Support Vector Machines in Face Recognition with Occlusions

Hongjun Jia and Aleix M. Martinez
The Department of Electrical and Computer Engineering
The Ohio State University, Columbus, OH 43210, USA
jia.22@osu.edu aleix@ece.osu.edu

Abstract

Support Vector Machines (SVM) are one of the most useful techniques in classification problems. One clear example is face recognition. However, SVM cannot be applied when the feature vectors defining our samples have missing entries. This is clearly the case in face recognition when occlusions are present in the training and/or testing sets. When k features are missing in a sample vector of class 1, these define an affine subspace of k dimensions. The goal of the SVM is to maximize the margin between the vectors of class 1 and class 2 on those dimensions with no missing elements and, at the same time, maximize the margin between the vectors in class 2 and the affine subspace of class 1. This second term of the SVM criterion will minimize the overlap between the classification hyperplane and the subspace of solutions in class 1, because we do not know which values in this subspace a test vector can take. The hyperplane minimizing this overlap is obviously the one parallel to the missing dimensions. However, this condition is too restrictive, because its solution will generally contradict that obtained when maximizing the margin of the visible data. To resolve this problem, we define a criterion which minimizes the probability of overlap. The resulting optimization problem can be solved efficiently and we show how the global minimum of the error term is guaranteed under mild conditions. We provide extensive experimental results, demonstrating the superiority of the proposed approach over the state of the art.

1. Introduction

The appearance-based approach to face recognition has resulted in the design of highly successful computer algorithms in the last several years [13]. In this approach, the brightness values of the image pixels are reshaped as a vector and then classified using a classification algorithm. A classification algorithm that has successfully been used in this framework is the well-known Support Vector Machines (SVM) [11], which can be applied to the original appearance space or a subspace of it obtained after applying a feature extraction method [8, 3, 10].

A major disadvantage of the appearance-based framework is that it cannot be directly used when some of the features (*i.e*. face pixels) are occluded. In this case, the values for those dimensions are unknown. To date, the major approach used to resolve this problem is as follows. First, learn the appearance representation of the face as stated above using non-occluded faces. When attempting to recognize a partially occluded face, use only the visible dimensions (*i.e*. features) common to the model and the test images. This approach can be implemented using subspace techniques [1, 2, 6] and sparse representations [12]. Most methods do not however address the problem of constructing a model (or classifier) from occluded images.

In Fig. 1 we show the three scenarios a realistic face recognition system ought to allow. In the first row, we have the most studied case – non-occluded faces in training and occluded faces in testing. The second and third rows illustrate two other cases: *a)* training with occluded and non-occluded faces, and *b)* training with occluded faces only. However, the approaches introduced above rely on a non-occluded training set.

In this paper we derive a criterion for SVM that can be employed in the three cases defined in Fig. 1. Note that the classical criteria of SVM cannot be applied to any of the three cases, because SVM assumes all the features are visible. In the sections to follow, we derive a criterion that can work with missing components of the sample and testing feature vectors. We will refer to the resulting algorithm as Partial Support Vector Machines (PSVM) to distinguish it from the standard criteria used in SVM.

The goal of PSVM is, nonetheless, similar to that of the standard SVM – to look for a hyperplane that separate the samples of any two classes as much as possible. In contrast with traditional SVM, in PSVM the separating hyperplane will also be constrained by the incomplete data. In the proposed PSVM, we treat the set of all possible values for the missing entries of the incomplete training sample as an *affine space* in the feature space to design a criterion which minimizes the probability of overlap between

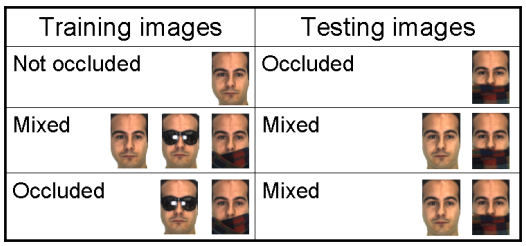

Figure 1. Different cases of face recognition with occlusions.

