Probabilistic Weighted Induced Multi-Class Support Vector Machines for Face Recognition

Aniruddha Dey
Department of Information Technology, MAKAUT, Salt Lake, Kolkata, India
E-mail: anidey007@gmail.com

Shiladitya Chowdhury
Department of Master of Computer Application, Techno India, Kolkata, India
E-mail: dityashila@yahoo.com

Abstract: This paper deals with a probabilistic weighted multi-class support vector machines (WMSVM) for face recognition. The support vector machines (SVM) has been applied to many application fields such as pattern recognition in last decade. The support vector machines determine the hyperplane which separates largest fraction of samples of the similar class on the same side. The SVM also maximizes the distance from the either class to the separating hyperplane. It has been observed that in many realistic applications, the achieved training data is frequently tainted by outliers and noises. Support vector machines are very sensitive to outliers and noises. It may happen that a number of points in the training dataset are misplaced from their true position or even on the wrong side of the feature space. The weighted support vector machines are designed to overcome the outlier sensitivity problem of the support vector machines. The main issue in the training of the weighted support vector machines algorithm is to build up a consistent weighting model which can imitate true noise distribution in the training dataset, i.e., reliable data points should have higher weights, and the outliers should have lower weights. Therefore, the weighted support vector machines are trained depending on the weights of the data points in the training set. In the proposed method the weights are generated by probabilistic method. The weighted multi-class support vector machines have been constructed using a combination of the weighted binary support vector machines and one-against-all decision strategies. Numerous experiments have been performed on the AR, CMU PIE and FERET face databases using different experimental strategies. The experimental results show that the performance of the proposed method is superior to the multi-class support vector machines in terms of recognition rate.

Povzetek: Opisana je metoda podpornih vektorjev za prepoznavanje obrazov.

1 Introduction

The SVM can be considered as an estimated implementation of the structural risk minimization method [1]. In 1998, Vapnik first devised the SVM to address the pattern classification and recognition problem [2]. The objective of the support vector machines is to determine the hyperplane that divides largest fraction of images in the related class on the same adjacent, whereas maximizing the space from the both class to the separating hyperplane. This separating hyperplane is known as optimal separating hyperplane (OSH). The OSH minimizes the misclassification risk. It may be noted that in many realistic applications, some training data points are placed far away from the accurate position or even on the wrong side of the feature space. These data points are called outliers. In general, the training dataset is severely affected by the outliers and different kind of noises. The SVMs are actual sensitive to outliers and different kind of noises. Therefore, in the training phase, the outliers with large Lagrangian coefficient can become a support vector [3]. In the past few decades, wide ranges of techniques have been introduced by several researchers to solve the aforementioned bottleneck of the SVM. Zhang [4] proposed central SVM (CSVM) in which class centres are used to build the support vector machines. For each training data point, the adaptive margin SVM (AMSVM) training algorithm [5] depends on the utilization of the adaptive margins. Song et al. [6], [7] proposed a robust SVM (RSVM) in which to generate an adaptive margin, the space between centre of every class of the training sample and the data point is computed. But this method has a drawback because it is very difficult to tune the penalty parameter. The method uses the averaging method which is partly sensitive for outliers and noises. Authors in [8] and [9] proposed fuzzy SVM (FSVM) to eliminate the outlier sensitivity problem. To moderate the effect of outliers, the method applies the fuzzy membership's values to the training data. Membership function selection is main drawback for the FSVM. Cao et al. [10] proposed the support vector novelty detector

(SVND) which detects the outliers more appropriately from the normal data points, and solve one-class classification problem.

Some new improvements on the support vector machines can be establish in the literature review. Quan *et al.* [11] established the weighted least squares support vector machine (WLS-SVM) local region algorithm. This algorithm calculates the nonlinear time series, as well as performs robust estimation for regression using the limited observations. In this method, there is a simple and effectual technique to model parameter selection based on the leave one-out cross-validation strategy. A weighting method on Lagrangian SVM (LSVM) is proposed by Hwang *et al.* [12]. This method deals with the imbalanced data classification problem. In this method, a weight parameter is added to the LSVM design. Therefore, the method can get better performance for the minority class with minimum control on classification performance of the majority class. Yu [13] proposed the asymmetric weighted least squares support vector machine (LSSVM) combined learning procedure. This methodology is based on the evolutionary programming (EP), and is used for software repository mining. A nonparallel plane classifier, namely, weighted twin support vector machines with local information (WLTSVM) is proposed by Ye *et al.* [14]. This method mines underlying similarity information within the samples as much as possible. Shao *et al.* [15] proposed the weighted Lagrangian twin support vector machines (WLTSVM) for the imbalanced data classification. Xanthopoulos *et al.* [16] suggested the weighted support vector machines for automated procedure checking and early error diagnosis. The robust LS-SVM (RLS-SVM) is proposed by Yang *et al.* [17], and the method is established on the truncated least squares loss function for classification and regression with noises. Zhang *et al.* [26] proposed an emotion recognition system based on facial expression images. In this work**,** the bi-orthogonal wavelet entropy is used to extract multi-scale features and the fuzzy multi-class support vector machine is used as classifier. More recently, Wang *et al.* offered a new intelligent emotion recognition system where stationary wavelet entropy are used to extract feature values and a single hidden layer feed forward neural network is employed as the classifier [27]. Aburomman and Reaz proposed ensemble classifiers are generated using the novel methods as well as the weighted majority algorithm (WMA) technique [28]. Some learning based discriminant analysis techniques have been suggested, such as local structure preserving discriminant analysis [29], Discriminant similarity and variance preserving projection [30] to abuse the label info contained in the data. Shi*et al.* established 3D face recognition method based on LBP and SVM.Hu and Cui proposed Digital image recognition based on Fractional order PCA-SVM coupling algorithm [32]. By improve the SVM gender classification accuracy using clustering and incremental learning suggested by Dagher and Azar [33].Kar*et al.* face expression recognition system based on ripplet transform type II and least square SVM [34].

