learning is to solve the conventional two-class classification problems where one of the two classes is undersampled, or only the data of one single class can

be available for training [28–30]. One-class classifiers are to find a compact description for a specific class (usually being referred to as *target class*) and can be built on just one single class, the target class. In this work, we treat faces as targets, while nonfaces as outliers. The decision boundary of a one-class classifier is then used to distinguish targets from outliers.

Various one-class classifiers have been proposed, such as the linear programming (LP) approach [31], single-class minimax probability machine (MPM) [32], Gaussian mixture model (GMM) [28], one-class SVM [33], SVDD [34, 35], and the Kernel PCA for novelty detection [36]. In this work, we design a robust face detector based on the SVDD of Tax and Duin.

SVDD is a kernel method for novelty detection. Given a target training set, SVDD maps the set into a higher-dimensional kernel-induced feature space and then finds a minimum-volume hypersphere that can enclose all or most of the mapped target data in this feature space. Due to the use of kernel tricks, the sphere boundary in the feature space becomes a flexible one in the original input space, thus being able to fit any irregularly shaped target data sets. This is particularly useful for face detection since face patterns are in general nonlinearly distributed [37, 38]. Recently, success of SVDD has been shown in a variety of novelty detection problems, such as anomaly detection in hyperspectral images [39], defect inspection [40], and novel percept detection for a vision-guidance mobile robot [41].

However, SVDD still has its limits. When a target training set is not a compact set but is formed by a set of disjoint clusters in the data space, the generalization performance of SVDD would drop significantly, as pointed out by Tax and Duin [34]. Unfortunately, face patterns from different individuals form a multicluster distribution in the space of patterns. Thus, using one single SVDD to discriminate faces from nonfaces may not be adequate. To solve this problem, we propose in this paper an SVDD committee.

The training of the proposed SVDD committee consists of two stages. In the first stage, a given face training set is automatically partitioned into disjoint clusters using fuzzy *c*-means (FCM) algorithm [42] and a partitioning entropy-based best cluster selection criterion [43]. The face patterns in each cluster form a compact face subset. Each face subset is then used to train a unique SVDD, which is the second stage. In addition, the decision boundary of SVDD often encloses the face training patterns tightly, which limits the generalization performance for testing faces. To improve the performance, we also modify the original decision function of SVDD so that the decision boundary of SVDD is enlarged. By doing so, face acceptance rate can be improved. Finally, if there are *c* face clusters, *c* SVDDs (members) will be trained. In the testing stage, each trained SVDD serves as a committee member. An unseen pattern is classified as a face pattern if it is accepted by any of the *c* SVDDs. Details of the SVDD will be introduced in Section 3.

The rest of this paper is organized as follows. In Section 2, we first review the basics of SVDD. Then, the SVDD committee will be introduced in Section 3. Results and discussions are provided in Section 4. Finally, conclusions are drawn in Section 5.

## 2. SVDD

Let  $T = \{\mathbf{x}_i \in \mathbf{R}^d\}_{i=1}^N$  be a face training set, where  $\mathbf{x}_i$  are face training patterns. SVDD maps the training patterns into a higher dimensional space  $F$  using a nonlinear map  $\phi : \mathbf{R}^d \rightarrow F$  and then finds a minimum-enclosing hypersphere with center  $\mathbf{a}_F$  and radius  $R$  in  $F$ , which can be formulated as the optimization problem as

$$\begin{aligned} & \text{Minimize} \quad R^2 + C \sum_{i=1}^N \xi_i \\ & \text{subject to} \quad \|\phi(\mathbf{x}_i) - \mathbf{a}_F\|^2 \leq R^2 + \xi_i, \\ & \quad \quad \quad \xi_i \geq 0, \quad \forall i = 1, \dots, N, \end{aligned} \quad (1)$$

where the penalty weight  $C$  is user specified and  $\xi_i$  are slack variables representing training errors. Taking the partial derivatives  $\partial L / \partial R = 0$ ,  $\partial L / \partial \mathbf{a}_F = 0$  and  $\partial L / \partial \xi_i = 0$ , where

$$L = R^2 + C \sum_{i=1}^N \xi_i - \sum_i \alpha_i (R^2 + \xi_i - \|\phi(\mathbf{x}_i) - \mathbf{a}_F\|^2) - \sum_i \beta_i \xi_i \quad (2)$$

is the Lagrangian function and  $\alpha_i$  and  $\beta_i$  are nonnegative Lagrange multipliers, and substituting the results back into  $L$  yield the dual constrained optimization problem as follows:

$$\begin{aligned} & \text{Maximize} \quad \sum_{i=1}^N \alpha_i K(\mathbf{x}_i, \mathbf{x}_i) - \sum_{i=1}^N \sum_{j=1}^N \alpha_i \alpha_j K(\mathbf{x}_i, \mathbf{x}_j) \\ & \text{subject to} \quad \sum_{i=1}^N \alpha_i = 1, \\ & \quad \quad \quad 0 \leq \alpha_i \leq C, \quad \forall i = 1, \dots, N, \quad C \in \left[ \frac{1}{N}, 1 \right], \end{aligned} \quad (3)$$

where  $K(\mathbf{x}, \mathbf{y}) = \phi(\mathbf{x}) \cdot \phi(\mathbf{y})$  is the kernel function. In this paper, the Gaussian function