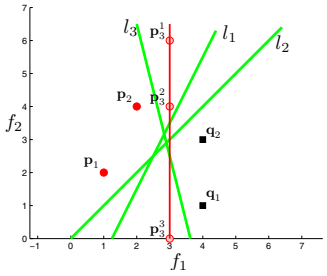

Figure 2. Classical SVM solutions for different (potential) filling-ins. $\{\mathbf{p}_1, \mathbf{p}_2\}$ and $\{\mathbf{q}_1, \mathbf{q}_2\}$ are in classes 1 and 2, respectively. The incomplete feature vector $\mathbf{p}_3 = (3, \bullet)^T \in$ class 1.

this affine space and the separating hyperplane. To model this, we incorporate the angle between the affine space and the hyperplane in the formulation. The resulting objective function is shown to have a global optimal solution under mild conditions, which require that the convex region defined by the derived criterion is close to the origin. Experimental results demonstrate that the proposed PSVM approach provides superior classification performances than those defined in the literature.

2. Face Recognition with Occlusions

2.1. Classical SVM algorithm

In the training stage of SVM, a hyperplane is obtained from a complete data set with labels by maximizing the geometric margin. Let the training set have n samples $\{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$, with labels $y_i = \pm 1$, $i = 1, \ldots, n$, each of them defined by a feature set $F = \{f_1, f_2, \ldots, f_d\}$. In this setting, a complete data sample can be treated as a point in a d-dimensional space, $\mathbf{x}_i = (x_{i1}, \ldots, x_{id})^T \in \mathbb{R}^d$. The best hyperplane, $\mathbf{w}^T\mathbf{x} = b$, to separate two classes is achieved by maximizing the geometric margin,

$$\max_{\mathbf{w},b} \frac{1}{\|\mathbf{w}\|}, \quad s.t. \quad y_i(\mathbf{w}^T\mathbf{x}_i - b) \geq 1, \; i = 1, \ldots, n, \quad (1)$$

where $\|\cdot\|$ is the 2-norm of a vector. Eq. (1) is equivalent to minimizing the quadratic term $\frac{1}{2}\|\mathbf{w}\|^2$ with the same constraints, which has an efficient solution [11].

Typically, the original set will not be linearly separable. To resolve this problem, it is common to define a soft margin by including the slack variables $\xi_i \geq 0$ and a regularizing parameter $C > 0$,

$$\begin{aligned} \min_{\mathbf{w},\xi,b} \quad & \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{n}\xi_i, \quad (2) \\ s.t. \quad & y_i(\mathbf{w}^T\mathbf{x}_i - b) \geq 1 - \xi_i, \quad i = 1, \ldots, n. \end{aligned}$$

However, when some of the features are missing, these distances can no longer be computed. One possible way to solve this problem is to attempt to fill-in the missing entries of each feature vector before using SVM. Unfortunately, the filling-in step leads us to a worst problem: *how to know the correct (or appropriate) values of the missing entries?*

If we consider the affine space S_i defined by all possible fill-ins of the corresponding partial data $\mathbf{x}_i$ as one single data unit, the ideal solution to the partial data classification is that it can classify the affine space correctly. That means the hyperplane should ideally be parallel to all the affine spaces defined by the incomplete data, which is generally impossible.

To illustrate this point, we show a simple example in Fig. 2. In this figure, two sets of points, $\{\mathbf{p}_1, \mathbf{p}_2\}$ and $\{\mathbf{q}_1, \mathbf{q}_2\}$, defined on the feature plane $\{f_1, f_2\}$ and corresponding to classes 1 and 2, are generated. The additional sample vector $\mathbf{p}_3$ has a known value for f_1 but a missing entry in f_2. Three possible filling-ins of $\mathbf{p}_3$ are shown in the figure – denoted $\mathbf{p}_3^1$, $\mathbf{p}_3^2$ and $\mathbf{p}_3^3$. For each of them, the classical SVM would give the hyperplanes denoted by l_1, l_2 and l_3, respectively. We can see that none of these three hyperplanes can give correct classifications for all $\mathbf{p}_i^j$.

To resolve the problem illustrated above, we resort to a new solution which focuses on classifying partial data correctly with the help of probabilities. In particular, we show how to add a new term to (1).

2.2. The angle between the hyperplane and the affine space

The values of the missing elements of our d-dimensional feature vector define an affine space in $\mathbb{R}^d$. We now show that the correct classification probability of a hyperplane on the affine space is determined by two factors: *a)* the relative position between them, and *b)* the classification result of the actual missing elements.