In this study, the probabilistic weighted multi-class support vector machine is devised to address the outlier sensitivity problem. The main issue in the training samples of the weighted support vector machines algorithm is to improve a reliable weighting model which can reflect true noise distribution in the training data, i.e., reliable data points should have higher weights, and the outliers should have lower weights. Therefore, dissimilar weights are allocated to different data points. Therefore, as per relative importance of the data points in the training set, the training algorithm of the weighted SVM determines the decision surface. The probabilistic method is used to generate the weights of the proposed probabilistic weighted multi-class support vector machines training algorithm. These weights are incorporated with all data points of the training set. The weighted support vector machines training algorithm maximizes the margin of separation with the help of weights to prevent some points. In this work, the generalized two-dimensional Fisher's linear discriminant (G-2DFLD) technique is applied for feature extraction [18]. The extracted features are applied on the proposed probabilistic weighted multi-class support vector machines for training, classification and recognition. The empirical results on the AR, CMU PIE and FERET face database illustrate that the proposed probabilistic weighted multi-class support vector machines (WMSVM) perform better than the multi-class SVM, in terms of face recognition.

Rest of the paper is ordered as follows. The basic idea of the SVM is given in Section 2. The proposed weight generating scheme, based on the probabilistic method, is discussed in Section 3. Section 4 describes the weighted support vector machines. The weighted multi-class support vector machines are defined in Section 5. The simulation results on the AR, CMU PIE, and FERET face databases are described in Section 6. Section 7 contains the concluding remarks.

2 Revisited support vector machines

The support vector machines were developed for binary pattern classification problem [1 -3]. It has been seen that in case of pattern classification problem, the SVMs provide satisfactory performance. The basic idea of the binary-class SVMs [1- 3] is to split two classes by a hyperplane. This separating hyperplane is created from the available training samples. The support vector machines find the hyperplane that splits largest fraction of samples of the alike class on the similar side, while maximizing the space from the each class to the separating hyperplane. This separating hyperplane is known as optimal separating hyperplane (OSH). The OSH reduces the misclassification risk.

3 Weight generation by the probabilistic method

Although, the support vector machines are very powerful for solving classification problem, however, it has some limitations as it treats all the training data points of a

given class uniformly. It has been seen that, all the data points of the training set are not equally important for classification and recognition purpose in many real world application domains. This limitation of the support vector machines can be overcome by designing the weighted support vector machines. In the weighted support vector machines, each and every data points are treated separately according to their weights.

The main issue of the training algorithm of the weighted support vector machines is to develop a reliable weighting model which can reflect actual distribution in the training set. The reliable data points should have higher weights, and the outliers should have lower weights. Therefore, dissimilar weights are assigned to different data points. The decision surface generated by the weighted SVM training algorithm considers the relative significance of data points in the training set. The weights employed in the proposed probabilistic weighted multi-class support vector machines are generated by the probabilistic method.

Let the cth class has N_c numbers of training samples. We consider the positive samples are belonging to class y_1 and negative samples are belonging to class y_2 to design the weighted SVM for c^{th} class.

Let $P(y_j)$; $j \in 1,2$, defined the prior probability of the sample which is included in y_j class. The prior probability of the sample belonging to y_1 class can be described as follows:

$$P(y_1) = \frac{N_c}{N} \tag{1}$$

Similarly, the prior probability of the sample belonging to y_2 class can be demonstrated as follows:

$$P(y_2) = \frac{N - N_c}{N} \tag{2}$$

Now for a positive training sample x_i the weight a_i is illustrated as follows:

$$a_i = P(y_1|x_i) = \frac{P(x_i|y_1)\,P(y_1)}{P(x_i|y_1)\,P(y_1) + P(x_i|y_2)\,P(y_2)} \tag{3}$$

Similarly, in case of a negative training sample x_i the weight a_i is generated as follows:

$$a_i = P(y_2|x_i) = \frac{P(x_i|y_2)\,P(y_2)}{P(x_i|y_1)\,P(y_1) + P(x_i|y_2)\,P(y_2)} \tag{4}$$

It is to be noted that $\varepsilon < a_i < 1$, and $\varepsilon (\varepsilon > 0)$ is sufficiently small. The term $P(y_j \mid x_i)$; $j \in 1,2$ is called posterior probability, i.e., probability of the class is y_j after we have performed measurement on the data x_i. Similarly, the term $P(x_i \mid y_j)$; $j \in 1,2$ is called conditional probability i.e., the probability that the class y_j has the feature value x_i. The equations (3) and (4) ensure that the lower weights are assigned to outliers or close to outliers.