$$K(\mathbf{x}, \mathbf{y}) = \exp\left(\frac{-\|\mathbf{x} - \mathbf{y}\|^2}{2\sigma^2}\right) \quad (4)$$

is used as the SVDD kernel, where  $\sigma$  is the parameter of the Gaussian kernel. According to the Kuhn-Tucker conditions, (1) the data points with  $\alpha_i = 0$  are inside of the hypersphere, (2) the data points whose  $0 < \alpha_i < C$  are on the sphere boundary, and (3) the data points whose  $\alpha_i = C$  fall outside the sphere and have nonzero  $\xi_i$ . The data points with  $\alpha_i > 0$  are support vectors (SVs). Further, the SVs with  $0 < \alpha_i < C$  are called unbounded SVs (UBSVs), while the SVs with  $\alpha_i = C$  are called bounded SVs (BSVs). The center of the sphere is spanned by the mapped SVs,

$$\mathbf{a}_F = \sum_{i=1}^{N_s} \alpha_i \phi(\mathbf{x}_i), \quad (5)$$

where  $N_s$  is the number of SVs. The sphere radius  $R$  is determined by taking any  $\mathbf{x}_k \in$  UBSVs and calculating the distance from its image to the center  $\mathbf{a}_F$  as follows:

$$\begin{aligned} R &= \sqrt{\|\phi(\mathbf{x}_k) - \mathbf{a}_F\|^2} \\ &= \sqrt{1 - 2 \sum_{\mathbf{x}_i \in \text{SVs}} \alpha_i K(\mathbf{x}_i, \mathbf{x}_k) + \sum_{\mathbf{x}_i \in \text{SVs}} \sum_{\mathbf{x}_j \in \text{SVs}} \alpha_i \alpha_j K(\mathbf{x}_i, \mathbf{x}_j)}. \end{aligned} \quad (6)$$

For Gaussian kernel,  $K(\mathbf{x}, \mathbf{x}) = \phi(\mathbf{x}) \cdot \phi(\mathbf{x}) = 1$ , for all  $\mathbf{x} \in \mathbf{R}^d$ . Mapping the sphere boundary  $f(\mathbf{x}) = 0$  back into the original space  $\mathbf{R}^d$  yields a flexible boundary that encloses the face training set. The free parameter  $\sigma$  controls the tightness of the boundary: the smaller the  $\sigma$  is, the tighter the boundary is. However, the  $\sigma$  cannot be too small; otherwise the boundary will be too tight to get a satisfactory face acceptance rate for unseen patterns. The decision function of SVDD is given by

$$\begin{aligned} f(\mathbf{x}) &= \|\phi(\mathbf{x}) - \mathbf{a}_F\|^2 - R^2 \\ &= (1 - R^2) - 2 \sum_{i=1}^{N_s} \alpha_i K(\mathbf{x}, \mathbf{x}_i) + \sum_{i,j=1}^N \alpha_i \alpha_j K(\mathbf{x}_i, \mathbf{x}_j). \end{aligned} \quad (7)$$

If  $f(\mathbf{x}) \leq 0$ , the test pattern  $\mathbf{x}$  is accepted as a face pattern, rejected as a nonface pattern otherwise.

## 3. SVDD Committee

### 3.1. Training

*Stage 1* (partitioning). The first stage is to partition the face training set  $T = \{\mathbf{x}_i \in \mathbf{R}^d\}_{i=1}^N$  into clusters which are disjoint. Here, the fuzzy  $c$ -means (FCM) algorithm is employed to accomplish this task, which solves the optimization problem as follows:

$$\text{minimize} \quad J(T; \mathbf{U}, \mathbf{V}) = \sum_{j=1}^c \sum_{i=1}^N u_{ji}^m \|\mathbf{x}_i - \mathbf{v}_j\|^2 \quad (8)$$

subject to the constraints

$$\begin{aligned} & \sum_{j=1}^c u_{ji} = 1, \quad \forall i = 1, \dots, N, \\ & 0 < \sum_{i=1}^N u_{ji} < N, \quad \forall j = 1, \dots, c, \end{aligned} \quad (9)$$

where  $\mathbf{U} = [u_{ji}]$  is the partition matrix,  $u_{ji} \in [0, 1]$  is the membership degree that the  $i$ th training pattern  $\mathbf{x}_i$  belongs to the  $j$ th cluster,  $\mathbf{V} = (\mathbf{v}_1, \dots, \mathbf{v}_j, \dots, \mathbf{v}_c)$  is a  $c$ -tuple of cluster prototypes,  $\mathbf{v}_j \in \mathbf{R}^d$  is the centroid of the  $j$ th cluster,  $c$  is the number of clusters ( $2 \leq c \leq N$ ), and  $m \in (1, \infty)$  is the weight controlling the degree of fuzziness of the matrix  $\mathbf{U}$  (we set  $m = 2$  in this study). FCM algorithm is performed in an iterative manner as follows.

*Step 1.* Initialize  $c$ .

*Step 2.* Initialize  $\mathbf{v}_j$ .