To get started, let us assume that there is only one missing element in $\mathbf{x}$ in class 1. Denote the affine space defined by this missing element as S, and the hyperplane which separates the two classes by l. This hyperplane can be readily obtained with the standard SVM criterion by simply substituting the missing entry by that of the mean feature vector $\bar{\mathbf{x}}$. If the hyperplane l and the affine space S are not parallel to each other, the intersection between the two divides the

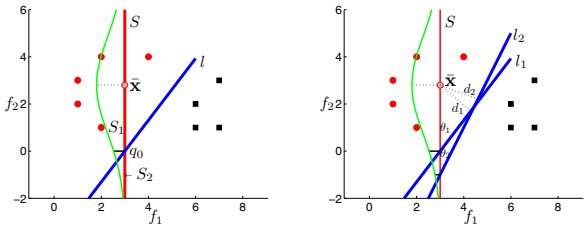

Figure 3. The Probability of Correct Classification (PCC) of a hyperplane. (a) Assuming a Gaussian distribution of S, (b) the angle between S and l_i is proportional to the distance $d(\bar{\mathbf{x}}, q_0)$.

affine space into two (non-overlapping) parts, S_1 and S_2. This partition is illustrated in Fig. 3(a). We see from this figure that the possible values of the missing entry that fall in S_1 will be correctly classified as class 1, whereas the values now in S_2 will be misclassified. Using this argument, we can compute the Probability of Correct Classification (PCC) of l over the affine space S as

$$PCC(l, S) = \int_{q \in S_1} p(q) dq, \tag{3}$$

where $p(q)$ is the probability density function and $q \in S$.

Under the above defined model, the goal is to minimize the probability of overlap between the most probable values of the samples in class 1, *i.e.*, we want to prevent l to cut over plausible values of the missing entries. To calculate this probability, we assume the sample data is Gaussian distributed, $p(q) \in N(\bar{x}, \sigma)$ with $\bar{\mathbf{x}}$ the mean and σ the variance. This is shown in Fig. 3(a). The intersection between S and l is at q_0. Maximizing PCC is thus equivalent to maximizing the distance between the value given by $\bar{\mathbf{x}}$ and q_0, $d(\bar{x}, q_0)$.

Note that for a fixed set of sample vectors, the angle between the subspaces S and l, $\theta(S, l)$, decreases proportionally to the increase of $d(\bar{\mathbf{x}}, q_0)$, Fig. 3(b). Hence, $\theta(S, l)$ is the term needed to account for the possible values of the missing elements of $\mathbf{x}$.

2.3. The objective function

We are now in a position to formulate the criterion which will properly model the aforementioned penalty term. This will take us to the definition of the PSVM algorithm. We start by presenting the solution for the linearly separable case.

To address the incomplete data problem efficiently, we first need to define an occlusion mask $\mathbf{m}_i \in \mathbb{R}^d$ for each sample vector $\mathbf{x}_i$, $i = 1, \ldots, n$. The elements of the occlusion mask $\mathbf{m}_i$ will be 0 wherever the corresponding feature in $\mathbf{x}_i$ is occluded and 1 otherwise. The affine space which is formed by all possible filling-ins of incomplete sample $\mathbf{x}_i$ is denoted S_i, and the hyperplane separating the two classes by $l : \mathbf{w}^T \mathbf{x} = b$, where $\mathbf{w} = (w_1, \ldots, w_d)^T$.

The angle between S_i and l is the same as the angle between the orthogonal space of S_i, $S_i^{\perp}$, and the normal vector of l, $\mathbf{w}$. The projection of $\mathbf{w}$ on $S_i^{\perp}$ is $\mathbf{w}_i^1 = \mathbf{w} \odot \mathbf{m}_i$, where $\odot$ is the Hadamard product (*i.e.* the element-by-element multiplication of two vectors, $\mathbf{a} \cdot \mathbf{b} = (a_1 b_1, \ldots, a_p b_p)$, $\mathbf{a}, \mathbf{b} \in \mathbb{R}^p$). The angle between S_i and l, $\theta(S_i, l)$, is given by

$$\cos\theta(S_i, l) = \cos\theta(S_i^{\perp}, \mathbf{w}) = \frac{\|\mathbf{w}_i^1\|}{\|\mathbf{w}\|}. \tag{4}$$