Every measurement must be assigned to one of these two classes y_1 or y_2. Therefore,

$$\sum_{j=1}^{2} P(y_j \mid x_i) = 1 \tag{5}$$

The posterior probability of the sample x_i $(P(y_j \mid x_i); j \in 1,2)$ is used as weight for designing the proposed probabilistic weighted multi-class support vector machines.

4 Weighted support vector machines

It has been seen that the training dataset is often tainted by outliers and noises in many real world applications. The support vector machines are very sensitive to outliers and noises. It may so happen that some patterns in the training set are outliers and misplaced far away from the true position or even on the wrong side of the feature space. During the training process, the outlier with large Lagrangian coefficient can become a support vector. The optimal hyperplane obtained by the support vector machines depends only on small part of the data points, i.e., support vectors. So, in presence of outliers, the decision boundary obtained by the support vector machines training algorithm deviate severely from the optimal separating hyperplane.

The weighted support vector machines are designed to address this issue. In weighted support vector machines, the data points of the training set are treated differently according to their weights. The training algorithm gives more effort to correctly classify more important data points (i.e., the data points with larger weights) while caring less effort to less important data points (i.e., the data points with lower weights, probably outliers).

Let B be a set of labeled training samples associated with weights:

$$B = \{(x_i, y_i, a_i)\}_{i=1}^{N}; \quad x_i \in \Re^d; \quad y_i \in \{+1, -1\} \tag{6}$$

where, x_i is the input pattern for the ith training sample, a_i is the weight assigned to x_i, and y_i is the class of the x_i. In the proposed probabilistic weighted multi-class support vector machines, the weight is generated by the weight generating technique described in section 3.

To achieve better performance, the weighted support vector machines training algorithm maximizes the margin of separation. The optimal separating hyperplane in the case of weighted support vector machines minimizes the following function:

$$\Gamma(\omega, \xi, a) = \frac{1}{2} \omega^T \omega + C \sum_{i=1}^{N} a_i \xi_i \tag{7}$$

with constraints defined [1, 2].

In the optimization problem, the effect of the parameter ξ_i is reduced by the small value of a_i. Therefore, the training algorithm of the weighted SVM considers the corresponding point (x_i, y_i) as less significant for classification.

The solution to the optimization problem (7), subject to the constraints defined in [1, 2], is given by the saddle point of the following Lagrange function:

$$L(\omega, b, \xi_i, \lambda) = \frac{1}{2} \omega^T \omega + C \sum_{i=1}^{N} a_i \xi_i - $$
$$\sum_{i=1}^{N} \lambda_i (y_i((\omega^T \cdot x_i) + b) - 1 + \xi_i) - \sum_{i=1}^{N} \gamma_i \xi_i \tag{8}$$

By expanding equation (8) term by term, the following equation is obtained.

$$L(\boldsymbol{\omega},b,\xi_i,\lambda) = \frac{1}{2}\boldsymbol{\omega}^T\boldsymbol{\omega} + C\sum_{i=1}^{N}a_i\xi_i - \sum_{i=1}^{N}\lambda_i y_i(\boldsymbol{\omega}^T \cdot \boldsymbol{x}_i) -$$

$$b\sum_{i=1}^{N}\lambda_i y_i + \sum_{i=1}^{N}\lambda_i - \sum_{i=1}^{N}\lambda_i\xi_i - \sum_{i=1}^{N}\gamma_i\xi_i \qquad (9)$$

The Lagrange multipliers γ_i are presented in equations (8) and (9) to ensure the non-negativity of slack variables ξ_i. At saddle point, the Lagrange function (8) has to be minimized with respect to $\boldsymbol{\omega}$, b, and ξ_i. It has to be also maximized with respect to λ_i where, $0 \le \lambda_i \le a_i C$.