*Step 3.* Compute  $u_{ji}$  by

$$u_{ji} = \frac{\left(1/\|\mathbf{x}_i - \mathbf{v}_j\|^2\right)^{1/(m-1)}}{\sum_{k=1}^c \left(1/\|\mathbf{x}_i - \mathbf{v}_k\|^2\right)^{1/(m-1)}}. \quad (10)$$

*Step 4.* Update  $\mathbf{v}_j$  by

$$\mathbf{v}_j = \frac{\sum_{i=1}^N u_{ji}^m \mathbf{x}_i}{\sum_{i=1}^N u_{ji}^m}, \quad j = 1, \dots, c. \quad (11)$$

*Step 5.* Repeat Steps 3 and 4 until  $\|\mathbf{U}\| < \varepsilon$ , where  $\varepsilon$  is a small positive value.

Choosing a proper number  $c$  of clusters is of primary importance. The best number  $c_{\text{best}}$  of clusters can be determined using the partitioning entropy-based criterion [43] as follows:

$$c_{\text{best}} = \arg \min_{c=2}^N \left\{ \min_{\mathbf{U} \in \Omega_c} [H(\mathbf{U}, c)] \right\}, \quad (12)$$

where  $\Omega_c$  is the set of solutions and

$$H(\mathbf{U}, c) = \frac{1}{N} \sum_{i=1}^N \sum_{j=1}^c |u_{ji} \ln u_{ji}| \quad (13)$$

is the partitioning entropy, providing a global validity measure for the clustering results. Finally, the training set  $T$  is partitioned into  $c_{\text{best}}$  subsets using the simple rule: the  $i$ th training data point  $\mathbf{x}_i$  is assigned to the  $k$ th cluster if its membership degree to the  $k$ th cluster is the highest; namely,

$$k = \arg \max_{j=1}^{c_{\text{best}}} u_{ji}. \quad (14)$$

*Stage 2* (training SVDDs). After the training set  $T$  is partitioned into the  $c_{\text{best}}$  subsets  $T_i$ ,  $i = 1, \dots, c_{\text{best}}$ , where there is no overlap between two subsets and  $|T| = \sum_{i=1}^{c_{\text{best}}} |T_i|$ , the second training stage is to train  $c_{\text{best}}$  SVDDs. The  $i$ th subset is used to train the  $i$ th SVDD. Namely, use the face training subset  $T_i = \{\mathbf{x}_j \in \mathbf{R}^d\}_{j=1}^{|T_i|}$  to obtain the Lagrange multipliers for the  $i$ th SVDD by solving the quadratic programming problem formulated in (3) and then use (6) to calculate the sphere radius  $R_i$ . However, the trained decision boundary tightly encloses the face training subset. For some test face patterns located around the subset's distribution, they may be outside of the boundary. Such face test patterns will be rejected as nonfaces as a result. In this study, we deal with this problem by enlarging the sphere; that is,

$$R_i \leftarrow R_i + \Delta R_i, \quad (15)$$

where  $\Delta R_i$  is a positive value.  $\Delta R_i$  cannot be too large, otherwise the outlier acceptance rate (nonfaces classified as faces) will be significantly increased although the face

acceptance rate is improved. Therefore,  $\Delta R_i$  should be much smaller than  $R_i$ . According to our preliminary testing results, setting  $\Delta R_i = R_i/15$  can improve the face acceptance rate without increasing the nonface acceptance rate. After updating the sphere radius by (15), the decision function  $f_i(\mathbf{x})$  for the  $i$ th SVDD is obtained.

*3.2. Testing.* For an unseen pattern  $\mathbf{x}$ , it is rejected as a nonface if it is rejected by all the committee members, accepted as a face otherwise:

$$\begin{aligned} \mathbf{x} \in \text{nonface}, & \quad \text{if } f_i(\mathbf{x}) > 0, \quad \forall i = 1, \dots, c_{\text{best}} \\ \mathbf{x} \in \text{face}, & \quad \text{otherwise.} \end{aligned} \quad (16)$$

In other words, the decision making strategy used in the SVDD committee is not based on the usual majority voting. Instead, the unseen pattern is classified as a face as long as at least one of the SVDDs accepts it.

## 4. Results and Discussions

In Section 4.1, we first introduce the data set used for experiments. Then, in Section 4.2 we give an illustrated example that shows how the proposed SVDD committee deals with the problem of multicluster face distribution by a visualization analysis. Then, we compare our method with SVM and other one-class classifiers in terms of face detection accuracy.

*4.1. Data Set.* The extended MIT face data set [44, 45] consists of a training set and a test set. The training set contains 489410 patterns, where 17496 are faces and the remaining 471914 are nonfaces. The test set contains 472 faces and 23573 nonfaces. Each pattern is represented by a 361-dimensional vector.

*4.2. An Illustrative Example.* We randomly select 100 face patterns from the extended MIT face data set and perform the principal component analysis (PCA) to reduce the dimensionality of the patterns by selecting two leading eigenvectors of the pattern covariance matrix. The projections of the 100 face patterns in the 2D PCA-based subspace are depicted in Figure 2.