A new term can now be formulated as a weighted summation over (4), *i.e.* $\sum_{i=1}^{n} K_i \|\mathbf{w}_i^1\| / \|\mathbf{w}\|$, where the weights $K_i \geq 0$ are chosen to be positive when $\mathbf{x}_i$ is incomplete and zero otherwise. To obtain the highest possible PCC, this term is to be maximized. This can be readily achieved by adding it to SVM optimization problem as follows

$$\begin{aligned} \max_{\mathbf{w},b} \quad & \frac{1}{\|\mathbf{w}\|} + K \sum_{i=1}^{n} K_i \frac{\|\mathbf{w}_i^1\|}{\|\mathbf{w}\|} \\ s.t. \quad & y_i(\mathbf{w}^T \bar{\mathbf{x}}_i - b) \geq 1, \quad i = 1, \ldots, n, \end{aligned} \tag{5}$$

where $K > 0$ is the regularizing parameter to control the overall tradeoff between the generalization performance of the hyperplane (defined by the maximal geometric margin, $1/\|\mathbf{w}\|$) and the classification accuracy on the incomplete data.

The objective function in (5) is neither linear nor quadratic, which usually does not yield efficient solutions. Nonetheless, we can transform (5) into a more tractable criterion (with the quadratic form of $\mathbf{w}$ in both denominator and numerator) as follows

$$\begin{aligned} \max_{\mathbf{w},b} \quad & \frac{1 + K \sum_{i=1}^{n} K_i \|\mathbf{w}_i^1\|^2}{\|\mathbf{w}\|^2} \\ s.t. \quad & y_i(\mathbf{w}^T \bar{\mathbf{x}}_i - b) \geq 1, \quad i = 1, \ldots, n. \end{aligned} \tag{6}$$

2.4. Optimization

We now show how to solve the above optimization problem in the linearly separable case. Without loss of generality, let us rework the above derived SVM solution (which was defined in $\mathbb{R}^n$, n the number of samples) in $\mathbb{R}^d$, d the number of dimensions. We can achieve this by using the following equality $\sum_{i=1}^{d} u_i w_i^2 = K \sum_{i=1}^{n} K_i \|\mathbf{w}_i^1\|^2$, which yields

$$\begin{aligned} \max_{\mathbf{w},b} \quad & f(\mathbf{w}) = \frac{1 + \sum_{i=1}^{d} u_i w_i^2}{\sum_{i=1}^{d} w_i^2} \\ s.t. \quad & y_i(\mathbf{w}^T \bar{\mathbf{x}}_i - b) \geq 1, \quad i = 1, \ldots, n. \end{aligned} \tag{7}$$

Since b only appears in the linear constraint as an offset of the separation hyperplane, it will not affect the convexity of the defined region. Therefore, in the following analysis, we focus on $\mathbf{w}$, which still needs to be shown to yield

convex regions to allow optimal solutions wrt the derived criterion.

To do this, note that the optimization problem in (7), with respect to $\mathbf{w}$, is defined on a polyhedral convex region in a d-dimensional space. We see that this region in the space of $\mathbf{w}$ does not cover the origin point $\mathbf{w} = \mathbf{0}$. If the above statement were not true, then we would need to use $\mathbf{0}$ to replace $\mathbf{w}$ in the constraint to get $y_i(-b) \geq 1, i = 1, \ldots, n$. Since y_i is either ± 1, and noting that each of these two values must be assigned at least once to y_i, we can choose $y_j = +1$ and $y_k = -1$ ($j, k \in \{1, \ldots, n\}$ and $j \neq k$) to get $b \geq 1$ and $-b \geq 1$. This results in a null set.

The target function is not convex on $\mathbf{w}$. Nonetheless, it has some good properties we can exploit to facilitate the optimization. Consider two points $\mathbf{w}_1$ and $\mathbf{w}_2$ ($\mathbf{w}_2 = r\mathbf{w}_1, r > 1$), then the corresponding function values satisfy

$$\begin{aligned} f(\mathbf{w}_1) &= \frac{1}{\sum_{i=1}^d w_{1i}^2} + \frac{\sum_{i=1}^d u_i w_{1i}^2}{\sum_{i=1}^d w_{1i}^2} \\ &\leq \frac{1}{\sum_{i=1}^d (r w_{1i})^2} + \frac{\sum_{i=1}^d u_i (r w_{1i})^2}{\sum_{i=1}^d (r w_{1i})^2} \\ &= \frac{1}{\sum_{i=1}^d w_{2i}^2} + \frac{\sum_{i=1}^d u_i w_{2i}^2}{\sum_{i=1}^d w_{2i}^2} = f(\mathbf{w}_2). \quad (8) \end{aligned}$$

The above result implies that the objective function is monotonically increasing on a line passing through the origin.