We can convert the Lagrange function (8) into its corresponding dual problem as follows:

$$\max_\lambda \boldsymbol{\omega}(\lambda) = \max_\lambda\{\min_{\omega,b,\xi_i}L(\boldsymbol{\omega},b,\xi_i,\lambda)\} \qquad (10)$$

Three optimal conditions can be derived from equation (9) as follows:

$$\frac{\delta}{\delta\boldsymbol{\omega}}L(\boldsymbol{\omega},b,\xi_i,\lambda) = \boldsymbol{\omega} - \sum_{i=1}^{N}\lambda_i y_i \boldsymbol{x}_i = 0 \qquad (11)$$

$$\frac{\delta}{\delta b}L(\boldsymbol{\omega},b,\xi_i,\lambda) = \sum_{i=1}^{N}\lambda_i y_i = 0 \qquad (12)$$

and

$$\frac{\delta}{\delta\xi_i}L(\boldsymbol{\omega},b,\xi_i,\lambda) = Ca_i - \lambda_i - \gamma_i = 0 \qquad (13)$$

The dual objective function can be obtained by substituting equations (11), (12) and (13) into the right side of the Lagrange function (9). Therefore, the dual problem for the weighted SVM can be formulated as follows:
Maximized:

$$R(\lambda) = \sum_{i=1}^{N}\lambda_i - \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\lambda_i\lambda_j y_i y_j(\boldsymbol{x}_i \cdot \boldsymbol{x}_j) \qquad (14)$$

with constraints defined in SVM and

$$0 \le \lambda_i \le a_i C \; ; \qquad i=1,2,..,N \qquad (15)$$

It can be seen that by setting $a_i = 1$ for all i, the weighted support vector machines will be similar to the support vector machines. There is only one free parameter (i.e., C) in support vector machines; whereas, in addition to C, the number of free parameters in weighted support vector machines is equal to the number of training samples.

It has been observed that the face individuals are highly non-linear because of the variations in facial expression, illumination condition, pose, etc. So, it is necessary to non-linearly map each sample into a high-dimensional feature space using a non-liner function $\varphi: \Re^d \to \Re^D; D \gg d$, and then the linear support vector machines can be implemented in high dimensional feature space. A positive definite kernel function K is selected *a priori* to perform inner product of vectors in the feature space to avoid explicit mapping φ and computational burden in the high-dimensional feature space. The kernel function can be defined as follows:

$$K(\boldsymbol{x}_i,\boldsymbol{x}_j) = \varphi(\boldsymbol{x}_i) \cdot \varphi(\boldsymbol{x}_j) \qquad (16)$$

where, $\varphi(\boldsymbol{x}_i)$ is the transformed vector of the person $\boldsymbol{x}_i$ by the non-linear function φ.

The polynomial and Gaussian radial basis function kernels are two well-known kernel functions:
Polynomial kernel:

$$K(\boldsymbol{x}_i,\boldsymbol{x}_j) = (\boldsymbol{x}_i \cdot \boldsymbol{x}_j)^r \qquad (17)$$

Gaussian radial basis function:

$$K(\boldsymbol{x}_i,\boldsymbol{x}_j) = \exp\left(-\frac{\|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2}{2\sigma^2}\right) \qquad (18)$$

where, r is a positive integer and $\sigma > 0$.

In the proposed probabilistic weighted multi-class support vector machines, we used the Gaussian radial basis function as kernel function. Therefore, the dual objective function (14) can be rewritten as follows:
Maximized:

$$F(\lambda) = \sum_{i=1}^{N}\lambda_i - \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\lambda_i\lambda_j y_i y_j K(\boldsymbol{x}_i,\boldsymbol{x}_j) \qquad (19)$$

with constraints defined in equations (15).

It can be observed that the objective function to be maximized for the dual problem of the support vector machines and weighted support vector machines is the same. The support vector machines are differing from the weighted support vector machines in that the constraint $0 \le \lambda_i \le C$ is replaced with more stringent constraint $0 \le \lambda_i \le a_i C$. The constraint optimization for the weighted support vector machines, and computations of the optimum values of the weight vector $\mathbf{w_o}$ and bias b_o proceed in the same way as in the case of the support vector machines.

Solving equation (19) with constraints defined in equations (15) determines the optimum Lagrange multipliers $\lambda_{o,i}$. Putting the values of optimum Lagrange multipliers $\lambda_{o,i}$ in equation, the optimum weight vector $\boldsymbol{\omega}_o$ can be obtained.

The Karush-Kuhn-Tucker (KKT) conditions in case of weighted support vector machines can be defined

$$\gamma_i\xi_i = 0; \quad i = 1,2,...,N \qquad (20)$$

By combining equations (13) and (20) the following equation can be formed:

$$(a_i C - \lambda_i)\xi_i = 0; \quad i = 1,2,...,N \qquad (21)$$

From SVM, it can be observed that

$$\xi_i = 0 \; ; \text{If } \lambda_i < a_i C \qquad (22)$$

The optimum bias b_o is determined by taking any data point $(\boldsymbol{x}_i, y_i)$ in the training set for which $0 < \lambda_{o,i} < a_i C$, and therefore $\xi_i = 0$, and using that data point.

In the proposed probabilistic weighted multi-class support vector machines, we solved the dual objective function using the sequential minimal optimization (SMO) algorithm [20].