It can be observed from Figure 2(a) that the face patterns do not form a compact distribution but a multimodal (multicluster) distribution in the 2D space. We first use one single SVDD to learn the boundary which encloses most of the face patterns. As the kernel parameter  $\sigma$  is set to a very large value ( $\sigma = 100$ ), the decision boundary of SVDD is nearly spherical shaped in the original 2D space. As the value of  $\sigma$  decreases to 10 from 100, the SVDD boundary becomes tighter (see Figure 2(c)). However, the boundary is still not tight enough because there still exist empty areas (the areas within the green circles). Nonface patterns falling into these empty areas will be accepted as faces since the patterns are also inside of the SVDD boundary. One way to avoid such a situation is to further decrease the value of  $\sigma$ . However, according to author's previous work [46], when  $\sigma$  is too small, all the mapped target training data will be orthonormal to each other in the Gaussian kernel-induced feature space  $F$

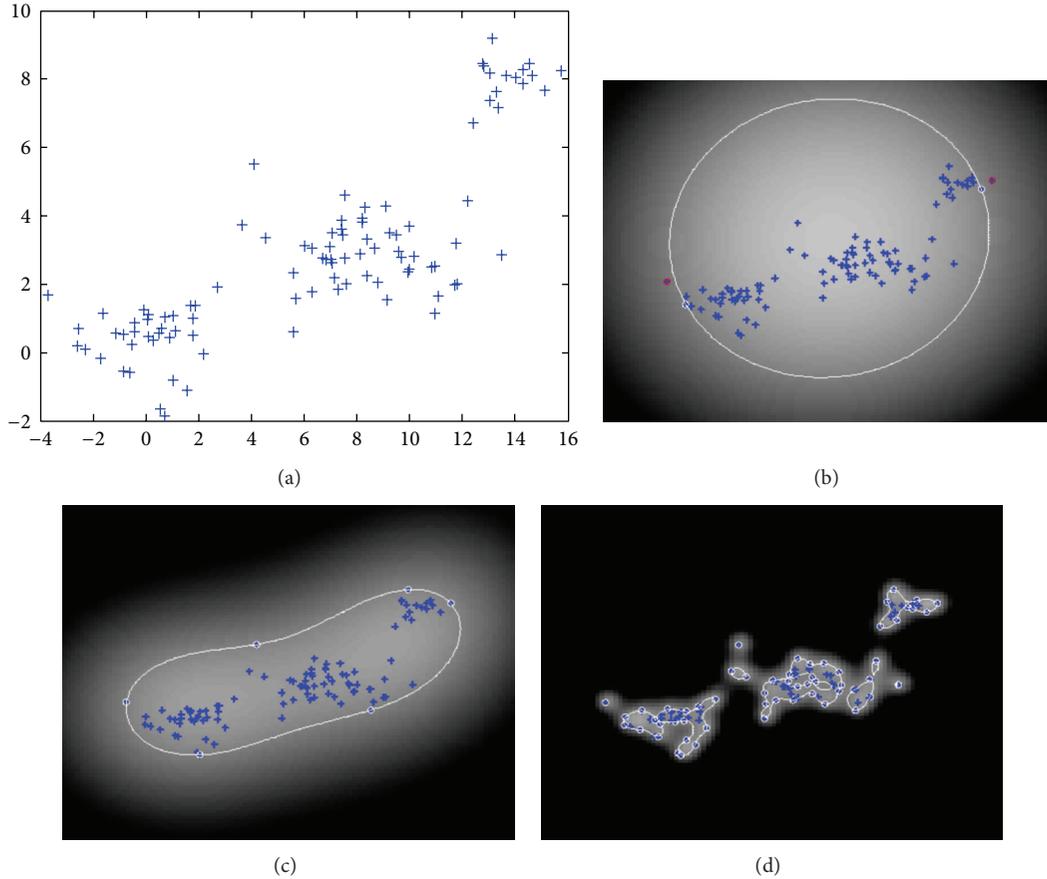


FIGURE 2: (a) Shows the distribution of the randomly selected face patterns in a 2D PCA-based subspace. (b), (c), and (d) show the SVDD training results using  $\sigma = 100$ ,  $\sigma = 10$ , and  $\sigma = 1$ , respectively. The white curves are the SVDD decision boundaries. The blue crosses with white circles denote the UBSVs, while the blue crosses with red circles in the upper-right subfigure are BSVs. The penalty weight  $C$  is set as 0.3 in all the experiments.

and become SVs. Also, when all or almost all the target training data becomes SVs, the boundary will be too tight to get a good target acceptance rate [33, 34]. As shown in Figure 2(d), the boundary overfits the face training set and most of the face training patterns become SVs as  $\sigma$  is set to 1. Although nonfaces can be rejected by this extremely tight boundary successfully, face test patterns are easily rejected as nonfaces as well, resulting in a poor face acceptance rate (i.e., face detection rate).

Instead of using the whole face training set to train an SVDD, the proposed SVDD committee partitions the whole training set into disjoint clusters and then utilizes the face patterns in each cluster to train an independent SVDD. By doing so, the performance drop due to the multicluster face distribution can thus be solved. An illustrated example is shown in Figure 3.

First, the 100 face patterns are partitioned into three clusters (three disjoint subsets) using the FCM algorithm and the best cluster selection criterion stated in Section 3. Compared with the whole training set, each subset forms a much more compact distribution. Then, each subset is utilized to train an SVDD. Finally, the three independent SVDDs constitute a committee. By comparing the results of

Figure 3 with the ones of Figure 2, we can see that using multiple SVDDs to find a description for a multicluster face distribution is more suitable than using one single SVDD.

### 4.3. Comparison with Other Methods

**4.3.1. Methods.** We compare our proposed SVDD committee with the frequently adopted face detector SVM and the one-class learning methods including the regular SVDD (i.e., single SVDD) and Kernel PCA [36].