Since (7) is defined on a convex region not covering the original point ($\mathbf{w} = \mathbf{0}$) and has the monotonically increasing property proved above, the optimal solution of (7) must be on the boundary of that region. Therefore, if we use the solution to the classical SVM as the initial point (on a complete training set), we can apply a gradient-descent method to solve for (7). The question is whether this procedure can provide the *global optimal* solution wrt our criterion. We can now show that under mild conditions, this global optimal is guaranteed.

To see this, let us maximize the lower bound of the objective function with an additional constraint, *i.e.* $(1 + \sum_{i=1}^d u_i w_i^2)/(\sum_{i=1}^d w_i^2) \geq \gamma$, or $\sum_{i=1}^d (\gamma - u_i) w_i^2 \leq 1$. This process yields the following optimization problem

$$\begin{aligned} \max_{\mathbf{w},b} \quad & \gamma \quad (9) \\ s.t. \quad & \sum_{i=1}^d (\gamma - u_i) w_i^2 \leq 1 \text{ and } y_i(\mathbf{w}^T \bar{\mathbf{x}}_i - b) \geq 1. \end{aligned}$$

Note that for any fixed value of $\gamma \geq \max\{u_1, \ldots, u_d\}$, the first constraint in (9) defines a convex region in the d-dimensional space. Therefore, the target function and the constraints are convex, which ensures a global optimization solution. This means that a global solution exists under the condition

$$\gamma_{max} \geq \gamma_0 = \max\{u_1, \ldots, u_d\}, \quad (10)$$

where γ_{max} is the solution to (9). This is indeed a very mild condition. In fact, it hold in all our experiential results to be presented later.

We see that whenever this condition holds, our problem is convex and can be solved using the general structure of a Second Order Cone Program (SOCP) [5]. With γ_{max} and the corresponding solution $\mathbf{w}_{max}, b_{max}$, it can be readily shown that any $\gamma \in (\gamma_0, \gamma_{max})$ will provide a solution for (9), since

$$\sum_{i=1}^d (\gamma - u_i) {w_{max}}_i^2 \leq \sum_{i=1}^d (\gamma_{max} - u_i) {w_{max}}_i^2 \leq 1. \quad (11)$$

Hence, the bisection search over $\gamma \in (\gamma_0, +\infty) \subset \mathbb{R}^+$ is an efficient and direct way to determine the value of γ_{max}.

2.5. Nonlinearly separable

Many classification problems are not linearly separable. These cases can be tackled with the inclusion of a soft margin. In this case, the slack variables $\xi = (\xi_1, \ldots, \xi_n)^T$ and the regularizing parameter $C > 0$ need to be added to (6). Since some incomplete data may now be incorrectly classified, we need to adjust the weights of the angle term according to the value of the slack variables. This can be done as follows,

$$\begin{aligned} \max_{\mathbf{w},b} \quad & \frac{1 + K \sum_{i=1}^n sgn(1-\xi_i) K_i \|\mathbf{w}_i^1\|^2}{\|\mathbf{w}\|^2} - C \sum_{i=1}^n \xi_i \quad (12) \\ s.t. \quad & y_i(\mathbf{w}^T \bar{\mathbf{x}}_i - b) \geq 1 - \xi_i, \xi_i \geq 0, i = 1, \ldots, n, \end{aligned}$$

where $sgn(\cdot)$ is the sign function to adjust the maximization of the corresponding cosine term based on the potential values taken by the missing entries of the incomplete feature vectors.