5 Weighted multi-class support vector machines

The weighted multi-class support vector machines are constructed using a combination of the weighted support vector machines and the decision strategy to decide the class of the input pattern. Each weighted SVM is separately trained. The weighted multi-class support vector machines can be implemented using the one-against-all [1] and one-against-one [21] decision strategies. The one-against-all decision strategy is adopted in the proposed probabilistic weighted multi-class SVM to classify samples, as it requires less amount of memory. This decision strategy is stated as follows:

Let the training set $T = \{\boldsymbol{x}_i, c_j, a_i\}; i \in \{1,2,...N\}; j \in \{1,2,...,M\}$ be the collection of the training sample, its class, and weight, respectively. We designed the weighted SVM for each class by discriminating that class from the rest of $(M\text{-}1)$ classes. Therefore, in this methodology, we have used M number of weighted support vector machines. The set of training samples and their required outputs $(\boldsymbol{x}_i, y_i)$ are used to design the weighted SVM for class l. For a training sample $\boldsymbol{x}_i$, the required output y_i is formulated as follows:

$$y_i = \begin{cases} +1 & if \quad c_j = l \\ -1 & if \quad c_j \neq l \end{cases} \qquad (23)$$

The desired output of the *positive* and *negative* samples are $y_i = +1$ and $y_i = -1$, respectively.

The classifier recognizes a test sample by using the *winner-takes-all* decision strategy. Let the test sample $\boldsymbol{x}$ is recognized as class c. The output of the classifier is defined as follows:

$$c = \max_{\text{arg}} \{f_l(\boldsymbol{x})\}; \qquad l = 1,2,...,M \qquad (24)$$

where, $f_l(\boldsymbol{x})$ is the output of the discriminant function of the weighted SVM constructed for class l.

6 Empirical results

We evaluate the performance of the proposed probabilistic weighted multi-class support vector machines on the *AR face database* [22], [23], *CMU PIE face database* [24], and *FERET face database* [25]. Figure 1 (i), (ii), and (iii) displays the face images of a individual from the AR, CMU PIE, FERET face database. The effectiveness of the weighted multi-class support vector machines has also been tested on a synthetic dataset.

The AR face record contains of 26 different frontal subject faces of 126 individual, among them 56 females and 70 males. Individuals are collected in two different sessions divided by two weeks with variation in facial expressions, illumination condition, and occlusion [22, 23].In the CMU PIE face database, there are 41,368 face images of 68 persons (subject) each of 13 different poses, 43 different illumination conditions, and 4 different

(i)

(ii)

(iii)

Figure 1: Few face images of person from the (i) AR, (ii) CMU PIE (iii) FERET face database.

Classifier	Recognition rate (%)	
	1st Experimental Strategy	2nd Experimental Strategy
Probabilistic weighted multi-class support vector machines	**82.50 (38×38)**	**62.50 (36×36)**
Multi-class support vector machines	82.00 (38×38)	61.91 (36×36)

Table 1: Comparisons among the probabilistic weighted multi-class support vector machines and the multi-class support vector machines in terms of recognition rates using the performance evaluation over time (first) and performance evaluation with occluded images (second)experimental strategy on the AR face database. In table within the parentheses represent the number of features size.

Classifier	Avg. recognition rate (%)							
	first experimental strategy				second experimental strategy			
	k=5	k=10	k=15	k=20	k=5	k=10	k=15	k=20
Probabilistic weighted multi-class support vector machines	**75.31 (26×26)**	**86.56 (24×24)**	**88.65 (20×20)**	**89.04 (20×20)**	**80.86 (24×24)**	**86.53 (20×20)**	**92.78 (20×20)**	**98.18 (20×20)**
Multi-class support vector machines	75.28 (26×26)	86.52 (24×24)	88.59 (20×20)	88.96 (20×20)	80.32 (24×24)	85.86 (20×20)	91.89 (20×20)	97.49 (20×20)

Table 2: Comparisons of the probabilistic weighted multi-class support vector machines and the multi-class support vector machines in terms of average recognition rates for the performance evaluation with pose and expression variations (first) and performance evaluation with illumination variation (second) experimental strategy on the CMU PIE face database. Figures within the parentheses denote the number of features.

Classifier	Recognition rate (%)							
	FERET Tests September 1996 testing methodology				FRVT 2000 Tests May 2000 testing methodology			
	fafb	fafc	Dup I	Dup II	P1_probe	P2_probe	P3_probe	P4_probe
Probabilistic weighted multi-class support vector machines	**98.33 (20×20)**	**97.94 (18×18)**	**89.34 (22×22)**	**83.76 (18×18)**	**68.50 (20×20)**	**49.25 (22×22)**	**28.50 (22×22)**	**22.25 (24×24)**
Multi-class support vector machines	98.16 (20×20)	96.91 (18×18)	88.78 (22×22)	83.33 (18×18)	67.75 (20×20)	48.75 (22×22)	27.75 (22×22)	21.75 (24×24)

Table 3: Comparison of performances between the probabilistic weighted multi-class support vector machines and the multi-class support vector machines in terms of recognition rates using FERET Tests September 1996 testing methodology and FRVT 2000 Tests May 2000 testing methodology on the FERET face database. Figures within the parentheses denote the number of features.