Kernel PCA is a nonlinear version of PCA, which was originally designed for pattern representation [47]. Recently, Tsang et al. [44] further extended their original idea to novelty detection. Kernel PCA uses the reconstruction error in a kernel-induced feature space as a novelty measure (see Section 3 of [44] for details). A test data point is accepted as a target if its reconstruction error is below a predefined threshold, rejected as an outlier otherwise. The Kernel PCA for novelty detection involves three free parameters: kernel parameter, number of chosen eigenvectors  $q$ , and the threshold of the reconstruction error  $T_{re}$ . Free parameters of the methods are listed in Table 1.

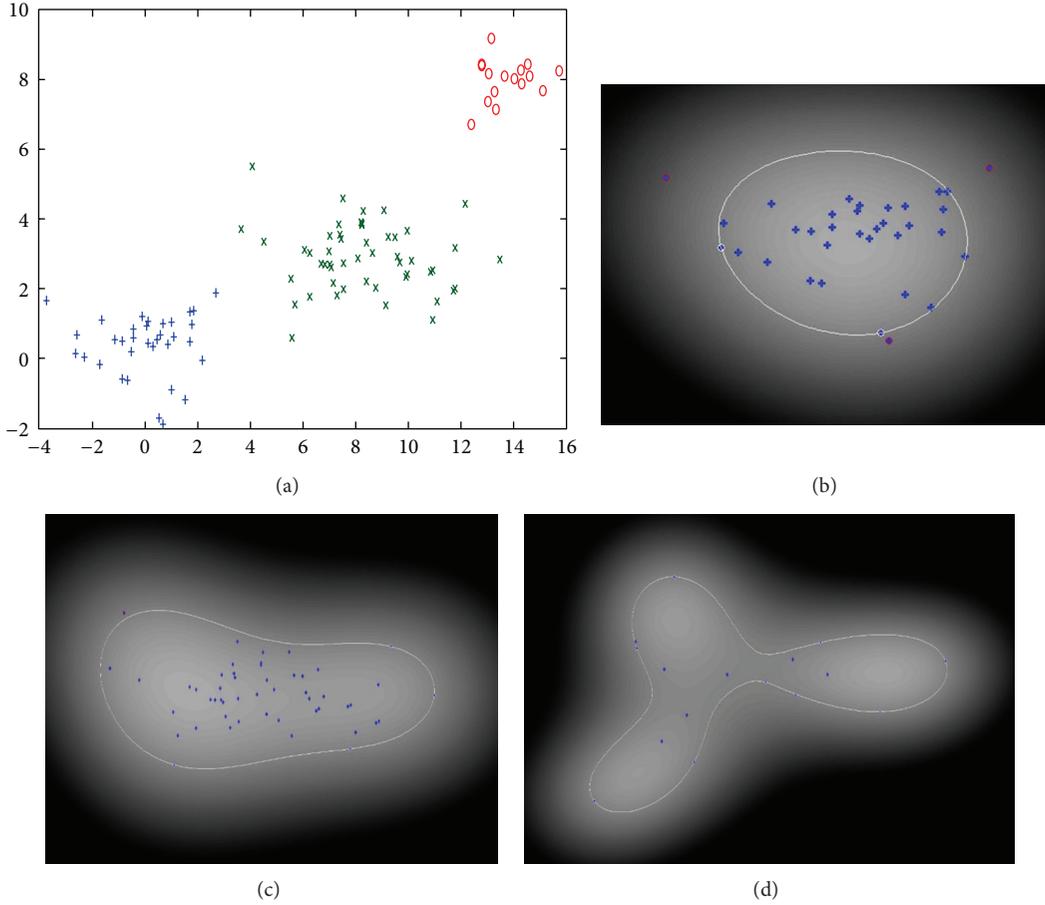


FIGURE 3: (a) shows that the 100 face patterns in the 2D PCA-based subspace are partitioned into three clusters. Face data sets in the three clusters are used to train three independent SVDDs, and the results are displayed in (b) (SVDD for cluster 1), (c) (SVDD for cluster 2), and (d) (SVDD for cluster 3) subfigures, respectively.

TABLE 1: Free parameters of the methods to be compared.

Methods	Parameters
SVM	$C, \sigma$
Kernel PCA	$\sigma, q, T_{re}$
SVDD	$C, \sigma$
SVDD committee	$C, \sigma$

4.3.2. *Performance Measure.* For SVM, faces are treated as positive (target) data while nonfaces as negative data (outlier). For one-class classifiers (SVDD, SVDD committee, and Kernel PCA) faces are treated as targets while nonfaces as outliers. For face detection problems, nonfaces often outnumber faces largely. If the data set is imbalanced, usual classification error rate is not an appropriate performance measure [48]. Therefore, the balanced loss

$$l_{\text{bal}} = 1 - \frac{\text{TAR} + \text{ORR}}{2} \quad (17)$$

suggested in [48] is adopted as the performance measure in the following experiments, where TAR and ORR stand for the target acceptance rate and outlier rejection rate, respectively.

A good face detector should be able to achieve a high TAR and a high ORR simultaneously. Therefore, the lower the balanced loss is, the better the face detector is. To facilitate the comparisons, the balanced loss is simply called error rate hereafter.