Although (12) is defined on a convex region, this equation is difficult to solve because the function sgn in not continuous. As it is common in such cases, we choose to optimize a closely related cost function

$$\begin{aligned} \max_{\mathbf{w},b} \quad & \frac{1 + K \sum_{i=1}^n (1-\xi_i) K_i \|\mathbf{w}_i^1\|^2 - C \sum_{i=1}^n \xi_i \|\mathbf{w}\|^2}{\|\mathbf{w}\|^2} \quad (13) \\ s.t. \quad & y_i(\mathbf{w}^T \bar{\mathbf{x}}_i - b) \geq 1 - \xi_i, \xi_i \geq 0, i = 1, \ldots, n, \end{aligned}$$

This defines the PSVM algorithm as

$$\begin{aligned} \max_{\mathbf{w},b} \quad & g(\mathbf{w}, \xi) = \frac{1 + \sum_{i=1}^d u_i(\xi) w_i^2}{\sum_{i=1}^d w_i^2} \quad (14) \\ s.t. \quad & y_i(\mathbf{w}^T \bar{\mathbf{x}}_i - b) \geq 1 - \xi_i, \xi_i \geq 0, i = 1, \ldots, n, \end{aligned}$$

where $u_i(\xi)$ is a function of ξ. Using the solution of (2) as an initialization and using the iterative method defined above to solve for $\mathbf{w}$ and ξ, we arrive at the desirable solution. To see this, note that if ξ is fixed, $g(\mathbf{w}, \xi)$ can be

maximized in the same way as the linearly separable case presented above; if $\mathbf{w}$ is fixed, $g(\mathbf{w}, \xi)$ becomes an easy linear optimization problem defined on a convex region.

After the hyperplane that separates the two classes has been learned, it can be readily used to classify a new test feature vector. If the test image is incomplete, however, we need to first determine the probability of its values. To do this, we will use the probabilistic view defined earlier. This we do in the section to follow.

2.6. Multi-weight data reconstruction

A SVM algorithm was derived to find the optimal hyperplane separating two classes with incomplete data. However, a complete test vector is needed for classification. To determine the values of the missing elements from those in the complete set, a linear least squares method can be applied. Here, we derive a multi-weight linear least squares approach.

For a test image $\mathbf{t}$, we define $\tilde{\mathbf{m}} \in \mathbb{R}^d$ as its occlusion mask. We use all $\mathbf{m}_i$ to form the occlusion mask of the training set, $\mathbf{M} = [\mathbf{m}_1, \ldots, \mathbf{m}_n]$. Let $\mathbf{M}^j$ denote the j^{th} row of this matrix. $\mathbf{M}^j$ defines the sample images that can be used to reconstruct the j^{th} image pixel of $\mathbf{t}$, t_j. Note that since each $\mathbf{M}^j$ has n values, there are 2^n possible patterns of features that can be used to reconstruct t_j. Let these patterns be labeled with the index l, with $l = 1, \ldots, 2^n$. Denote L_l as the set containing the indices of those training samples with observed values in the l^{th} pattern.

Now consider those features in the feature set F that can be reconstructed using the same pattern l, and denote these features Δ_l. The set Δ_l can be further divided into two subsets Γ_l and Π_l, where Γ_l contains the indices of the observable features in $\mathbf{t}$ and Π_l defines the indices of the occluded ones. Thus, $\Gamma_l \cup \Pi_l = \Delta_l$, $\Gamma_l \cap \Pi_l = \varnothing$, and we can attach the superscript $(\cdot)^{\Gamma_l}$ (or $(\cdot)^{\Pi_l}$) to a vector to denote the corresponding part by keeping only those elements with the indices in Γ_l (or Π_l). Using this notations, a linear approximation for the pattern l can be expressed as $\mathbf{t}^{\Gamma_l} \approx \sum_{j \in L_l} \omega_j^l \mathbf{x}_j^{\Gamma_l}$, where the weights $\{\omega_j^l | j \in L_l\}$ are given by

$$\arg \min_{\{\omega_j^l | j \in L_l\}} \left\| \mathbf{t}^{\Gamma_l} - \sum_{j \in L_l} \omega_j^l \mathbf{x}_j^{\Gamma_l} \right\|_2 . \tag{15}$$

The weights calculated in (15) can be used to give the estimation of the missing part on pattern l,

$$\hat{\mathbf{t}}^{\Pi_l} = \sum_{j \in L_l} \omega_j^l \mathbf{x}_j^{\Pi_l} . \tag{16}$$

If for some pattern l, the feature set Π_l is not empty but Γ_l is, it means that the corresponding weights cannot be computed. In this case, we use the average value of the training set to determine the most probable value of the missing entries (*i.e*. the value with highest probability term assuming the data is Normally distributed).