expressions. The FERET face database [25] is used to measure the ability of the face recognition system to handle large databases, changes in people's appearance over time, variations in illumination, scale, and pose. Figure 1 (iii) shows example images of a subject from the FERET face database. In this work, experiments are carried out using two standard testing methodology, namely, i) FERET Tests September 1996 testing methodology, and ii) FRVT Tests May 2000 testing methodology. In FERET Tests September 1996 testing

methodology, the frontal face images of 1196 subjects are present. The training set contains 1196 face images, one image from each of 1196 distinct subjects. In this testing methodology, there are four test sets, namely, fafb, fafc, Dup I and Dup II. The test sets fafb, fafc, Dup I and Dup II contain 1195, 194, 722 and 234 images, respectively. In FRVT Tests May 2000 testing methodology, the face images of 200 subjects are present. The training set contains 200 frontal images, one image per subject from 200 distinct subjects. In this

testing methodology, there are four test sets, namely, P1_probe, P2_probe, P3_probe and P4_probe.

The comparison of performances between the probabilistic weighted multi-class support vector machines and the multi-class support vector machines in terms of recognition rates are illustrated in Table 1, 2, 3 on the AR, CMU-PIE, and FERET face database, respectively. From experimental results, it can be again observed that the performance of the probabilistic weighted multi-class support vector machines is better than the multi-class support vector machines in terms of recognition rate.

In this experiment, a *synthetic dataset E* containing 2D data from two different classes is randomly generated. In this dataset there are 50 data points, where 25 data points belong to one class and remaining 25 data points belong to another class. Let the dataset *E* can be defined as follows:

$$E = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^{50}; \quad \boldsymbol{x}_i \in \Re^2; \quad y_i \in \{+1, -1\} \qquad (25)$$

To test the effectiveness of the proposed probabilistic weighted multi-class support vector machines, the data present in the dataset *E* are separately

applied on both the multi-class support vector machines as well as on the probabilistic weighted multi-class support vector machines. The optimal separating hyperplane generated by the multi-class support vector machines and the probabilistic weighted multi-class support vector machines are shown in Figures 2(a) and 2(b), respectively.

The encircled data points are support vectors and the distance between the two dotted lines is the margin of separation between two classes in both Figures. The line between these two dotted lines is optimal separating hyperplane. In case of the multi-class support vector machines, 11 data points are present within the margin of separation region, as shown in Figure 2(a). Whereas, in case of the probabilistic weighted multi-class support vector machines, 10 data points are present within the margin of separation region, as shown in Figure 2(b). Therefore, the probabilistic weighted multi-class support vector machines successfully reduces the probability of misclassification, and produces better generalization than that with the multi-class support vector machines.
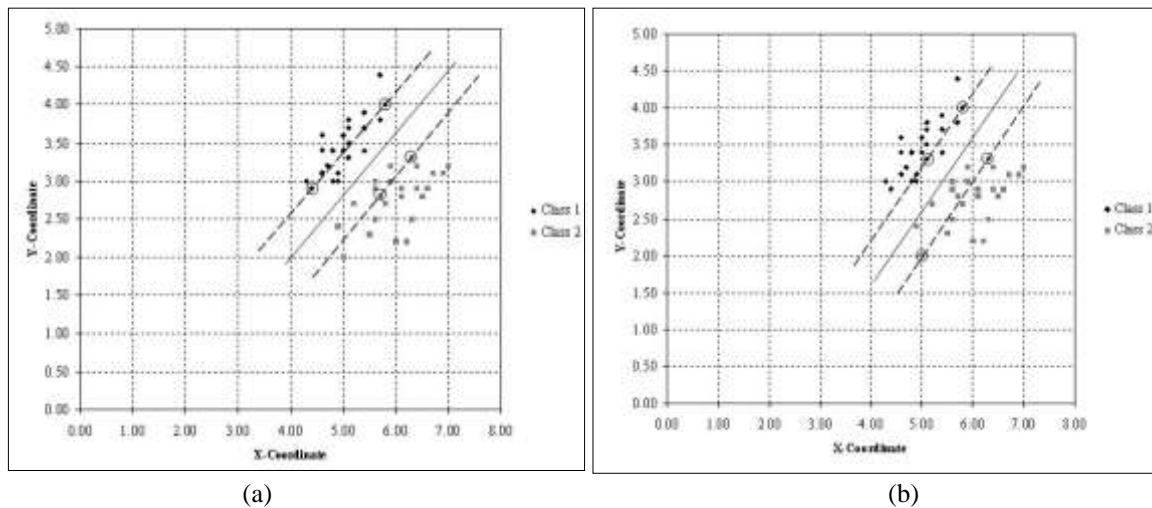

| (a) | (b) |

Figure 2: Comparative study in terms of the optimal separating hyperplane generation (a): multi-class support vector machines (b): proposed probabilistic weighted multi-class support vector machines, on the dataset *E*.