4.3.3. *Experiments and Results.* All methods are optimized by performing a fivefold cross validation procedure on a data set. The data set is not the entire training set of the extended MIT face data set because the scale of the training set is very large: the training set contains 17496 faces and 471914 nonfaces. Since the intent of this paper is not on proposing a method that can deal with large-scale data sets but on proposing a robust face detector which can be better than the widely used face detector SVM and further improving the performance of the one-class classifier SVDD. Thus, only a subset of the training set is used for training, and a subset of the test set is used for testing. The experiment is as follows.

(1) *Step 1.* Select  $n_f$  face and  $n_{\text{non}}$  nonface patterns from the training set by random. The collected  $n_f + n_{\text{non}}$  patterns form a new training set. In addition, randomly select 200 face and 400 nonface patterns from the test set. The 600 patterns form

TABLE 2: Comparison of average error rate among different methods on extended MIT face data set (in %).

	Exp A	Exp B	Exp C
SVM	20.12	18.24	14.24
Kernel PCA	16.80	16.03	13.45
SVDD	18.83	17.12	14.27
SVDD committee	14.24	14.02	12.23

a new test set. There is no overlap between the new training set and the new test set.

(2) *Step 2.* Perform 10-run twofold cross validation [30] on the new training set to optimize the methods.

(3) *Step 3.* Feed the new test set to the methods to obtain their error rates.

(4) *Step 4.* Repeat Step 1–Step 3 ten times. Compute the average error rate of each method.

We set  $n_f = 500$  and  $n_{\text{non}} = 500, 1000, \text{ and } 1500$ . There are three experiments in total: Exp A ( $n_f = 500, n_{\text{non}} = 1000$ ), Exp B ( $n_f = 500, n_{\text{non}} = 1000$ ), and Exp C ( $n_f = 500, n_{\text{non}} = 1500$ ). The average error rates of the methods are listed in Table 2.

According to Table 2, we can first observe that the average error rates of SVM in Exp A, Exp B, and Exp C are 20.12%, 18.24%, and 14.24%, respectively. The comparison results indicate that increasing the number of nonface training patterns can improve generalization performance of SVM in face detection. This could be due to the fact that when the number of nonface training patterns increases, the distribution of nonface training patterns can better represent the true distribution of the nonface class, thus being able to gain better generalization performance. Moreover, all the one-class classifiers perform better than the two-class classifier SVM in all experiments except for Exp C where SVM (14.24%) is slightly better than SVDD (14.27%). Also, the proposed SVDD committee gives the best results in the three experiments and greatly improves the accuracy of the original SVDD in face detection, which can be seen from Table 3 where the error reduction ratio (ERR) is defined as

$$\text{ERR} = \frac{e_{\text{SVDD}} - e_{\text{SVDD.Com}}}{e_{\text{SVDD}}}, \quad (18)$$

where  $e_{\text{SVDD}}$  and  $e_{\text{SVDD.Com}}$  denote the average error rates of SVDD and SVDD committee, respectively. The results reported in Table 3 demonstrate that the proposed SVDD committee performs better than one single SVDD in face detection.

**4.3.4. Training and Testing Speeds.** We further compare the computational complexities of the methods. The training time and testing time of each method are recorded during the experiments. A 3.40 GHz-CPU (i7-3770) computer (with 8 GB RAM) running on Windows 8 is used. The methods are implemented with MATLAB 7.10, which is also a 64-bit

TABLE 3: Error reduction ratios (ERRs) of SVDD committee to SVDD in the three experiments (in %).

	Exp A	Exp B	Exp C
ERR	24.38	18.11	14.30

TABLE 4: Training time of each method (in s).

	Exp A	Exp B	Exp C
SVM	14.56	39.80	120.33
Kernel PCA	2.18	2.11	2.31
SVDD	1.67	1.58	1.66
SVDD committee	3.07	3.23	3.12

TABLE 5: Testing speed of each method (ms/pattern).

	Exp A	Exp B	Exp C
SVM	1.32	2.43	2.69
Kernel PCA	1.31	1.01	1.03
SVDD	0.64	0.45	0.56
SVDD committee	1.31	1.12	1.54

version. Finally, training time and testing are summarized in Tables 4 and 5, respectively.

We can see from Table 4 that the training time of SVM increases with the size of the training set, because SVM training has  $O(n^3)$  time complexity, where  $n = n_f/2 + n_{\text{nf}}/2$  (for twofold cross validation, 50% of the patterns are used for training). For one-class classifiers, only half of the face training patterns are included in the training. For example, SVDD's training time complexity is  $O(n^3)$ , where  $n = n_f/2$ . Since  $n_f = 500$  in all the three experiments, the actual training time of SVDD is almost the same for the three cases. SVDD committee takes more training time than SVDD. However, compared with SVM, SVDD committee has a much faster training speed, especially when the number of training faces increases. The testing speeds reported in Table 5 indicate that SVDD has the fastest testing speed. Although SVDD committee is lower than SVDD, its testing speed (1.12–1.54 ms/pattern) is acceptable for real-time face detection.

## 5. Conclusion

In this paper, we have presented a novel face detector called SVDD committee. The proposed SVDD committee is based on one-class learning and partitioning strategies, thus being able to improve the generalization performance of the original SVDD in face detection. Also, nonface patterns are not required in the training of SVDD committee. Therefore, the face detection accuracy of SVDD committee would not be affected by the chosen nonface patterns. On the contrary, the frequently adopted face detector SVM is a two-class classifier. Its face detection accuracy depends on the number  $n_f$  of collected nonface training patterns, and its training time increases with  $n_f$ . Experiments have demonstrated that the proposed SVDD committee not only performs better

than SVM, but also significantly improves the generalization performance of the original SVDD in face detection. Also, the testing speed is acceptable for applications where real-time face detection is required.