Figure 4. (a-f) Shown here are the six images of the first session for one of the subjects in the AR face database.

3. Experimental Results

In this section, several experiments are implemented to show the effectiveness of the proposed PSVM algorithm by comparing it with the state-of-the-art on two popular data-sets with synthetic and real occlusions. These data-sets are the AR face database [7] and the FRGC (Face Recognition Grand Challenge) version 2 data-set [9].

The AR face database contains frontal view images of over 100 individuals. Here, we use a total of 12 images per person. Fig. 4 shows the first six images taken during a first session. We will labeled the pictures from the first session *a* through *f*, and those of the second session *a'* through *f'*. All images are cropped and resized to 29×21 pixels as shown in Fig. 4. The locations of the eyes, nose and mouth are used to align the faces. For FRGC data-set, we choose 100 subjects, and 8 images for each subject (two sessions), and resize images to 30×26 with fixed eye location.

The parameters $\{K_1, \ldots, K_n\}$ controlling the relative weights among different incomplete observations are set to 1. The regularizing constant K (or equivalently, the norm of $\mathbf{u}$), controlling the tradeoff between the accuracy and the generalization, needs to be fixed. We will use a set of different $\|\mathbf{u}\|$ chosen from $\{1, 10, 20, 40\}$ to compute the hyperplane. The occlusion masks $\mathbf{m}_i$, are constructed using a skin color detector learned from an independent set of face images.

3.1. Synthetic occlusions

Occlusions are added to the training images by overlaying a black square of $s \times s$ pixels in a random location. Fig. 5(a) shows the results with $s = 0, 3, 6, 9, 12$ on AR database. We use the neutral, happy and sad faces in the first session (*a*, *b*, and *c*) in AR database for training, and the screaming face (*d*) for testing. Next, we use the images of the first session (*a*, *b*, *c*, *d*) for training, and the duplicates (*a'*, *b'*, *c'*, *d'*) for testing. The results are demonstrated as the curves AR(*d*), AR(*a'*) - AR(*d'*) in Fig. 5(a). Note the $s \times s$ occlusion masks are randomly added to the images in the training and testing sets. Similarly, we run two experiments on FRGC data-set with the same synthetic occlusion

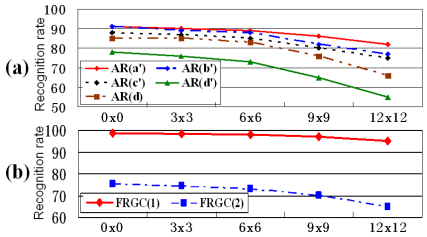

Figure 5. Classification accuracy with synthetic occlusions on the AR database and the FRGC data-set.

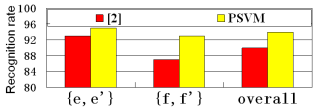

Figure 6. Experimental results for testing data with occlusions only. Training and testing set: $\{a, b, c, a', b', c'\}$,$\{e, f, e', f'\}$.

Training set	Testing Set	PSVM	[4]
[*a,e,f*]	[*b,c,d*]	88.9	85.7
[*a',e',f'*]	[*b',c',d'*]	90.8	84.7
[*a,b,c,e,f*]	[*d*]	88.2	82.0
[*a,b,c,e,f*]	[*d'*]	58.8	52.0
[*a,b,c,e,f,a',b',c',e',f'*]	[*d,d'*]	83.5	75.5

Table 1. Experimental results (recognition rate in percentages) with a variety of training and testing sets.

Training set	Testing Set	PSVM	[4]	NN_2	NN_1
[*e,f*]	[*a*]	96.0	89.0	45.0	79.0
[*e,f*]	[*a'*]	79.4	71.0	31.0	50.0
[*e,f*]	[*b,c,d*]	80.0	72.0	31.7	59.7
[*e,f*]	[*b',c',d'*]	58.7	47.3	20.3	32.7
[*e,f*]	[*e',f'*]	57.0	55.0	25.5	29.0
[*e,f,e',f'*]	[*b,c,d,b',c',d'*]	86.6	76.2	31.3	56.5
[*e,f,e',f'*]	[*a,a'*]	96.4	95.0	48.5	83.0

Table 2. Experimental results (recognition rate in percentages) with incomplete data in the training set.

mask and show results in Fig. 5(b). We first use two images of each session for the training purpose and the other two images for testing, and then use one whole session of each subject for training and the other one for testing. The curves FRGC(1) and FRGC(2) in Fig. 5(b) show the corresponding results. We see that in all cases, occlusions of up to 6×6 pixels do not affect the recognition rates.