7   Conclusion

In this paper, we present the probabilistic weighted multi-class support vector machines for efficient face recognition. Support vector machines usually used for pattern classification and recognition as well as computer vision domains due to its high generalization ability. However, support vector machines have some limitations because it treats all the training data points of a given class uniformly. As a result, in presence of outliers the training algorithm of the support vector machines can make the decision boundary to be deviated severely from the optimal hyperplane. This limitation of support vector machines can be overcome by the weighted support vector machines where each data point is treated separately according to its weight. In the proposed probabilistic weighted multi-class support vector machines, a reliable weighting model is developed where

higher weights are assigned to reliable data points, and lower weights are assigned to outliers. These weights are generated by the probabilistic method; therefore it will take more computing times due to the weight generating algorithm. The training algorithm of the probabilistic weighted support vector machines learns the decision surface according to the relative importance of the training data. The proposed probabilistic weighted multi-class support vector machines have been constructed using a combination of weighted binary support vector machines and one-against-all decision strategy. Several experiments have been carried out on the AR, CMU PIE and FERET face databases using different experimental strategies. The facial features extracted by the G-2DFLD method are separately applied on both the proposed probabilistic multi-class support vector machines as well as on the weighted multi-class support vector machines for training, classification and recognition. The

experimental results show that the performance of the probabilistic weighted multi-class support vector machines is superior to the multi-class support vector machines in terms of recognition rate.

8 Acknowledgement

9 References

[1] L.J. Cao, K.S. Chau, W.K. Chong, H.P. Lee, and Q.M. Gu, "A comparison of PCA, KPCA and ICA for dimensionality reduction in support vector machine", Neurocomputing, Vol. 55, No. 1-2, pp. 321-336, 2003.
https://doi.org/10.1016/S0925-2312(03)00433-8

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

[3] C.J.C. Burges, "A tutorial on support vector machines for pattern recognition", *Data Mining and Knowledge Discovery*, Vol. 2, No. 2, pp. 121-167, 1998.
https://doi.org/10.1023/A:1009715923555

[4] X. Zhang, "Using class-center vectors to build support vector machines", *Proc. of the IEEE Signal Processing Society Workshop*, pp. 3-11, 1999.

[5] R. Herbrich, and J. Wetson, "Adaptive margin support vector machines for classification", *Proc of the Ninth International Conference on Artificial Neural Networks*, Vol. 2, pp. 880-885, 1999.
https://doi.org/10.1049/cp:19991223

[6] Q. Song, W. Hu, and W Xie, "Robust support vector machine with Bullet hole image classification", *IEEE Transactions on Systems, Man and Cybernetics*, Vol. 32, No 4, pp. 440-448, 2002.
https://doi.org/10.1109/TSMCC.2002.807277

[7] W.J. Hu, and Q. Song, "An accelerated decomposition algorithm for robust support vector machines", *IEEE Transactions on Circuits and Systems II*, Vol. 51, No. 5, pp. 234-240, 2004.
https://doi.org/10.1109/TCSII.2004.824044

[8] C. Lin, and S. Wang, "Fuzzy support vector machines", *IEEE Transactions on Neural Networks*, Vol. 13, No. 2, pp. 464-471, 2002.
https://doi.org/10.1109/72.991432

[9] C. Lin, and S. Wang, "Training algorithms for fuzzy support vector machines with noisy data", *Pattern Recognition Letters*, Vol. 25, No. 2, pp. 1647-1656, 2004.
https://doi.org/10.1016/j.patrec.2004.06.009

[10] L.J. Cao, H.P. Lee, and W.K. Chong, "Modified support vector novelty detector using training data with outliers", *Pattern Recognition Letters*, Vol. 24, No. 14, pp. 2479-2487, 2003.
https://doi.org/10.1016/S0167-8655(03)00093-X

[11] [11] T. Quan, X. Liu, and Q.Liu, "Weighted least squares support vector machine local region method for nonlinear time series prediction", *Applied Soft Computing*, Vol. 10, No. 2, pp. 562-566, 2010.
https://doi.org/10.1016/j.asoc.2009.08.025

[12] J. P. Hwang, S. Park, and E. Kim, "A new weighted approach to imbalanced data classification problem via support vector machine with quadratic cost function", *Expert Systems with Applications*, Vol. 38, No. 7, pp. 8580-8585, 2011.
https://doi.org/10.1016/j.eswa.2011.01.061

[13] L. Yu, "An evolutionary programming based asymmetric weighted least squares support vector machine ensemble learning methodology for software repository mining", *Information Sciences*, Vol. 191, pp. 31-46, 2012.
https://doi.org/10.1016/j.ins.2011.09.034

[14] Q. Ye, C. Zhao, S. Gao, and H. Zheng, "Weighted twin support vector machines with local information and its application", *Neural Networks*, Vol. 35, pp. 31-39, 2012.
https://doi.org/10.1016/j.neunet.2012.06.010

[15] Y. Shao, W.Chen, J. Zhang, Z. Wang, and N. Deng, "An efficient weighted Lagrangian twin support vector machine for imbalanced data classification", *Pattern Recognition*, Vol. 47, No. 9, pp. 3158-3167, 2014.
https://doi.org/10.1016/j.patcog.2014.03.008