This work does not apply any robust feature extraction methods, because the focus of this work is on the development of a novel face detector based on one-class learning. It is believed that the SVDD committee-based face detection accuracy can be further improved if advanced feature extractions are applied, such as LBP and Haar-like features. In addition, the presented work does not address other critical issues such as learning from large-scale data sets. Those will be our future works.

### Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

### Acknowledgment

This study was supported by a grant from the Institute of Nuclear Energy Research, Atomic Energy Council, Taiwan.

### References

- [1] E. Hjelmås and B. K. Low, "Face detection: a survey," *Computer Vision and Image Understanding*, vol. 83, no. 3, pp. 236–274, 2001.
- [2] M.-H. Yang, D. J. Kriegman, and N. Ahuja, "Detecting faces in images: a survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 1, pp. 34–58, 2002.
- [3] C. Zhang and Z. Zhang, "A survey of recent advances in face detection," Tech. Rep. MSR-TR-2010-66, Microsoft Corporation, 2010.
- [4] P. Viola, M. J. Jones, and D. Snow, "Detecting pedestrians using patterns of motion and appearance," *International Journal of Computer Vision*, vol. 63, no. 2, pp. 153–161, 2005.
- [5] G. Gualdi, A. Prati, and R. Cucchiara, "Multistage particle windows for fast and accurate object detection," *The IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, pp. 1589–1604, 2012.
- [6] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'01)*, pp. 1511–1518, December 2001.
- [7] C. Liu, "A Bayesian discriminating features method for face detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 6, pp. 725–740, 2003.
- [8] C. A. Waring and X. Liu, "Face detection using spectral histograms and SVMs," *IEEE Transactions on Systems, Man, and Cybernetics B*, vol. 35, no. 3, pp. 467–476, 2005.
- [9] H. Zhang, W. Gao, X. Chen, and D. Zhao, "Object detection using spatial histogram features," *Image and Vision Computing*, vol. 24, no. 4, pp. 327–341, 2006.
- [10] Y. H. Liu and Y. T. Chen, "Face detection using kernel PCA and imbalanced SVM," in *Advances in Natural Computation*, vol. 4221 of *Lecture Notes in Computer Science*, pp. 351–360, Springer, Berlin, Germany, 2006.
- [11] H. A. Rowley, S. Baluja, and T. Kanade, "Neural network-based face detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 1, pp. 23–38, 1998.
- [12] R. Féraud, O. J. Bernier, J.-E. Viallet, and M. Collobert, "A fast and accurate face detector based on neural networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 1, pp. 42–53, 2001.
- [13] C. Garcia and M. Delakis, "Convolutional face finder: a neural architecture for fast and robust face detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 11, pp. 1408–1423, 2004.
- [14] H. Schneiderman and T. Kanade, "Statistical method for 3D object detection applied to faces and cars," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '00)*, pp. 746–751, June 2000.
- [15] E. Osuna, R. Freund, and F. Girosi, "Training support vector machines: an application to face detection," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'97)*, pp. 130–136, June 1997.
- [16] T. J. Terrillon, M. N. Shirazi, M. Sadek, H. Fukamachi, and S. Akamatsu, "Invariant face detection with support vector machines," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'00)*, vol. 4, pp. 210–217, 2000.
- [17] S. Romdhani, P. Torr, B. Schölkopf, and A. Blake, "Computationally efficient face detection," in *Proceedings of the 8th IEEE International Conference on Computer Vision (ICCV'01)*, pp. 695–700, Vancouver, Canada, 2001.
- [18] I. Buciu, C. Kotropoulos, and I. Pitas, "Combining support vector machines for accurate face detection," in *IEEE International Conference on Image Processing (ICIP'01)*, pp. 1054–1057, Thessaloniki, Greece, October 2001.
- [19] B. Heisele, T. Serre, S. Prentice, and T. Poggio, "Hierarchical classification and feature reduction for fast face detection with support vector machines," *Pattern Recognition*, vol. 36, no. 9, pp. 2007–2017, 2003.
- [20] M. Rätsch, S. Romdhani, and T. Vetter, "Efficient face detection by a cascaded support vector machine using haar-like features," in *Pattern Recognition*, vol. 3175 of *Lecture Notes in Computer Science*, pp. 62–70, Springer, Berlin, Germany.
- [21] P. Wang and Q. Ji, "Multi-view face detection under complex scene based on combined SVMs," in *Proceedings of the 17th International Conference on Pattern Recognition (ICPR'04)*, pp. 179–182, August 2004.
- [22] J. Chen, R. Wang, S. Yan, S. Shan, X. Chen, and W. Gao, "Enhancing human face detection by resampling examples through manifolds," *IEEE Transactions on Systems, Man, and Cybernetics A*, vol. 37, no. 6, pp. 1017–1028, 2007.
- [23] G.-Y. Zhang, L.-Y. Guo, J.-H. Wu, H.-M. Li, and G.-Q. Guo, "Face detection using Gabor wavelet and SVM," in *Proceedings of the International Conference on Computer Application and System Modeling (ICCSM'10)*, vol. 2, pp. 588–590, October 2010.
- [24] C. Kyrkou and T. Theocharides, "SCoPE: towards a systolic array for SVM object detection," *IEEE Embedded Systems Letters*, vol. 1, no. 2, pp. 46–49, 2009.
- [25] M. Kawulok and J. Szymanek, "Precise multi-level face detector for advanced analysis of facial images," *IET Image Processing*, vol. 6, no. 2, pp. 95–103, 2012.
- [26] P. Viola and M. J. Jones, "Robust real-time face detection," *International Journal of Computer Vision*, vol. 57, no. 2, pp. 137–154, 2004.