3.2. Real occlusions

In [2], the authors use the images $\{a,b,c,a',b',c'\}$ for training and the images $\{e,f,e',f'\}$ for testing. The results of this approach are now compared to those obtained with the approach presented in this paper, Fig. 6.

In [4], the authors present a method with state-of-the-art recognition rates. In their experiments, the authors use a variety of training and testing sets. In Tables 1 and 2 we show the recognition rates obtained with their method and the PSVM approach derived in this paper. Table 2 presents the most challenging cases, some of which include $\sim 50\%$ occlusions in training and testing. To further illustrate the difficulty of the task, we have included the results obtained with a simple nearest neighbor (NN) approach with the 2- and 1-norms, NN_2 and NN_1. For example, we see that when the training set is $\{e, f, e', f'\}$ and the testing set is $\{b, c, d, b', c', d'\}$, we boost the results from 56.5% for the NN_1 algorithm to 86.6% for PSVM.

4. Conclusion

We have introduced a SVM approach for face (object) recognition with partial occlusions. The proposed algorithm allows for partial occlusions to occur in both, the training and testing sets. To achieve this goal, the derived algorithm incorporates an additional term to the SVM formulation indicating the probable range of values for the missing entries. We have shown that the resulting criterion is convex under very mild conditions. The proposed method has then been shown to obtain higher recognition rates than the algorithms defined in the literature in a variety of experiments.

Acknowledgments

This research was supported in part by the National Science Foundation, grant 0713055, and the National Institutes of Health, grant R01 DC 005241.

References

[1] R. M. Everson and L. Sirovich. Karhunen-Loeve procedure for gappy data. *Journal of the Optical Society of America*, 12(8):1657–1664, 1995.

[2] S. Fidler, D. Skočaj, and A. Leonardis. Combining reconstructive and discriminative subspace methods for robust classification and regression by subsampling. *IEEE Trans. PAMI*, 28(3):337–350, 2006.

[3] B. Heisele, T. Serre, and T. Poggio. A component-based framework for face detection and identification. *IJCV*, 74(2):167–181, 2007.

[4] H. Jia and A. M. Martinez. Face recognition with occlusions in the training and testing sets. *Proc. Conf. Automatic Face and Gesture Recognition*, 2008.

[5] M. Lobo, L. Vandenberghe, S. Boyd, and H. Lebret. Applications of second-order cone programming. *Lin. Alg. and Its Appl.*, 284:183–228, 1998.

[6] A. M. Martinez. Recognizing imprecisely localized, partially occluded and expression variant faces from a single sample per class. *IEEE Trans. PAMI*, 24(6):748–763, 2002.

[7] A. M. Martinez and R. Benavente. The AR face database. *CVC Tech. Rep. No. 24*, 1998.

[8] E. Osuna, R. Freund, and F. Girosit. Training support vector machines: an application to face detection. *Proc. of CVPR*, pages 130–136, 1997.

[9] P. J. Phillips, P. J. Flynn, T. Scruggs, K. W. Bowyer, J. Chang, K. Hoffman, J. Marques, J. Min, and W. Worek. Overview of the Face Recognition Grand Challenge. *Proc. of CVPR*, 2005.

[10] Q. Tao, D. Chu, and J. Wang. Recursive support vector machines for dimensionality reduction. *IEEE Trans. NN*, 19(1):189–193, 2008.

[11] V. Vapnik. *Statistical Learning Theory*. John Wiley and Sons, New York, 1998.

[12] J. Wright, A. Yang, A. Ganesh, S. Sastry, and Y. Ma. Robust face recognition via sparse representation. *IEEE Trans. PAMI*, 31(2):210–227, 2009.

[13] W. Zhao, R. Chellappa, P. J. Phillips, and A. Reosenfeld. Face recognition: A literature survey. *ACM Computing Survey*, 34(4):399–485, 2003.