[16] P. Xanthopoulos, and T. Razzaghi, "A weighted support vector machine method for control chart pattern recognition", *Computers & Industrial Engineering*, Vol. 70, pp. 134-149, 2014.
https://doi.org/10.1016/j.cie.2014.01.014

[17] X. Yang, L. Tan, and L. He, "A robust least squares support vector machine for regression and classification with noise", *Neurocomputing*, Vol. 140, pp. 41-52, 2014.
https://doi.org/10.1016/j.neucom.2014.03.037

[18] S. Chowdhury, J.K. Sing, D.K. Basu, and M. Nasipuri, "Face recognition by generalized two-dimensional FLD method and multi-class support vector machines", *Applied Soft Computing*, Vol. 11, No. 7, pp. 4282-4292, 2011.
https://doi.org/10.1016/j.asoc.2010.12.002

[19] C. Cortes, and V. Vapnik, "Support-vector network", *Machine Learning*, Vol. 20, No. 3, pp. 273-297, 1995.
https://doi.org/10.1007/BF00994018

[20] J. Platt, "Fast training of support vector machines using sequential minimal optimization", *Advances in Kernel Methods-Support Vector Learning*, MIT Press, Cambridge, pp. 185-208, 1999.

[21] S. Knerr, L. Personnaz, and G. Dreyfus, "Single-layer learning revisited: A stepwise procedure for building and training a neural network", *Neurocomputing*, Vol. 68, pp. 41-50, 1990.
https://doi.org/10.1007/978-3-642-76153-9_5

[22] A.M. Martinez, and R. Benavente, "The AR face database", *CVC Technical Report. #24*, June 1998.

[23] A.M. Martinez, and A.C. Kak, "PCA versus LDA", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 23, No. 2, pp. 228-233, 2001.

https://doi.org/10.1109/34.908974

[24] T. Sim, S. Baker, and M. Bsat, "The CMU pose, illumination, and expression (PIE) database", *Proc. of the Fifth IEEE International Conference on Automatic Face and Gesture Recognition*, pp. 46-51, 2002.

[25] P.J. Phillips, H. Wechsler, J. Huang, and P.J. Rauss, "The FERET database and evaluation procedure for face-recognition algorithms", *Image and Vision Computing*, Vol. 16, No. 5, pp. 295-306, 1998.
https://doi.org/10.1016/S0262-8856(97)00070-X

[26] Y. Zhang, Z. Yang, H. Lu, X. Zhou, P. Phillips, Q. Liu, and S. WANG, "Facial emotion recognition based on biorthogonal wavelet entropy, fuzzy support vector machine, and stratified cross validation", *Emotion-aware Mobile Computing*, Vol. 4, pp. 8375 – 8385, 2016.
https://doi.org/10.1109/ACCESS.2016.2628407

[27] S. Wang, P. Phillips, Z. Dong, Y. Zhang, "Intelligent facial emotion recognition based on stationary wavelet entropy and Jaya algorithm", *Neurocomputing*, Vol. 272, pp. 668-676, 2018.
https://doi.org/10.1016/j.neucom.2017.08.015

[28] A.A. Aburomman, M.B.I. Reaz, "A novel SVM-kNN-PSO ensemble method for intrusion detection system", Applied Soft Computing. Vol. 38, pp. 360–372. 2016.
https://doi.org/10.1016/j.asoc.2015.10.011

[29] P. Huang, C. Chen, Z. Tang, and Z. Yang, "Feature extraction using local structure preserving discriminant analysis", *Neurocomputing*, Vol. 140, pp. 104-113, 2014.
https://doi.org/10.1016/j.neucom.2014.03.031

[30] P. Huang, C. Chen, Z. Tang, and Z. Yang, "Discriminant similarity and variance preserving projection for feature extraction," *Neurocomputing*, Vol. 139, pp. 180-188, 2014.
https://doi.org/10.1016/j.neucom.2014.02.047

[31] L. Shi, X. Wang, and Y. Shen, "Research on 3D face recognition method based on LBP and SVM" *Optik*, Vol. 220, pp., 2020.
https://doi.org/10.1016/j.ijleo.2020.165157

[32] L. Hu, J. Cui"Digital image recognition based on Fractional-order-PCA-SVM coupling algorithm" *Measurement*, Vol. 145, pp. 150 -159, 2019.
https://doi.org/10.1016/j.measurement.2019.02.006

[33] I. Dagher, F. Azar"Improving the SVM gender classification accuracy using clustering and incremental learning", *WileyExpert System*, Vol. 145, pp. 1 -17, 2019.
https://doi.org/10.1111/exsy.12372

[34] N.B. Kar, K.S. Babu, A.K. Sangaiah, S Bakshi"Face expression recognition system based on ripplet transform type II and least square SVM"*Multimedia Tools Application,* Vol. 78, pp.4789–4812, 2019.
https://doi.org/10.1007/s11042-017-5485-0