- [27] V. N. Vapnik, *Statistical Learning Theory*, Adaptive and Learning Systems for Signal Processing, Communications and Control Series, John Wiley & Sons, New York, NY, USA, 1998.
- [28] M. Markou and S. Singh, "Novelty detection: a review, part I: statistical approachess," *Signal Processing*, vol. 83, no. 12, pp. 2481–2497, 2003.
- [29] M. Markou and S. Singh, "Novelty detection: a review, part II: neural network based approaches," *Signal Processing*, vol. 83, no. 12, pp. 2499–2521, 2003.
- [30] K. Y. Lee, D.-W. Kim, K. H. Lee, and D. Lee, "Density-induced support vector data description," *IEEE Transactions on Neural Networks*, vol. 18, no. 1, pp. 284–289, 2007.
- [31] C. Campbell and K. P. Bennett, "A linear programming approach to novelty detection," in *Advances in Neural Information Processing Systems*, vol. 13, pp. 395–401, MIT Press, Cambridge, Mass, USA, 2001.
- [32] G. R. G. Lanckriet, L. E. Ghaoui, and M. I. Jordan, "Robust novelty detection with single-class MPM," in *Advances in Neural Information Processing Systems*, vol. 15, pp. 929–936, MIT Press, Cambridge, Mass, USA, 2003.
- [33] B. Schölkopf, J. C. Platt, J. Shawe-Taylor, A. J. Smola, and R. C. Williamson, "Estimating the support of a high-dimensional distribution," *Neural Computation*, vol. 13, no. 7, pp. 1443–1471, 2001.
- [34] D. M. J. Tax and R. P. W. Duin, "Support vector domain description," *Pattern Recognition Letters*, vol. 20, no. 11–13, pp. 1191–1199, 1999.
- [35] D. M. J. Tax and R. P. W. Duin, "Support vector data description," *Machine Learning*, vol. 54, no. 1, pp. 45–66, 2004.
- [36] H. Hoffmann, "Kernel PCA for novelty detection," *Pattern Recognition*, vol. 40, no. 3, pp. 863–874, 2007.
- [37] J. Lu, K. N. Plataniotis, and A. N. Venetsanopoulos, "Face recognition using kernel direct discriminant analysis algorithms," *IEEE Transactions on Neural Networks*, vol. 14, no. 1, pp. 117–126, 2003.
- [38] Y.-H. Liu and Y.-T. Chen, "Face recognition using total margin-based adaptive fuzzy support vector machines," *IEEE Transactions on Neural Networks*, vol. 18, no. 1, pp. 178–192, 2007.
- [39] S. Khazai, A. Safari, B. Mojaradi, and S. Homayouni, "Improving the SVDD approach to hyperspectral image classification," *IEEE Geoscience and Remote Sensing Letters*, vol. 9, no. 4, pp. 594–598, 2012.
- [40] Y.-H. Liu, Y.-K. Huang, and M.-J. Lee, "Automatic inline defect detection for a thin film transistor-liquid crystal display array process using locally linear embedding and support vector data description," *Measurement Science and Technology*, vol. 19, no. 9, Article ID 095501, 2008.
- [41] X. Wang, X. L. Wang, and D. M. Wilkes, "An automated vision based on-line novel percept detection method for a mobil robot," *Robotics and Autonomous Systems*, vol. 60, no. 10, pp. 1279–1294, 2012.
- [42] J. C. Bezdek, *Pattern Recognition with Fuzzy Objective Function Algorithms*, Plenum Press, New York, NY, USA, 1981.
- [43] C. T. Lin and C. S. G. Lee, *Neural Fuzzy Systems: A Neuro-Fuzzy Synergism to Intelligent System*, Prentice-Hall,, Englewood Cliffs, NJ, USA, 1996.
- [44] I. W.-H. Tsang, J. T.-Y. Kwok, and J. M. Zurada, "Generalized core vector machines," *IEEE Transactions on Neural Networks*, vol. 17, no. 5, pp. 1126–1140, 2006.
- [45] I. W. Tsang, A. Kocsor, and J. T. Kwok, "LibCVM," 2011, <http://www.c2i.ntu.edu.sg/ivor/cvm.html>.
- [46] Y.-H. Liu, Y.-C. Liu, and Y.-J. Chen, "Fast support vector data descriptions for novelty detection," *IEEE Transactions on Neural Networks*, vol. 21, no. 8, pp. 1296–1313, 2010.
- [47] 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.
- [48] J. Weston, B. Schölkopf, E. Eskin, C. Leslie, and S. Noble, "Dealing with large diagonals in kernel matrices," in *Principles of Data Mining and Knowledge Discovery*, vol. 243 of *Lecture Notes in Computer Science*, pp. 494–511, Springer, Helsinki, Finland, 2002.



# Hindawi

Submit your manuscripts at  
<http://www.hindawi.com>

